Plural Semantics for Natural Language Understanding

A Computational Proof-Theoretic Approach

Thesis
presented to the Faculty of Arts
of the University of Zurich
for the degree of Doctor of Philosophy

by
Uta Schwertel
of
Germany

Accepted on the recommendation of

Professor Michael Hess, Ph.D.
Norbert E. Fuchs, Ph.D.

Zurich, May 2005

Acknowledgements

I developed this thesis as a research assistant and Ph.D. student in the Requirements Engineering Research Group at the Department of Information Technology (*Institut für Informatik*) of the University of Zurich, Switzerland. Part of my research was funded by the Swiss National Science Foundation who provided grants for the projects *Attempto – Controlled English for Requirements Specifications* (no. 20-47151.96 and no. 20-53806.98) and *Practical Applications of Attempto Controlled English* (no. 20-61932.00). These projects were lead by Dr. Norbert E. Fuchs. Further financial support was granted by the Department of Information Technology of the University of Zurich and by the *STRATMAS* project of the Swedish National Defence College. I wish to express my thanks for all this financial support which allowed me to develop this thesis.

Moreover, I am grateful to many persons who contributed to this thesis with their scientific, practical or personal help.

First of all I want to thank my supervisor Dr. Norbert E. Fuchs for his constant support and for many open, critical and fruitful discussions. Norbert showed me that having the courage to investigate questions from a non-standard perspective often paves the trail for new and interesting insights. I am also grateful to Norbert for creating a pleasant, flexible and creative working atmosphere.

I would also like to express my gratitude to Prof. Dr. Michael Hess, who heads the *Institute of Computational Linguistics* at the University of Zurich, for his offer to act as the second examiner of this thesis. He followed my research with interest and pointed me to some critical questions the investigation of which lead to a broader foundation of my thesis and to a clarification of many important concepts.

During my research I was part of the Requirements Engineering Research Group lead by Prof. Dr. Martin Glinz. I wish to thank Martin Glinz for offering excellent working conditions and for his personal support during my research. I would also like to thank his current and former group members for creating a friendly and cooperative atmosphere.

Among the many colleagues that followed my work I am particularly grateful to Rolf Schwitter for many discussions, ideas and technical support in the earlier stages of my thesis, and to Stefan Höfler for his motivation, his critical questions and his personal and practical assistance in the final stages of my thesis. I would also like to thank our office-mate Arun Mukhija in particular for his patience and tolerance with which he accepted the extensive discussions of Norbert, Stefan and myself in our office. Furthermore, I am grateful to Denis Antonioli, Johannes Ryser, Martin Volk and Yong Xia for following the development of my thesis with interest, motivation and many practical hints.

Moreover, I wish to thank the secretaries and the system administrators of the Department of Information Technology who were always very helpful concerning minor and major administrative and technical difficulties.

I would also like my friends to know that the rare free time that I spent with them on the run, in the mountains, at the crags, or with a glass of wine gave me the necessary distraction and cleared my mind so that I was able to continue studying complex problems.

Furthermore, I wish to give special thanks to my parents whom I always could rely on. They taught me the confidence and endurance important to finish this thesis.

Finally and most importantly, constant indispensable encouragement was given to me by Uwe Maaz to whom I express my special gratitude. His critical questions often brought me back to the focus of my work when I was in danger of getting lost in thousands of details. His patience, concern and motivation constantly accompanied me during my research.

Zurich, October 2003

Uta Schwertel

Abstract

The semantics of natural language plurals poses a number of intricate problems – both from a formal and a computational perspective. In this thesis I investigate problems of representing, disambiguating and reasoning with plurals from a computational perspective. The work defines a computationally suitable representation for important plural constructions, proposes a tractable resolution algorithm for semantic plural ambiguities, and integrates an automatic reasoning component for plurals.

My solution combines insights from formal semantics, computational linguistics and automated theorem proving and is based on the following main ideas. Whereas many existing approaches to plural semantics work on a model-theoretic basis using higher-order representation languages I propose a proof-theoretic approach to plural semantics based on a flat first-order semantic representation language thus showing that a trade-off between expressive power and logical tractability can be found. The problem of automatic disambiguation of plurals is tackled by a deliberate decision to drastically reduce recourse to contextual knowledge for disambiguation but rely instead on structurally available and thus computationally manageable information. A further central aspect of the solution lies in carefully drawing the borderline between real ambiguity and mere indeterminacy in the interpretation of plural noun phrases. As a practical result of my computational proof-theoretic approach to plural semantics I can use my methods to perform *automated* reasoning with plurals by applying advanced first-order theorem provers and model-generators available off-the shelf.

The results are prototypically implemented within the two logic-oriented natural language understanding applications DRoPs and Attempto. DRoPs provides an automatic plural disambiguation component for uncontrolled natural language whereas Attempto works with a constructive disambiguation strategy for controlled natural language. Both systems provide tools for the automated analysis of technical texts allowing users for example to automatically detect inconsistencies, to perform question answering, to check whether a conjecture follows from a text or to find equivalences and redundancies.

Zusammenfassung

Die Beschreibung der Semantik natürlichsprachlicher Pluralkonstruktionen wirft sowohl aus theoretischer als auch aus anwendungsorientierter Sicht komplexe Probleme auf. In meiner Dissertation untersuche ich Probleme der Repräsentation, der Desambiguierung und der logischen Inferenz mit Pluralen aus einer computersemantischen Perspektive. Die Arbeit definiert eine geeignete Repräsentation für die automatische semantische Verarbeitung von wichtigen Pluralkonstruktionen, entwickelt einen Algorithmus zur automatischen Auflösung semantischer Pluralambiguitäten, und integriert eine Komponente für die automatische Deduktion mit Pluralen.

Meine Lösung kombiniert Einsichten aus der formalen Semantik, der Computerlinguistik, und des automatischen Theorembeweisens und basiert auf den folgenden Grundideen. Wohingegen viele existierende Ansätze zur Pluralsemantik auf einer modell-theoretischen Basis unter Verwendung höherstufiger Repräsentationssprachen arbeiten, schlage ich einen beweistheoretischen Ansatz für die Pluralsemantik vor, der auf flachen, erststufigen Repräsentationen beruht. Dies zeigt gleichzeitig, dass ein Kompromiss zwischen Ausdrucksstärke und logischer Praktikabilität gefunden werden kann. Ich behandle das Problem der automatischen Desambiguierung von Pluralen, indem ich bewusst die Verwendung von kontextuellen Faktoren für die Desambiguierung von Pluralen drastisch reduziere und statt dessen strukturelle, und daher praktisch handhabbare Informationen verwende. Ein weiterer zentraler Aspekt der Lösung liegt darin, dass eine sorgfältige Unterscheidung zwischen echter Ambiguität und bloßer Unbestimmtheit bei der Interpretation von Pluralnominalphrasen gezogen wird. Ein praktisches Ergebnis meines computersemantischen beweistheoretischen Ansatzes zur Beschreibung der Pluralsemantik besteht darin, dass ich bereits weit entwickelte, frei verfügbare erststufige Theorembeweiser und Modellgeneratoren anwenden kann, um automatische Deduktion mit Pluralen zu realisieren.

Die Resultate habe ich prototypisch implementiert in zwei logik-basierten Systemen zur Simulation von Sprachverstehen: DRoPs und Attempto. DRoPs bietet eine Komponente zur automatischen Desambiguierung von Pluralen für unkontrollierte natürliche Sprache, während Attempto mit einer konstruktiven Desambiguierungsstrategie für kontrollierte natürliche Sprache arbeitet. Beide Systeme stellen Werkzeuge für die automatische Analyse von technischen Texten zur Verfügung. Die Werkzeuge erlauben den Benutzern beispielsweise, dass logische Widersprüche in einem Text vom System automatisch entdeckt werden, dass Fragen zu einem Text beantwortet werden können, dass geprüft werden kann, ob ein Text aus einem anderen Text folgt, oder dass Äquivalenzen und Redundanzen im Text gefunden werden.

Contents

List of Tables

List of Figures

1 Introduction

1.1 Computational Proof-Theoretic Semantics for Plurals

The main goal of this thesis is to develop computationally suitable techniques to describe and process the semantics of natural language plurals and to integrate these techniques into practical natural language understanding applications.

In general, semantic processing of natural language consists of systematically assigning natural language sentences logical representations that approximate the intuitively acceptable readings of the sentences in a formal way. All and only those representations should be generated that correspond to the natural readings of the sentence. The availability of logical representations allows us then to reconstruct intuitive reasoning processes in a formal way. For example, entailment relations or inconsistencies between sentences can be modelled by applying well-defined proof-procedures on formal representations. From a natural language processing perspective, we additionally want that the semantic component can be efficiently implemented, is robust, and is general and reusable for different applications. These two perspectives on semantic processing – formal and computational semantics – do often conflict and a trade-off between linguistic adequacy on the one hand and computational tractability on the other hand has to be found.

A solution for the problem of plural semantics will be proposed in this thesis. In particular, I develop logical representations that on the one hand approximate the intuitively acceptable readings of plural sentences and on the other hand are suitable for computational purposes. For plurals this task turns out to be particularly difficult for several reasons. First, in many cases our intuitions about the readings of plural sentences are not at all clear and it is accordingly difficult to define an adequate semantic representation. Furthermore, plurals are highly ambiguous and therefore lead to a proliferation of readings, making the computational treatment very hard. A major task will therefore be to keep the ambiguities, i.e. the number of postulated readings, within reasonable bounds and yet be able to explain inferences triggered by the occurrence of plurals. Moreover, many existing formal semantic theories argue for the introduction of higher-order languages to describe the semantics of plurals adequately. For computational reasons, however, a first-order language has to be preferred and is therefore proposed in this thesis.

As a practical result my thesis proposes a computational proof-theoretic approach to plural semantics that is applied within the two logic-oriented natural language understanding applications DRoPs and Attempto. These applications provide techniques for a logical analysis of English technical texts which describe a problem domain very precisely. Examples are natural language software specifications or medical documentation texts. A logical analysis of these texts allows users to detect inconsistencies, to check whether a sentence can be logically deduced from the text, to answer queries, to detect logical equivalences, to identify redundancies or incompletenesses, and so on. The analysis consists of two steps: an English text is automatically translated into a formal first-order semantic representation which is then the basis for the logical analysis of the text. The logical analysis consists predominantly of logical deduction which is automatically carried out by extended and modified versions of the existing automated theorem provers Otter and Satchmo. Finally, feedback about the results of the logical deduction is given back to the user again in natural language.

I have divided the development of a computational plural semantics for the intended applications into the following subtasks:

- delineating the plural constructions frequently found in English technical texts and describing their intuitive interpretation,
- investigating existing formal and computational approaches to plural semantics,
- defining a suitable semantic representation language that is linguistically and computationally adequate for the intended applications,
- developing methods to cope with the combinatorial explosion of ambiguities triggered by plural constructions,
- automatically constructing the semantic representations from natural language sentences,
- integrating automatic reasoning techniques for plurals,
- using the techniques in practical natural language understanding applications.

The wide range of the tasks did not allow me to investigate all constructions and problems related to plurals in adequate detail. Yet, my proposal is designed to be flexible enough to smoothly integrate new phenomena.

Although much remains to be done a main achievement of this thesis lies in combining methods from formal semantics, computational linguistics and automated reasoning to propose a computationally oriented approach to plural semantics that can be used in practical applications. I am not aware of other approaches that integrate a corresponding wide range of requirements and that are worked out and practically tested in similar detail.

1.2 Main Theses

The investigation of the above tasks has led to the following main underlying problems, theses and practical results of my work.

The Combinatorial Explosion Puzzle

Semantic theories predict that sentences containing plural noun phrases are highly ambiguous. On closer inspection the sentences have many more interpretations than pre-theoretically realized. Yet, human beings appear to be able to deal with these sentences with little effort if they care to disambiguate at all. This "Combinatorial Explosion Puzzle" (Poesio 1996) is a substantial problem for developing realistic natural language processing systems.

The Problematic Role of Context for Disambiguation

When humans process language it is commonly argued that they use context as a source of information to reduce or completely eliminate ambiguity or vagueness in the interpretation of an utterance. 'Context', however, is an extremely difficult concept (cf. Hirst 2000). Context can include "just about anything in the circumstances of the utterance, and just about anything in the participants' knowledge or prior or current experience" (Hirst 2000, p. 7), or, in other words, context "can be the whole world in relation to an utterance act" (Pinkal 1985, p. 36).

My contention is that although context determines human language processing context can not be reasonably used for *automatic* disambiguation of natural language. The reason is that "a reliable large-scale modelling of contextual disambiguation factors is far beyond the capabilities of current computational systems" (Lappin and Leass 1994, p. 559).

Plural Disambiguation Without Contextual Knowledge is Feasible

To automatically disambiguate plural ambiguities I therefore established an automatic disambiguation algorithm that is largely independent of contextual factors but predominantly relies on tractable structural properties of the sentence plus a very limited amount of lexical knowledge. My findings suggest that with this strategy good results for the automatic disambiguation of a non-trivial fragment of plural sentences occurring in technical texts can be achieved.

Plurals are Not Always Ambiguous But Often Indeterminate

A basic aspect of my approach is the distinction between ambiguity and indeterminacy. I argue that many sentences that formal theories of semantics claim to be ambiguous are in fact indeterminate. More precisely, I suggest that the collective reading can be very indeterminate concerning what really happened between the individuals involved. Most often speakers do not intend to convey this detailed information at all. I argue that the semantics assigned to a sentence should not be more specific than the language itself and not promote descriptions of what could have happened between the individuals as alternative interpretations of the sentence. Indeterminacy is a powerful tool to express just the information that is appropriate or known in a given situation, and allows the speaker not to be more precise than necessary.

A First-Order Proof-Theoretic Approach to Plural Semantics

Many existing approaches to plural semantics work on a model-theoretic basis using higher-order representation languages. In contrast, I propose a proof-theoretic approach to plural semantics based on a flat first-order semantic representation language. I thus show that a trade-off between expressive power and logical tractability can be found. As a practical result, I can use my approach to perform *automatic* reasoning with plurals by applying advanced first-order theorem provers and model-generators available off-the shelf. My approach thus also demonstrates that reusability is an important criterion for modern semantic theories.

Practical Applications

I show how my approach is prototypically implemented and used within two natural language understanding applications: the DRoPs (Disambiguating and Reasoning with Plurals) system and the controlled natural language application Attempto. Both systems work with the same underlying semantic representation and are equipped with basically the same reasoning component, using the same core of auxiliary logical axioms. Only the disambiguation components are different. The DRoPs disambiguation component is intended for the semantic processing of full natural language technical texts and generates a set of plausible readings ordered according to plausibility. The disambiguation component of the Attempto system is used for the *controlled* natural language Attempto Controlled English (ACE) and generates just one reading that is uniquely predictable from the construction and interpretation rules defining the controlled natural language ACE.

1.3 Overview

The thesis is structured as follows.

Chapter 2 introduces the problem of ambiguity in natural language and the consequences for natural language processing. Different types of ambiguities are introduced and a brief overview of the semantic ambiguities caused by plural noun phrases is given. Ambiguity is distinguished from vagueness and indeterminacy – a distinction that proves relevant for this thesis. Two basic approaches to ambiguity processing in natural language are introduced: the *Generate and Test* and the *Underspecification* approach. This thesis combines insights from both approaches to process plural ambiguities.

Chapter 3 explains important existing techniques within formal semantics to represent plural ambiguities. Plural ambiguities occur in connection with plural noun phrases. In section 3.2 I will therefore give a brief overview of two influential approaches to formal noun phrase semantics: *Generalized Quantifier Theory* and *Discourse Related Approaches*. The overview also establishes a common terminology for chapter 4. Readers familiar with the two approaches can safely skip this overview. In the rest of chapter 3 I will then present some influential and prototypical existing formal semantic approaches to the representation of plural noun phrases. These include the proposals of Scha (1981) and of Link (1983, 1991). Whereas Scha locates the ambiguity of the plurals within the noun phrase, Link attributes it to verb phrases. Scha's noun phrase centred approach suffers from an overgeneration of semantic readings and uses computationally impractical higher-order logical forms. Link's verb phrase centred approach can be formulated in a computationally more practical first-order language yet the problem of ambiguity explosion is not solved. A third approach by Verkuyl and van der

Does (1991) tries to give plural noun phrases just one semantically weak representation that encompasses the other readings. The approach, however, suffers from empirical shortcomings and the approach is also formulated in a computationally impractical language. Finally, there are approaches that attribute the ambiguity not to single elements but to global factors. For computational applications I argue this to be the most plausible strategy, however, the existing approaches are not formulated in a computationally practical way. In chapters 4 and 5 of this thesis I will therefore develop a computationally suitable formalization of this global strategy.

Chapter 4 develops a computationally suitable flat first-order semantic representation for plurals. In developing a semantic representation for plurals one has to deal with questions like where to locate the ambiguity, how many readings to assume, which underlying ontology to adopt and how to represent the readings. All questions have to be answered with a special emphasis on the suitability for computational applications. Section 4.2 deals with a set of requirements that a representation of plural ambiguities must fulfil to fit the applications pursued in this thesis. The section includes a brief overview of different perspectives on semantic processing, viz. formal vs. computational semantics and truth-conditional vs. proof-theoretic semantics. In my thesis, I propose a proof-theoretic computational approach to plural semantics based on a flat first-order semantic representation language. Section 4.3 will then deal with the question how many readings are assumed and I will argue that it is sufficient to assume just collective and distributive, but no cumulative readings. The collective reading is assumed to be indeterminate with respect to the concrete realization of the constellations. Section 4.4 will then introduce the formal setting: as a representation language I propose a flat first-order variant of discourse representation theory, and as the underlying ontology I assume lattices that include plural entities as ordinary objects of the domain. In section 4.5 the flat first-order representations for a number of important plural phenomena will be introduced, also dealing with the question where to locate the ambiguity. Furthermore, I introduce additional first-order axioms that are necessary to describe inferences triggered by plurals.

Chapter 5 develops a plural disambiguation algorithm that is based on the interaction of a set of computationally manageable rules. The algorithm uses information available in the text plus lexical information that can be automatically extracted. I will show a prototypical implementation of the algorithm. The algorithm will offer the best reading or, alternatively, a selection of several preferred readings ordered according to preference. The advantages of the algorithm compared to other approaches are that the influence of different disambiguation factors can be integrated, that the algorithm offers an ordered set of plausible readings (with the option to choose but the best reading), that the formulation of the rules does not rely on world-knowledge or context and, finally, that the integration of new disambiguation sources (e.g. more fine-grained semantics of verbs) can be neatly added. Section 5.2 first gives an overview over existing approaches to plural and scope disambiguation from which I will borrow some ideas and techniques. In section 5.3 I will summarize the "accessible" information sources that my approach uses for the automatic disambiguation of collective/distributive ambiguities plus some resulting scope ambiguities. I will only briefly address scope-ambiguities but will not include a complete algorithm. For scope ambiguities there is much more literature than for plural ambiguities. I will, however, show in my approach how the two related problems can be integrated. Section 5.4 introduces the basic algorithm for the automatic disambiguation of plural ambiguities based on the disambiguation information elaborated in 5.3. This algorithm is currently prototypically implemented within my system DRoPs (Disambiguating and Reasoning with Plurals). I will show the data structures used for the implementation and explain how parts of the algorithm are implemented on the basis of the data structures.

Natural language understanding in practical applications requires an appropriate reasoning component. In chapter 5 I show how the DRoPs disambiguation algorithm is complemented by off-the shelf theorem-proving techniques which allow the system to derive further information from disambiguated structures, to detect inconsistencies, redundancies, equivalences, and so on. In section 6.2 I will give an overview of the architecture and the basic working of the DRoPs reasoning component which is based on the reasoner RACE developed for the Attempto project as described in chapter 7. Section 6.3 will explain the extension of the reasoner with auxiliary first-order axioms that are necessary for automatic reasoning with plurals. In section 6.4 practical limitations of the current implementation are addressed and promising topics of further research are presented.

In chapter 7 I show how techniques developed for the DRoPs system are applied and modified to process plurals within the natural-language understanding application Attempto (Schwitter 1998, Fuchs, Schwertel and Schwitter 1999a, Schwertel, Fuchs and Höfler 2003). The core of the Attempto system is the controlled natural language Attempto Controlled English (ACE). Since ACE is not a full, but a *controlled* natural language the Attempto system has partly different requirements than the DRoPs system. In particular, disambiguation in ACE follows different principles than disambiguation of full natural language in DRoPs. Whereas the DRoPs system generates a hierarchy of plausible readings, the Attempto system generates for each sentence just one reading that is uniquely predictable from a set of construction and interpretation rules. This deterministic approach is a result of the constructive disambiguation strategy chosen for the Attempto system. In section 7.2 I will give an overview of the philosophy of ACE. I will explain the motivation for developing the language ACE, explain ACE's constructive disambiguation strategy, and give examples for important construction and interpretation rules that concern the singular fragment of ACE. Section 7.3 describes the main ideas of extending ACE with plural constructions. The section shows how syntactic and semantic plural ambiguities can be handled using the constructive disambiguation approach of ACE. Section 7.4 will give a brief overview of the architecture of the Attempto system. Finally, in section 7.5 I will conclude, address possible extensions and improvements and give examples for other currently investigated applications of ACE which show the flexibility and the usefulness of the approach pursued in the Attempto project.

I will conclude the thesis by summarizing the main results and addressing open problems and issues for further research.

2 Plurals and Ambiguity

[…] natural language processing machines have great trouble dealing with ambiguous input, and it is probably fair to say that ambiguity is one of the most daunting problems for automatic analysis of natural language (van Deemter et. al. 1996, p. xvi).

2.1 Overview

This chapter introduces the problem of ambiguity in natural language and the consequences for natural language processing. A brief overview of the ambiguities caused by plural noun phrases (NPs) is given which shows that plurals can cause a combinatorial explosion of ambiguities. For my further investigation I will distinguish the concept ‘ambiguity’ from the related concepts ‘vagueness’ and ‘indeterminacy’ – a distinction that proves relevant for the sequel. Finally, basic approaches to ambiguity processing in natural language are introduced and their problems are discussed.

More concretely, section 2.2 explains different types of ambiguities that occur in natural language (e.g. lexical, syntactic, semantic ambiguity). Plurals are a frequent source of semantic ambiguities. Therefore the current investigation is mainly restricted to semantic ambiguities. In section 2.3 important types of plural constructions are briefly introduced and the resulting ambiguities are explained using simple examples. Furthermore, a common terminology for naming plural ambiguities is established. The section shows in particular the problem of combinatorial explosion of readings resulting from the use of plurals. Section 2.4 gives a precise definition of the concept ‘ambiguity’ and contrasts it to the concepts ‘vagueness’ and ‘indeterminacy’. These concepts are important for the treatment of plurals since it is often not clear whether plural constructions are really ambiguous (i.e. generate two different semantic representations) or whether the constructions are only indeterminate (i.e. generate just one less specific semantic representation). In my study I will argue that many cases of plural constructions can be analysed as indeterminate thus reducing the number of postulated semantic representations which considerably facilitates automatic processing of the sentences. Section 2.5 explains requirements that a semantic ambiguity processing component for natural language processing purposes must fulfil. I will briefly address the main ideas of the *Generate and Test* approach and the *Underspecification* approach that are important in this respect, and I will list some common problems.

2.2 Types of Ambiguity

Ambiguity in a pre-theoretic sense means uncertainty or unclearness of meaning. Ambiguity is noticed when an expression has two or more clearly distinct meanings and the context does not make clear which meaning is intended. There are several types of ambiguity depending on what causes the ambiguity. I will briefly explain lexical, structural and semantic ambiguities. In Hirst (1987) a detailed overview with many examples is given.

Lexical Ambiguity. As for *lexical ambiguity* two subtypes are distinguished. One type of lexical ambiguity occurs when a word is associated with different senses. If the senses are related to one another we speak of polysemy, if the senses are unrelated we speak of homonymy. An example for a polysemous word is *open* that has many senses concerning unfolding, expanding, revealing etc. An example for a homonymous word is *bank* that may denote, among other things, a financial institution or a sloping land beside a river. A second type of lexical ambiguity occurs when the same written realization of a word belongs to different syntactic categories. This type is often called *categorial ambiguity*. For example the string *can* may be used as a modal auxiliary or as a noun.

Structural Ambiguity. *Structural ambiguities* occur when a sentence or a phrase can be assigned more than one syntactic parse. Famous are so-called attachment ambiguities like

(1) I saw the girl with the telescope.

where the prepositional phrase (PP) *with the telescope* can modify either the noun phrase *the girl* or the verb phrase *saw the girl*. In the first case the sentence means that the girl had a telescope with her, in the second case the sentence means that the instrument of seeing the girl was a telescope. Another example for a structurally ambiguous expression is the noun phrase *Tibetan history teacher* that can be given different underlying structures: *[Tibetan history] teacher* or *Tibetan [history teacher]*. Many English sentences have more than one parse, however, there is usually a unique preferred parse if the sentence is uttered in a certain context. If the following sentence (taken from Hirst 1987, p. 9)

(2) They're cooking apples.

is uttered in a kitchen where people are performing cooking activities we prefer to analyse *cooking* as the head of the verb phrase whereas when the same sentence is uttered on a vegetable market we prefer to analyse *cooking* as an adjective modifying the noun *apples*. Hirst (1987, pp. 131) gives an extensive list of other possible structural ambiguities.

Semantic Ambiguity. A further type of ambiguity occurs when a sentence has a unique syntactic structure but this structure does not uniquely determine the logical form of the sentence. I will call this type of ambiguity *semantic ambiguity*. Depending on the reading a semantically ambiguous sentence is assigned different logical meaning representations. Note that the term "semantic ambiguity" is often used in a more general sense and also includes lexical ambiguity. Typical examples for semantic ambiguities are *scope ambiguities* as in the following sentence.

(3) A device controls every pump.

Though the surface syntactic structure is unequivocal the sentence has two interpretations. Either the same device controls every pump in which case *a device* is said to have wide scope in the logical form with respect to *every pump*, or every pump is controlled by a possible different device, in which case *every pump* has wide scope with respect to *a device*. Other examples for scope inducing elements in natural language are negation (*not*, *no*), modals (*can*, *must*) or adverbs of quantification (*always*, *twice*). As with lexical and structural ambiguities, in scopally ambiguous sentences usually one of the interpretations is clearly preferred. The sentence

(4) She knows a solution to every problem. (Poesio 1994, p. 5)

prefers an interpretation with a different solution to every problem, that is *every problem* has wide scope with respect to *a solution*. In other cases like (3) above the preference for one or the other scoping is weaker. Necessary knowledge is missing that would lead to a preferred interpretation.

Another type of semantic ambiguities are the so-called *collective/distributive ambiguities* that result from the use of plural noun phrases.

(5) a. John and Mary buy a computer.
b. Two men lifted a table.

Sentence (5)a can mean that John buys a computer and Mary buys a (possibly) different computer (*distributive* interpretation) or that they buy a computer together (*collective* interpretation). Similarly, in (5)b the two men each may have lifted a (possibly different) table or they may have lifted a table together.

In the following sections we will focus on semantic ambiguities caused by plurals. We will see that there is a combinatorial explosion of ambiguities when a sentence contains more than one plural noun phrase. A further complicating factor is that distributive interpretations also induce scope ambiguities leading to a further explosion of ambiguities.

After looking at a number of semantically ambiguous plural sentence in section 2.3 I will give a more precise definition of the concept ‘ambiguity’ in section 2.4 and contrast the concept ambiguity to the concepts ‘vagueness’ and ‘indeterminacy’. I will then address in section 2.5 which strategies are chosen by natural language processing systems to deal with the problem of ambiguity.

2.3 Plural Constructions and Ambiguities – An Overview

2.3.1 Collectivity and Distributivity

2.3.1.1 Collective and Distributive Readings

One of the most fundamental issues in the semantics of plurals is that sentences with plural noun phrases like

(6) Two men lift a table.

allow for distinct readings. On the so-called *distributive reading* the three men each lifted a (possibly different) table, i.e. there are two table-lifting events. The predicate *lift a table* applies to each of the two men. The distributive reading can be explicitly triggered by words like *each* as exemplified in (7).

(7) Two men each lift a table.

On the so-called *collective reading* the men acted together to lift a table, i.e. there is just one table that the two men lift together in one event. In this case the predicate *lift a table* does not apply to individuals, but to the group consisting of two men. The collective reading can be suggested by adverbs like *together* as in (8).

(8) Two men lift a table together.

It is important to note that the collective and the distributive readings may not be identified with concepts like collaboration or a lack thereof as the example above might suggest. Collective and distributive readings not only occur in subject position of agentive verbs like *lift*, *carry*, *write*, *buy* but also together with other verb types that do not suggest collaboration, e.g. metric verbs as in (9)a, or predicative constructions with metric adjectives as in (9)b.

(9) a. The two books cost 20 dollars.
b. The books are heavy/expensive.

There are predicates that are not compatible with one of the two readings. Predicates like *sneeze*, *sleep*, *die* make sense only for a distributive reading since the properties cannot sensibly apply to whole groups.

(10) a. The students are asleep.
b. Two students sneezed.

On the other hand there are predicates that cannot be true of single individuals, that is they do not allow for fully distributive readings. Examples are predicates like *meet*, *agree*, *marry* that are sometimes called covert reciprocals (Langendoen 1978) since the reciprocal element (*each other*) is not made explicit. Other complex verb phrases like *be alike*, *be numerous*, *be colleagues*, *be friends*, or verbs of configuration like *assemble*, *gather*, *disperse* also show this behaviour (see Dowty 1986 and Gillon 1996 for more examples).

(11) a. The students gathered in the classroom.
b. The students dispersed.
c. The students are numerous.

Strict classifications of verbs according to one or the other type are, however, not unproblematic especially since "collective" verbs like *meet* or *gather* also allow for distribution, however not to the level of individuals as is exemplified by

(12) The men and the women met.

Sentence (12) can mean that there is one big meeting of the men and the women together, or that there is a separate meeting of the men and another meeting of the women. A further form of distribution of "collective" predicates can be seen in sentences like

(13) Three committees gathered.

where a so-called collective noun like *committee* is used that – though in the singular – denotes a group object. Thus the predicate *gather* can distribute to each committee meaning that each committee met separately, or the predicate can alternatively apply to all committees together meaning that a group consisting of three committees met. Furthermore, "collective predicates" do exhibit so called distributive sub-entailments (Dowty 1986). For example, in (11)a although the predicate *gather* does not distribute to the individual members of a group, it entails that each of the students must come to the classroom at a certain time that overlaps with the entering of other students. Thus *gather* distributively entails some properties of the members of its group, although the predicate itself is only true of the whole group. Most "collective" verbs do have distributive subentailments of different types – with the exception of a small class of complex predicates that include e.g. *be numerous*, *be a couple*, *be a large group*. Dowty (1986, p. 101) calls this class "pure cardinality predicates".

In the examples above we have only considered plural noun phrases in subject position. However, collective/distributive ambiguities can be observed in any argument position of verbs. These positions include, beside the subject, the direct and indirect object of verbs, and also plural noun phrases in prepositional adjuncts. For the direct object of transitive verbs clear examples of ambiguity are more difficult to argue for. The reason is that many verbs show distributive subentailments in the direct object position which blur the distinction between collective and distributive readings. Take for example the sentence

(14) John lifted three tables.

If John lifted three tables as a whole he also had to lift each of the three tables. Thus the collective reading entails the distributive reading. However, there are other examples that show that the direct object position can be ambiguous:

(15) a. She summarized the proposals . (Dowty 1986, p. 107)
b. John juggled with six plates. (Link 1998a, p. 32)
c. Samantha quickly polished the boots. (Parsons 1990, p. 46)

d. Fred could not lift three tables. (Carpenter 1995, p. 7)

In the collective reading (15)a means that the main ideas of all the proposals where summarized without necessarily summarizing each proposal. The distributive reading on the other hand means, that each of the proposals was summarized separately which does not necessarily imply that there is also a summary of all the proposals as a whole. In (15)b it makes a difference whether John juggles with six plates at the same time or one after the other. The example shows, that especially when the temporal relations between the various events expressed by a plural sentence are considered the collective/distributive distinction in the direct object position becomes relevant. This can also be seen in sentence (15)c which means in the distributive reading that there are many polishings, one per boot, and each polishing is said to be quick. In the collective reading the whole polishing of the group of boots is said to be is quick. In this case neither of the readings implies the other reading; each individual polishing might be quick without the polishing of the boots being quick and vice versa. As a final example, sentence (15)d can be true on a collective but false on a distributive reading of the object; that is, Fred might be able to lift three tables one after the other, but not the tables as a whole. Gillon (1996) lists other verbs that can show collective/distributive ambiguities in direct object position, e.g. *to play against* or *to visit* and others. Note furthermore, that there are verbs that are "collective" in their second argument, e.g. *collect*, *enumerate* etc.

Collective/distributive ambiguities also occur with respect to indirect objects as in

(16) John told the children a story.

which can mean that John told each of the children a possibly different story, or that he told them together a single story.

Furthermore plural noun phrases within prepositional adjuncts of the verb can exhibit a collective/distributive ambiguity

(17) a. John went to his favourite restaurant with two friends.
b. John met his sister on two summits.
c. John drove through the redwoods. (Gillon 1996)

Though preferring a collective reading (17)a can also have a distributive reading where John took each of his friends separately to the restaurant. Sentence (17)b prefers a distributive reading since it is very unlikely to be on two summits at the same time. Sentence (17)c has been given by Gillon as an example for the collective/distributive ambiguity of plural noun phrases as arguments of prepositions.

It is often neglected in the literature on plural semantics that collective/distributive ambiguities do not only occur in connection with verbs but also within complex noun phrases where two noun phrases co-occur. This point has been emphasized by Gillon (1996, pp. 453).

(18) a. the children of Betty and John
b. the weight of these two suitcases

The complex noun phrase in (18)a can denote the common descendants of Betty and John or it can denote a set of children one part of which are descendants of Betty and the other part of which are descendants of John. In (18)b *these two suitcases* can have a collective interpretation in which case the noun phrase refers to the weight of both suitcases taken together. A different interpretation arises if *these two suitcases* has a wide scope distributive interpretation in which case the weight of each of the two suitcases is considered separately. The ambiguity also occurs with prepositional phrases the head of which is not *of* as has been noted by Quirk et al. (1985, ch. 13.69-70) and discussed by Gillon (1996, pp. 453). Further complexities occur in sentences like

(19) The authors of the best conference paper wrote a book in 2001.

where *the authors* as the head of the complex subject noun phrase can be understood collectively with respect to its complement *the best conference paper*, but can be understood distributively with respect to the verb phrase *wrote a book in 2001*. The same conflicts occur with relative clauses complements as in

(20) The two men who repaired the machine got a bonus.

The sentence can mean that the men together repaired the machine but that each of them got a separate bonus.

Collective/distributive ambiguities within noun phrases can also occur with certain types of adjectives as in

(21) a. the heavy suitcases
b. the expensive books

The above examples showed that every argument position associated with a verb can get a collective or a distributive reading. Furthermore, argument positions associated with a noun are also liable to collective/distributive ambiguities. This already indicates that the occurrence of plurals leads to a massive increase of ambiguities.

2.3.1.2 Multiple Plurals and Scope Alternations

In the above examples we have mostly considered the effects of the collective/distributive ambiguity for a single occurrence of a plural noun phrase in a sentence. Problems increase if we consider the interaction of more than one plural noun phrase in a sentence. Take for example a sentence with a two-place predicate containing two plural noun phrases as in

(22) Two men lifted three tables.

If we assume that both noun phrases can have a distributive and a collective reading we have to assume $2 \times 2 = 4$ interpretations for this simple two place sentence. If both noun phrases are read collectively there are 3 tables involved, if the subject noun phrase is read distributively there are up to six tables involved. Even worse, since the distributive interpretation of plural noun phrases introduces a quantificational element (universal quantification over individual members) plural noun phrases enter into scope alternations – we have seen this already above, for example in (18)b. Since in principle the object noun phrase can take scope over the subject noun phrase the number of possible readings has to be multiplied by 2. Since for a doubly collective reading the relative scope of the two noun phrases makes no difference we end up with 7 possible readings for a simple sentence like (22) which shows a massive increase of ambiguities. For examples like (22) it has been argued that it is very unlikely that the direct object is interpreted as having wide scope over the subject noun phrase (see also chapter 5 of this thesis). There are, however, other examples where two plural noun phrases show scope alternations.

(23) a. The company sent two letters to 500 customers.
b. John told two stories to several children.
c. Five insurance associates gave a $25 donation to several charities.
(Roberts 1987, p. 113)

Sentence (23)a can mean that there are 500 customers and each of the customers receives two letters. In this case the distributively interpreted noun phrase *500 customers* does have wide scope over *two letters*. Similarly, in (23)b there can be several children each of which is told two stories that are possibly different for every child. The same holds for (23)c.

Further complicating quantificational effects of plurals occur in connection with negation or modals. Carpenter (1995, section 1.5) argues with examples of the following type

(24) a. No students passed the exam/gathered in the hall.
b. Three students didn't pass the exam.
c. Three students didn't gather in the hall.

Carpenter analyses *no* as a negative existential quantifier. It negates the existence of a group which itself can be read collectively or distributively. In (24)b there are scope ambiguities between the negation and the noun phrase *three students*. Independent of its scope the plural noun phrase *three students* can be read collectively or distributively. Assuming a distributive reading and considering scope alternations we arrive at 3 distributive readings for (24)b:

(25) a. There are three students each of which didn't pass the exam.
b. It's not the case that there are three students each of which passed the exam.
c. There are three students and it's not the case that each of them passed the exam.

The examples show that taking collective/distributive ambiguities and additionally scope ambiguities into account the number of possible readings explodes further.

2.3.2 Intermediate Readings

Several authors have argued that sentences like

(26) a. Half a million children gathered all over the country. (Link 1991)
b. Hammerstein, Rodgers and Hart are composers.

They have written musicals. (Gillon 1987)

admit a reading beyond the collective and the distributive reading. Depending on the author the reading is called neutral, mixed, participatory, intermediate or partitional reading. The neutral reading was first mentioned by Scha (1981) and further discussed by Gillon (1987), Verkuyl and van der Does (1996), van der Does (1993), Schwarzschild (1992, 1994). Although they formalize the readings differently the main idea is that the group of objects introduced by the plural subject may be broken down into subgroups such that the predicate applies to these subgroups. For example, (26)a can be true if the children didn't meet all together, but there are subgroups of the children each of which gathered separately. In a way, what has been called a collective reading above could be seen as a special case of the mixed reading (if there is just one subgroup). Gillon argued that the second sentence of (26)b can be used truthfully if *they* refers to the group of Rodgers, Hart and Hammerstein and Rodgers and Hart collaborated in writing operas and so did Rodgers and Hammerstein, but none of them wrote operas on his own nor did they write an opera together. Proposals to treat the neutral reading as an extra reading differ in that they put different restrictions on possible partitionings of the group: partitioning, minimal covering, "pseudo-partitioning" (see Verkuyl 1999, article 6 for an overview). In (26)b for example a partitioning is too strict since subgroups do overlap.

The question whether intermediate readings should constitute a separate reading in the formal representation is controversial. I will discuss this issue in sections 3.3 and 4.3.

2.3.3 Cumulative Readings

Scha (1981) noticed that sentences like

(27) 600 Dutch firms own 5000 American computers.

have an interpretation where there is a total of 600 Dutch firms and a total of 5000 American computers and each of the computers is owned by one or more Dutch firms and each of the Dutch firms owns one ore more computers. The actual realization of the owning relation is not further specified, only the amount is relevant. The computers could have been bought in arbitrary groups and the firms could form arbitrary unions to buy the computers – as long as all of the firms and all of the computers were involved. Scha assumed a separate reading for this interpretation and called it the *cumulative reading*. In the cumulative reading neither of the noun phrases has scope over the other noun phrase, that means the cardinalities of the two groups are determined independently. Cumulative readings can be generalized to the occurrence of more than two plural noun phrases as is exemplified with a ternary predicate in

(28) 600 Dutch firms donated 300 American computers to 200 schools. (Carpenter 1995)

Cumulative readings can also occur within complex noun phrases. As the following examples show it is hard determine whether there is a cumulative reading or a scoped reading:

(29) a. fifty students from five countries
b. 570 subjects from 9 administrations

c. five students with three cars
d. 16 words of 16 bits
e. 6 lines of 72 characters

Example (29)a prefers a cumulative reading or a wide-scope distributive reading of *five countries*, the same holds for (29)b. In (29)a world-knowledge rules out a wide-scope distributive reading of the head noun phrase *fifty students*. Example (29)c on the other hand prefers either a cumulative reading or a wide-scope distributive reading of five students whereas (29)d and (29)e both prefer a wide scope distributive reading of the head of the noun phrase.

Scha's examples for cumulative readings can be seen as generalizations of Langendoen's (1978) discussion of "elementary plural relational sentences". Langendoen discussed for example a sentence with two definite noun phrases

(30) The women released the prisoners.

As with (28) above in sentence (30) the two definite noun phrases refer independently of each other, that is, they do not have scope over each other. Langendoen investigated how the elements denoted by the noun phrases can participate in the relation expressed by the predicate to make the sentence true. Langendoen distinguishes between *weak* and *strong interpretations*. According to the strong interpretation each member of the subject must be related to each member of the object and vice versa – an unlikely interpretation for (30). On a weak interpretation it is sufficient that for every element of the subject there is an element of the object that stands in the corresponding relation and for every element of the object there is an element of the subject. Furthermore, Langendoen noticed that even this weak interpretation can be too strong because for (30) to be true it could be the case that some of the women acted together to release one or more of the prisoners which is why Langendoen introduced a *weak interpretation for subsets* where each member of the subject must be a part of an object that released one or more prisoners, and vice versa for the object denotation. But even this schema for subsets can be too strict since there are sentences where it need not be the case that all elements of the subject denotation participate directly as in

(31) The ten players scored three goals. (Carpenter 1995, p.10)

The sentence involves a collective action but it does not involve direct participation by everyone. Does this constitute yet another reading as Kamp and Reyle (1993) suggest with their *shared-responsibility reading* (see below in section 2.3.4)?

Cumulative readings are different from strictly collective readings.

(32) John and Mary own 3 houses.

In a strictly collective sense John and Mary together own a group of three houses and neither of them owns his or her own house. On a cumulative interpretation John might own one house, Mary another house, and John and Mary own together the third house. Yet, the question is whether this difference requires the assumption of two different semantic representation or whether the difference can be explained differently, for example by assuming a weak collective reading and subsuming both, strictly collective and cumulative readings, under this weaker reading.

2.3.4 Shared Responsibility and Generic Readings

For definite plural noun phrases Kamp and Reyle (1993) introduce yet another reading, the so-called *shared-responsibility reading* as exemplified in

(33) The guys in 5b have been cheating on the exam again.

Kamp and Reyle (1993, pp. 410) say that (33) can be true even if only a small number of the guys cheated, but the class as a whole is held responsible for the acts of some people. Kamp and Reyle say that the shared responsibility reading cannot be subsumed under the collective or the distributive reading. The shared-responsibility reading in Kamp's sense says that the predicate is true of a group of objects if the predicates holds of some members of the group, i.e. the predicate applies to some but not all individuals members. The assumption that shared-responsibility readings constitute a genuine reading that needs an extra representation in the logical form is controversial. Most authors argue that the shared-responsibility reading should be seen as a special case of the collective reading. I will also argue in this direction.

Kamp and Reyle also stress that their shared-responsibility reading should not be mixed up with so-called *generic readings* that they argue for in sentences like

(34) The children in this city thrive.

The sentence can be true if not all of the children thrive but only all "typical" or all "normal" cases. Again one can argue that generic readings should be subsumed under a collective reading in a wider sense. The topic of generic readings is a research field on its own and will not be dealt with in this investigation.

2.3.5 A Map of Readings

Assuming just collective and distributive readings for single noun phrases and additionally a cumulative reading for multiple plurals a sentence like

(35) Four men lifted three tables.

receives 8 different reading. Figure 1 on page 18 (adapted from Link 1991) summarizes the postulated readings and indicates the resulting problem of combinatorial explosion of readings.

2.3.6 Other Plural Constructions

Apart from the examples above I will look at a number of plural constructions that occur frequently in technical texts. Since I will discuss the phenomena in more detail in section 4.5 I will give only a brief overview here.

Cardinality Quantifiers. As we have seen in the above examples certain types of plural noun phrases can be read collectively. Clear cases are definite noun phrases (*the men*, *John and*

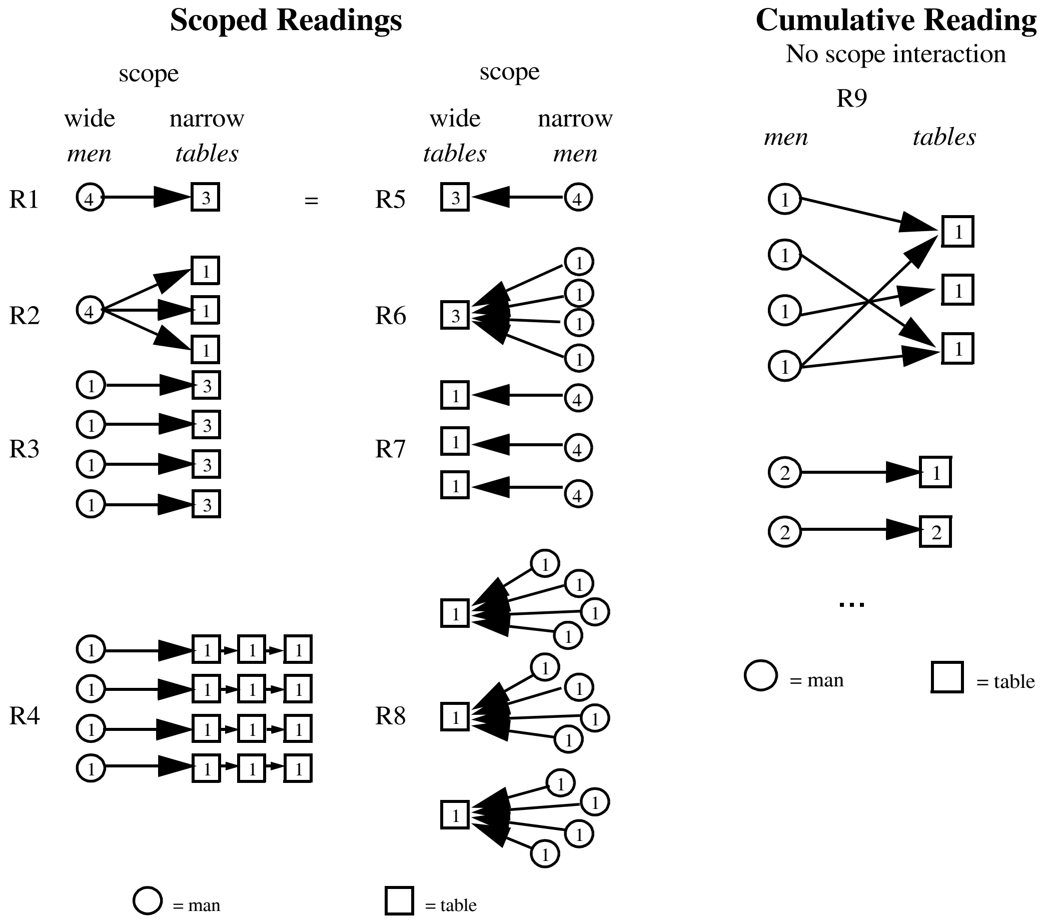


Figure 1 Eight Readings of '*Four men lift three tables*.'

Mary), indefinite noun phrases (*some men*) and certain cardinality noun phrases (*three men*). There is a set of other plural determiners that can also lead to a collective reading, e.g. *many*, *at least two*, *few*, *at most two*, *exactly two N*, but that usually prefer a distributive reading. Since these cardinality quantifiers occur in technical texts their semantics will be discussed.

Bare Plural Noun Phrases. Bare plurals are plural NPs without overt article. Sentences with bare plurals can have existential (*John owns old cars.*), universal (*Cars have four wheels.*) or generic readings (*Birds fly.*). A subclass of bare plurals are dependent plurals. Dependent plurals need not be understood as denoting groups. In the sentence

(36) All students own cars.

the bare plural NP *cars* is interpreted as "one or more" cars, not necessarily as "more than one" car. Dependent plurals can occur in the scope of other plural noun phrases:

(37) Several jobs place entries on a data queue.

Furthermore bare plurals occur in constructions with the copula *be*

(38) a. Five professors are women.

b. Some application-dependent objects are not journaled database files.

where the use of the bare plural is simply a matter of agreement. For example the property of being a woman applies only to individuals.

In my discussion I will mainly consider existential readings of bare plurals. The problem of generic readings will not be dealt with.

Coordination. We have seen above that coordination of definite and indefinite noun phrases can lead to distributive and collective ambiguities.

(39) a. John and Mary enter a card.
b. The boys and the girls bought a piano.
c. A man and some women buy a boat.

A further complicating factor concerns the level to which the predicate distributes. In example (39)b the predicate can distribute to each individual member, to the boys as a group and the girls as a group, or to the whole group of boys and girls taken together.

Enumeration. Like coordination different forms of enumeration can lead to ambiguities:

(40) The cards are installed in slots E01, E02 and E03.
(41) John presses button A, B and C.
(42) We have plotted two curves on the axes: two 1000-bit block error rates and two bit error rates.

Plural Anaphora. Plural anaphora like pronouns (*they*) or definite noun phrases (*the pumps*) refer to plural objects that are introduced in the previous context.

(43) John and Mary enter a Visacard. They are customers.
(44) The system comprises four pumps. Four devices supervise the pumps.

A particular problem with plural anaphora are so-called “split” antecedents, i.e. the antecedent of the plural anaphor has to be constructed from noun phrases that occur in different positions in the sentence:

(45) John meets Mary. They are in London.

The problem of plural anaphora resolution is a research topic on its own that will not form a particular focus within this work. For an introductory discussion see Kamp and Reyle (1993, pp. 426).

Floating Quantifiers and Event Modifiers. There is a certain set of words that play an important role in the disambiguation of plurals, among these are so-called floating quantifiers like *each*, *all* or *both*:

(46) a. John and Mary each lifted a table.
b. The customers all lifted a table.
c. John and Mary both lifted a table.

The function of the floated quantifiers is – similar to non-floated counterparts – to distribute over the group denoted by the subject term. Note that Dowty (1986) has noted that the effect of *all* is not identical to *each*. His hypothesis is that *all* distributes the so-called “subentailments” of verbs down to the individual participants.

Explicit modifiers like *together* in

(47) The customers lifted a table together.

imply a joint, however, not necessarily purely collective action (see for a discussion Lasersohn 1995, chapter 11). Thus (47) favours a collective reading but does not necessarily require it. Other examples show that – besides a collectivizing effect – *together* can also have spatial proximity or temporal simultaneity readings as in

(48) a. John and Mary sat together.
b. John and Mary stood up together.

Sentence (48)a indicates spatial proximity rather than a contrast between sitting individually versus sitting collectively.

Syntactically it is possible that a floating quantifier does not occur directly after the noun phrase to which it relates. This can lead to ambiguities if there is more than one plural noun phrase in the sentence as in

(49) The boys gave the girls each a present.

In sentence (49) the floating quantifier can distribute with respect to the group denoted by *the boys*; alternatively there is a reading where *each* triggers a distributive reading of *the girls*. Thus using floating quantifiers as disambiguation triggers requires selecting the correct plural noun phrase which is to be understood distributively (see Kamp and Reyle 1993, pp. 441).

A number of other explicit modifiers will have to be considered with respect to their disambiguating effects, for example *individually*, *separately*, *collectively*, *as a whole*, *as a group*, *one-by-one*, *one at a time*, *altogether, at the same time* etc.

(50) a. The system has to manage these internal entries separately.
b. You may need to restore some document library objects separately.
c. The servers can be started all together or individually.
d. When you enter CL commands individually, each command is separately processed.
e. You define these requirements individually.
f. Collectively, these three programs provide a valuable tool.
g. If there are five FTP client sessions established at the same time, there will be 15 FTP servers running.
h. You can place pages one-by-one on the flatbed and scan them.

‘Same’ and ‘Different’. Also *same* and *different* have a disambiguating effect that is not only related to resolving collective/distributive ambiguities with plurals but also to scope ambiguities. As Moltmann (1997) shows there are different uses of *same/different*. I will only demonstrate two of theses uses, for other uses I refer the reader to Moltmann's analysis. On the one hand *same/different* may receive a bound interpretation with a quantified antecedent, as in

(51) a. Everyone saw the same film/a different film.
b. Each of the database networks in library LIB1 reaches a different checkpoint.
c. Each of the tapes support the same density.

The most interesting use in our context is that *same/different* may relate to a preceding plural NP. This use is often called the *internal reading* (Carlson 1987, see also Moltmann 1997, p. 136).

(52) a. John and Mary saw the same film/different films.
b. Two failed storage units are in different mirrored pairs.
c. A source journal could have two remote journals on two different systems.
d. The networks connected by a router may use the same or different physical network protocols.
e. Several employees shall be able to work together on the same project.
f. The students are discussing the same subject.
g. No two packages from one office can have the same addressee.

In a first approximation we can say that in each sentence of (52) the "antecedent" of *same/different* receives a distributive interpretation. The effect of *same* is to pick one and the same object for each individual, the effect of *different* is to pick for each individual element of the antecedent NP one ore more objects that differ from the object(s) for other individuals. For a more technical analysis I recommend Moltmann (1997), and the brief discussion in Kamp and Reyle (1993, pp. 471).

Plurals in Prepositional Phrases. Plural noun phrases often occur as arguments of prepositions.

(53) The verification methodology requires standardized measurement methodologies to eliminate ambiguities in the measurements.

In particular constructions with *of*-PPs occur frequently.

(54) This minimizes the effects of distortions in the corners of the targets.

Furthermore, there are prepositions like *between* or *among* that require plural objects as arguments.

(55) a. Any routing is subject to agreements between the Administrations.
b. A robot has to deliver in-house mail exchanged between *n* offices that lie along a hallway.

As we have seen above the use of plurals in prepositional phrases can also lead to semantic ambiguities. Syntactic attachment ambiguities related to prepositional phrases will not be dealt with in this investigation.

Classifier Constructions. Among the constructions in English that have the surface structure "(Det) +Noun + of + (Det) + Noun" Lehrer (1986) distinguishes a subgroup that he calls classifiers. Classifiers are constructions that include partitives (*two of the students*), pseudo-partitives (*a bunch of bananas*) and measurement noun phrases (*two pounds of apples*). I will briefly address each subgroup in the following paragraphs.

Partitives. Partitives are of the form "Determiner + *of* + definite NP", e.g. *each/ half/ most/ none/ two/ several of the students*. In contrast to purely quantifying noun phrases like *each student* partitive constructions introduce a definite totality of objects; the determiner then selects a suitable part of this totality.

(56) a. Each of the physical files has one access path.
b. The water level reaches one of the two limit values.
c. All of the objects reached a checkpoint together.

Partitives can have both distributive and collective readings although in many cases the distributive readings seem to be preferred.

Pseudo-Partitives. Pseudo-partitives have a similar structure like partitives but they do not have a part-of semantics. Examples for pseudo-partitives are

(57) a. a group of three men
b. a number of children
c. a bunch of bananas

A subtype of pseudo-partitives are so-called measurement noun phrases.

Measurement Noun Phrases. Measurement noun phrases are of the form "numeral + measure word + *of* + mass noun or plural count noun". Examples are

(58) a. two pounds of apples
b. three ounces of gold.
c. a cup of coffee

The part "numeral + measure word" is sometimes called the *measure phrase*. The mass or plural count noun indicates what kind of substance is measured. Measure words can be exact measurement units like *ounce*, *litre* but also less precise units of measurement like container words (*cup*, *box*, *bowl* etc.). See also Schwarzschild (2002) for a more recent discussion measure phrases.

It is not always easy to distinguish measurement constructions from noun phrases with *of*-PP modifiers like

(59) two photographs of a man

Akmajian and Lehrer (1976) argue that the constructions have a different syntactic structure. In the sentence (59) the head noun of the complex NP is *photograph* and the *of*-phrase is a PP-modifier, while in (58)a the head noun of the complex noun phrase is *apples* and *two pounds of* is a quantificational phrases that acts as a complex determiner determining the amount of apples. There are complex noun phrases that can be used in both constructions:

(60) a. Two bottles of wine broke.
b. Two bottles of wine spilled.

In (60)a the head of the complex NP is *bottles*, whereas (60)b is a measurement construction where the head of the noun phrase is *wine*. In computational applications measure words have to be predefined in a lexicon to be automatically recognized. The problem is that although the set of exact measure words is small and closed and can be easily predefined, the set of inexact measurement words cannot be realistically predetermined since many container words can be reinterpreted as measurement words (*a pocketful of coins*, *a pocket of coins*, *a shelf of books* etc.).

Apart from measurement constructions there are other possibilities to express measurements in English, for example

(61) John is five feet tall.
(62) The size of John is five feet.
(63) The apples weigh three kilos.

These measurement constructions are related to the semantics of special words expressing the dimension of measuring, e.g. *tall*, *size*, *weigh*. I will not go into details of the syntax and semantics of this type of measurement constructions.

Reciprocals. In the generative tradition there is a vast amount of literature on so-called reciprocal constructions (constructions with *each other*, *one another* etc.). Since these constructions have a systematic connection to plurals I will address them shortly.

(64) John and Mary know each other.
(65) The students know each other.

Reciprocals operate on the verb phrase resulting in predicates that require a plural subject. The semantic literature on reciprocals concentrates on the question how many objects have to stand in the respective relation to make the sentence true. For example, in (65) it is not clear how many students have to know how many other students to make the sentence true. For discussions on reciprocals see e.g. Langendoen (1978), Roberts (1987), Heim, Lasnik and Mey (1991), Schwarzschild (1992) or Moltmann (1992).

Copula ‘Be’. The copula *be* can have different types of subjects and predicative complements which can be in the singular or in the plural. For example, *be* often takes adjectival phrases, prepositional phrases or noun phrases as complements. Sentences with adjectives as predicative complements can have a collective or distributive reading depending on the adjective.

(66) a. The books are heavy/expensive.
b. The two numbers are equal/identical/adjacent.
c. The last four characters are numeric.

Sentence (66)a is ambiguous between a collective and a distributive reading, in (66)b a distributive reading is not possible, whereas (66)c suggests a distributive reading of the subject noun phrase.

With prepositional phrases as complements we have the full range of phenomena associated with relational plural sentences, as in

(67) a. The books are in the shelves.
b. The books are in a library.
c. The books are in university libraries.
d. The terms are in the correct order.
e. These files are in ASCII format

With noun phrases as predicative complements we find constructions with indefinite noun phrases as in (68)a and (68)b, definite noun phrases as in (68)c and (68)d or coordination as in (68)e.

(68) a. 25 books are history books.
b. John and Mary are a happy couple.
c. These are the numbers used in the program's source list.
d. Sustained oxidation and erosion are the major modes of degradation.
e. The two parts referred to are the user space and the user index.

2.3.7 Combinatorial Explosion Problem

The relatively simple examples in the previous sections showed that the use of plurals is a pervasive phenomenon in natural language. Plurals occur in a variety of different syntactic constructions, and, what is more, they can lead to a combinatorial explosion of possible readings if one assumes collective, distributive, cumulative and possibly other interpretations of noun phrases plus resulting scope ambiguities. Assuming that a semantic interpretation module assigns each of the possible readings a separate logical form as a semantic representation we end up with a combinatorial explosion of representations even for very simple sentences – which is not only computationally inefficient but also theoretically implausible.

First, it is questionable whether all of the proposed readings are available at all or whether traditional systems rather suffer from an empirically not adequate massive overgeneration of semantic representations. This relates to the question whether plurals are really ambiguous in all cases (leading to the multiplicity of semantic representations) or whether some of the cases are rather a matter of indeterminacy (leading to just one less specific representation). I will address the difference between ambiguity and indeterminacy in the following section 2.4.

Furthermore, human beings seem to be able to deal with most plural sentences without effort. Very often they do not perceive that a sentence is ambiguous since knowledge about the context, the world or the speaker's intention rule out a number of theoretically possible readings and leave only plausible readings. This conflict between theoretically predicted ambiguities and perceived ambiguities has been called the "combinatorial explosion puzzle" by Poesio (1996). Poesio describes the combinatorial explosion puzzle as "one of the most fundamental questions to be addressed by a theory of language processing and a substantial problem for developers of Natural Language Processing (NLP) systems." (Poesio 1996, p. 159). Whereas humans easily arrive at a preferred interpretation NLP systems have great trouble in dealing with ambiguous input. This conflict is in particular relevant for NLP systems that indeed have to disambiguate. Take for example an NLP system that upon receiving natural language input has to perform actions like booking a flight or reserving a seat. The system must either be able to arrive at the preferred interpretation of the input, or, if there is no clear preference, the system has to realize that the input is ambiguous and ask for clarification.

Thus we can distinguish three major tasks: distinguish ambiguity from indeterminacy, detect real ambiguities and resolve ambiguities. Distinguishing ambiguity from indeterminacy is the task of the designers of the NLP system since only they decide which constructions are assigned different semantic representations, the second task – detecting systematic ambiguities – is in principle no problem for an NLP system since the possible ambiguities are generated by systematic rules where no unconscious background knowledge is involved that blocks possible readings, the final task however – disambiguation – is a challenge.

The next section will give a precise definition of the concepts 'ambiguity' and 'indeterminacy', section 2.5 will discuss possible strategies to deal with ambiguities in practical applications. The plural specific applications will be discussed in section 5.2.2.

2.4 A Definition of Semantic Ambiguity

2.4.1 Ambiguity, Vagueness and Indeterminacy

The definition of the concept 'ambiguity' has been the topic of much debate. Not all of the cases listed above are classified to be ambiguous by all authors. It has been argued that some of the cases are in fact a matter of vagueness or of indeterminacy. Several structural and semantic definitions of and distinctions between ambiguity and vagueness have been proposed in the literature (Pinkal 1985, 1991, 1995, Gillon 1987). Since plurals cause semantic ambiguities I will only discuss *semantic* characterizations of the concepts ambiguity, vagueness and indeterminacy.

Ambiguity. The general intuition is that an expression is ambiguous if it has more than one meaning, which means that it can be understood in more than one way. The process of associating syntactic expressions with their meaning is commonly called 'semantic interpretation' – which can be done in terms of a semantic representation or in terms of values in a model. Thus we can also say that an expression is ambiguous if it has more than one interpretation (more than one semantic representation or more than one value in a model).

As we have seen above there are different types of ambiguity depending on what is causing the ambiguity, e.g. single words or whole sentences can be ambiguous. Arguing in terms of values in a model we can say that a *word* is ambiguous if there can be at least one state of affairs in which the word denotes different sets of objects depending on which of the meanings is chosen. For example, the word *croak* denotes under one interpretation the set of objects making a sound like a frog, under a different interpretation it denotes the set of objects that die unexpectedly. In all situations where at least one object does not both die and make a sound like a frog these denoted sets of objects are different. If a *sentence* is semantically ambiguous there can be at least one state of affairs in which the sentence is true under one interpretation (i.e. has the value 'true' in a model) but false under one of its other interpretations. For example, in all situations where Kermit does not at the same time die and make a sound like a frog the sentence *Kermit croaks* is either true or false depending on which interpretation is chosen.

Gillon (1987) captures this intuition by giving the following definition of semantic ambiguity of sentences:

Definition 1. Ambiguity (Gillon 1987, p. 202)

> A sentence is ambiguous iff, with respect to a given state of affairs, the sentence can be both truly affirmed and truly denied. (Gillon 1987, p. 202)

This definition, however, is not fully precise. Imagine a state of affairs where Kermit has both the property of dying and of making a frog-like sound. In such a state of affairs the sentence cannot be truly denied in neither of its meanings, thus the sentence wouldn't come out as ambiguous. The sentence has to be evaluated with respect to all possible worlds, i.e. the two meanings of the sentence correspond to different 'propositions'. To capture this intuition but without elaborating on the concept proposition I therefore propose to slightly modify Gillon's criterion to:

Definition 2. Ambiguity (Modification of Gillon 1987)

> A sentence is ambiguous iff, there is at least one possible state of affairs, which respect to which the sentence can be both truly affirmed and truly denied.

This semantic definition of ambiguity makes it difficult to distinguish ambiguity from the concept 'vagueness' to which we come now.

Vagueness. Intuitively, an expression (word or sentence) is vague if the meaning of the expression is not exactly determined, i.e. if in certain states of affairs there are no clear-cut criteria whether a word applies to an object or whether a sentence is true in that state of affairs. Several types of vagueness are distinguished (Pinkal 1991); for the purpose of illustration we list just two of them. One type of vagueness results from *uncertain boundaries*. That means the applicability of a word is clear for a majority of cases but for boundary cases the applicability is uncertain. Take as an example the word *red*. There are many clear cases where we are in no doubt whether the word applies or not. But there are also boundary cases where an object is e.g. dark-pink and it is uncertain whether we would still apply the word *red* to the object. Other typical examples are *middle-aged* or *bald*. A second type of vague expressions are *relative expressions* where in almost all cases it is uncertain whether the expression applies or not. Typical examples are degree adjectives like *fast* or *tall* which are vague in the sense that the precise degree of speed or tallness is not clear. Depending on the given classes of comparison the words, or the sentences containing the words, are interpreted differently.

It is important to note that the uncertainty whether vague expressions apply or not is not due to the ignorance of the hearers. Even if you know that a person is 39 this additional knowledge doesn't settle the question whether *middle-aged* applies to that person or not. A definite answer is not possible due to the intrinsic uncertainty or impreciseness in the meaning of the concept *middle-aged*.

Indefiniteness. Both ambiguous and vague expressions (words or sentences) have in common that there are possible states of affairs where is not definite whether the word applies or whether the sentence is true or false. Building on this observation Pinkal (1985, translated as 1995) gives an interesting distinction of the concepts 'ambiguity' and 'vagueness'. In essence his definition is similar to that of Gillon. Pinkal's definition, however, has become prominent in the recent literature about the treatment of semantic ambiguity (e.g. Poesio 1996). We have said that both ambiguous and vague expressions can have more than one interpretation, which Pinkal (for the case of sentences) puts as:

Definition 3. Indefiniteness (Pinkal 1995, p. 15)

> … in certain situations, despite sufficient knowledge of the relevant facts, neither "true" nor "false" can be clearly assigned as [the] truth value [of the sentence].

Pinkal goes on to argue that ambiguity and vagueness are both instances of 'indefiniteness'. Indefinite sentences can be made more precise and in doing so can be made true or false. The difference between vagueness and ambiguity is whether this 'precisification' is continuous or discrete. For a vague predicate like *red* or *fast* there is a continuum of possible precisifications depending on the shades of red or the degrees of speed agreed to. An ambiguous predicate like *croak* has a much sharper division between its specific senses.

Since in this thesis I'm focusing on ambiguity, not vagueness, I will not further elaborate on Pinkal's general definition of indefiniteness but mainly use Gillon's more specific definition of ambiguity.

Indeterminacy. It is important to distinguish 'ambiguity' and 'vagueness' as defined by Gillon or Pinkal from the concepts 'indeterminacy' or 'non-specificity'. A sentence is indeterminate, if it is definitely true or false but it could be made more specific. For example, the sentence *John found a glove.* is indeterminate about whether it was a left- or a right-handed glove; the sentence *John and Mary left.* is indeterminate about whether John and Mary left together or whether they left separately. The predicate *glove* is not vague as to left- or right-handedness, neither is the verb *leave* vague as to leaving together or separately, rather the predicates are indeterminate or non-specific as to the properties mentioned. Sometimes the concepts 'non-specific', 'unspecified', 'uninformative', 'non-determined' and others are used instead of the concept ‘indeterminate’ (see Zwicky and Sadock 1975, p. 2). Some authors also include the concept ‘vague’ in this category. I refer, however, to Pinkal’s definition of vagueness. It hardly needs to be noted that just about any natural language expression is indeterminate or non-specific in some respect.

The main difference between ambiguity and indeterminacy is that an ambiguous sentence cannot be assigned a definite truth-value in certain situations because it is not clear which of the possible interpretations has to be chosen. Unless the intended interpretation is ‘precisified’ the truth-value cannot be determined. If the sentence is merely indeterminate, however, it does have just one interpretation and thus a definite truth-value in a given state of affairs. It is, however, possible to elaborate on this interpretation and thus make it more specific. But the facts added during elaboration do not affect the truth value of the original indeterminate sentence. Of course, any type of completely irrelevant information could be added to make an indeterminate statement more specific with respect to certain properties, e.g. the word “parent” is indeterminate with respect to being of any particular size.

Since almost all natural language sentences are indeterminate with respect to some (irrelevant) information, ‘indeterminacy’ at first sight doesn’t seem to be a very interesting or useful concept. However, as soon as we construct semantic representations for natural language sentences and are interested in reducing a proliferation of semantic representations the distinction between ambiguity and indeterminacy becomes important. An ambiguous sentence gets different, often logically incompatible semantic representations and thus can lead to a proliferation of readings. An indeterminate sentence, in contrast, gets just one semantic representation that encompasses possible situations in a unified way. This is in line with the speaker’s intention to leave open details about the state of affairs that is being described.

It is this aspect of the concepts ambiguity and indeterminacy that is important for my treatment of plural ambiguities. When does the occurrence of a plural noun phrases lead to real ambiguity (i.e. to more than on semantic representation) and when merely to indeterminacy (i.e. just one less specific semantic representation)? In section 4.3 I will argue that so-called ‘collective’ readings of plurals are in many cases indeterminate with respect to how the individuals making up the collection are involved in the relation denoted by the verb. Take again as an example sentence *John and Mary left*. We argued that the sentence is indeterminate with respect to whether John and Mary left separately or whether they left together. For my considerations concerning plurals I will restrict the term indeterminacy to dimensions of this type.

Underspecification. The concept ‘underspecification’ used in recent research on semantic ambiguity (see e.g. van Deemter and Peters 1996) has to be contrasted to the concepts ambiguity and indeterminacy. Generally, the term ‘underspecified’ is used for expressions which may have different (truth-)values depending on the way additional facts are ‘filled in’. In this general sense underspecified sentences correspond to Pinkal’s class of indefinite expressions, i.e. to ambiguous and vague expressions. The concept ‘underspecified’ is also used more specifically for the kind of semantic representations that are devised to represent ambiguous expressions. Rather than disjunctively enumerating all possible readings of ambiguous expressions, the set of readings of such an expression is modelled by an 'underspecified' representation that comprises all its possible readings. It is this more technical definition of underspecification that I will refer to when using the term 'underspecified'.

2.4.2 Definitions Adopted for the Study of Plural Ambiguity

For my investigations of plural ambiguity and plural disambiguation I will mainly use the concepts 'ambiguity' and 'indeterminacy'. Furthermore, I'm mainly investigating ambiguity and indeterminacy of sentences, not of words. Therefore the definitions will be restricted to the sentence level.

The following definitions will be used in my investigation.

Definition 4. Ambiguity
A sentence is **ambiguous** iff, there is at least one possible state of affairs, which respect to which the sentence can be both truly affirmed and truly denied.

This corresponds basically to the definition of ambiguity given by Gillon (1987). Ambiguous sentences are assigned different logically independent semantic representations. Assume for example that the sentence

(69) Two men lifted a table.

is uttered in a situation where there are two men and each of them lifted a table, but they didn't lift a table together. Then definition Definition 4. predicts that the sentence is ambiguous since with respect to the given state of affairs the sentence can be truly affirmed in its distributive reading and truly denied in its collective reading.

Definition 5. Indeterminacy
A sentence s is **indeterminate**, iff with respect to any given state of affairs s is definitely true or false but it can be **elaborated** on, i.e. made more specific in different ways: A sentence can be elaborated in different ways if there are at least two elaborating sentences s_1 and s_2 both of which entail s but there is at least one possible state of affairs with respect to which s_1 and s_2 have different truth values.

Note that an elaboration s_1 of a sentence s is true with respect to a subset of the state of affairs in which s is true. Indeterminate sentences are assigned just one semantic representation. The semantic representations of the elaborations entail the original indeterminate semantic representations. For example, assume the sentence

(70) The students read the newspapers. (Link and Schütze 1991)

is uttered in a situation where there is a set of students and a set of newspapers and each student reads each of the newspapers. The sentence is definitely true with respect to that situation. Assume the same sentence is uttered in a situation where the students as a group read each of the newspapers. Again the sentence is definitely true. I assume, like many other authors (Roberts 1987, Link 1991, Carpenter 1995 and others) that sentence (70) is not ambiguous between a strong symmetric or a weak symmetric reading but that in its doubly collective reading it is indeterminate with respect to the participation of the elements of the subject and object denotation with respect to the predicate. How the elements participated is a matter of elaboration. For example, the sentences

(71) a. The students individually read the newspapers one after the other.
b. The students as a group read the newspapers one after the other.

are both elaborations of (70) which entail (70) but there is at least one possible state of affairs with respect to which (71)a is true and (71)b is false; for example in a situation where the students as a group read each of the newspapers but it is not the case that each student reads each of the newspapers. With Carpenter (1995, p. 10) I will argue that the levels of participation are a matter of lexical semantics and should not get a separate representation in the logical form.

2.4.3 Ambiguity Tests

Beside the abstract definitions of the concepts ambiguity and indeterminacy a set of syntactic tests have been devised that help to distinguish ambiguous from non-ambiguous examples in problematic cases. Zwicky and Sadock (1975) discuss some of these tests. An example is the "do so" conjunction test which belongs to the class of "identity tests" developed by Lakoff (1971). When we form a conjunction like

(72) John saw her duck, and so did George.

where the first conjunct is a truly ambiguous sentence we do not get a 'crossed understanding' for the second sentence, e.g. we do not get a reading where John saw e.g. Mary's swimming bird whereas George saw Mary moving her body downwards. Take as a contrast the following sentences:

(73) a. John and Martha left, and so did Dick and Pat.
b. John lifted two tables, and so did Mary.

The first conjunct in sentence (73)a can have two 'interpretations'. It can mean that John and Mary left separately, or that they left together. In contrast to (72) sentence (73)a allows for a 'crossed understanding' where John and Mary leave separately and Dick and Pat leave together. This result suggests that sentence (72) is ambiguous whereas (73)a is merely indeterminate. The same observation holds for (73)b.

Note however, that Zwicky and Sadock (1975, pp. 23) observe, that identity tests can only properly identify "polar" ambiguities (such as the ambiguity of *duck* between two contradictory readings), but not "privative" ambiguities, where one semantic representation is more specific than the other, but the readings are not contradictory. For example, the term *dog* can be used to indicate the general meaning 'canine' and the specific meaning 'male canine'. The problem is that any sentence with a more specific understanding is consistent with the more general understanding, logically the more specific understanding implies the general understanding. This problem also occurs in the scopally ambiguous sentence

(74) Every man lifted a table, so did every woman.

where the first conjunct can mean that every man lifted a different table and the second conjunct can mean that every woman lifted the same table – thus a crossed understanding is possible. A similar problem occurs with (73)b. Thus, with the identity test ambiguities cannot be distinguished from more specific elaborations of an indeterminate sentence.

A further test is the negation of the sentence that is to be checked for ambiguity or indeterminacy. Negation of an ambiguous sentence just negates one of its interpretations not the other interpretations. For the sentence

(75) Kermit didn't croak.

to be true it could be the case that Kermit didn't die, but still he could have made frog-like sound or the other way round. However, negation of an indeterminate sentence operates on all of its possible elaborations. For example in

(76) John and Martha didn't leave.

both possibilities are denied, for the sentence to be true neither did John and Martha leave separately nor did they leave together. The negation test is a consequence of the definition of the concepts 'ambiguity' and 'vagueness'. More comments on the negation test can be found in Verkuyl (1988) and Carpenter (1995).

There are a number of criticisms against both the definitions of ambiguity and intedeterminacy in section 2.4 and the ambiguity tests in this section. It is commonly held that ambiguity tests have flaws and that they are difficult to apply consistently. We will see in chapter 4 that there is not always a definite answer whether it is adequate to assume ambiguity or indeterminacy. Rather, there are cases where the assumption of ambiguity or indeterminacy is dependent on the task of the semantic representation, e.g. whether we develop semantic representations and evaluate them with respect to data bases or whether we develop semantic representations that represent the meaning of natural language text efficiently. Thus, for the time being I take the definition and the tests as a rough guideline to distinguish ambiguity from indeterminacy.

2.5 Semantic Ambiguity Processing in NLP

Ambiguity occurs on different levels of language processing. Plurals are a frequent source of semantic ambiguities. I have addressed the combinatorial explosion puzzle – i.e. the discrepancy between possible and perceived ambiguity – which is difficult to solve within natural language processing systems. Different strategies to deal with the problems of ambiguity and disambiguation in natural language processing systems have been proposed (FRACAS Deliverable D9, 1994c).

The *Generate and Test* approach first generates for every theoretically possible reading a semantic representation and in a second step eliminates the implausible ones. The elimination can be done by presenting all interpretations to the user who selects the intended one (Macias and Pulman 1995), or by formalizing relevant contextual knowledge (world knowledge, linguistic context, lexical semantics, pragmatics etc.) to determine the intended interpretation (Hindle and Rooth 1993). The disadvantage of this approach is that complex sentences tend to have a large number of readings and it is computationally inefficient to first generate all representations and then to eliminate most of them. Manually selecting one of the many representations is a burden on the user. Furthermore, formalizing relevant contextual knowledge to filter the readings has proven to be very difficult, if not impossible.

To avoid problems with combinatorial explosion, many have proposed the use of *underspecified semantic representations*. In the *Underspecification* approach a semantically ambiguous sentence gets – in a first processing step – just one representation which is derivable from the syntactic form of the sentence leaving certain aspects of the meaning – for example the scope of operators – not specified. Fully specified readings are obtained by filling in material that is derived from contextual knowledge instead of eliminating fully specified but contextually implausible interpretations as in the *Generate and Test* approach. Examples for underspecified languages are the 'Logical Form' proposed by Schubert and Pelletier (1982), the 'Logical Form' discussed in Allen (1995), the 'Quasi Logical Form (QLF)' used in Alshawi (1992), Reyle's (1993) 'Underspecified DRT', Poesio's (1994) 'Conversation Representation Theory', or the 'Hole Semantics' by Bos (Bos et al. 1996). A good survey of the area of underspecification can be found in the collection edited by van Deemter and Peters (1996).The Underspecification approach tackles the efficiency problem of the *Generate and Test* method. However, the drawback remains that it is very hard to formalize the contextual knowledge that leads to more specific interpretations.

Originally underspecified approaches were introduced in computational applications (Woods 1977) to separate 'context-independent' from 'context dependent' aspects of the interpretation, thus making either part reusable for different applications. The final aim was, however, to generate fully specified readings and underspecified representations were just an intermediate step. Thus these approaches to underspecification were in a way conservative in that the underspecified representation was only a computationally practical intermediate step to arrive at a fully specified meaning. In my approach to plural disambiguation I will also suggest an intermediate underspecified representation.

Recently, researches started to investigate the properties of the underspecified representations themselves (van Deemter and Peters 1996), for example, they try to define inference procedures directly on underspecified representations. This is particularly interesting, because for natural language processing tasks like automatic translation ambiguities do not always have to be resolved but can be transferred to the target language. However, currently there are no well-developed reasoning mechanisms that perform reasoning directly on underspecified representations. Existing reasoning techniques mostly rely on disjunctively listing all the representations that the underspecified representation comprises. But then there is no efficiency gain compared to *Generate and Test*.

There are also statistical methods for disambiguation, but these approaches will not be dealt with in this thesis (see e.g. Allen 1995, ch. 7 for a first overview).

3 Existing Approaches To Represent Plurals

3.1 Overview

This chapter explains important existing techniques within formal semantics to represent plural ambiguities. Plural ambiguities occur in connection with plural noun phrases. In section 3.2 I will therefore give a brief overview of two influential approaches to formal noun phrase semantics: *Generalized Quantifier Theory* and *Discourse Related Approaches*. The overview also establishes a common terminology for chapter 3. Readers familiar with the two approaches can safely skip this section.

In section 3.3 I will then present some influential existing approaches to the representation of plural noun phrases. These include the proposals of Scha (1981) and of Link (1983, 1991). Whereas Scha locates the ambiguity of the plurals within the noun phrase, Link attributes it to verb phrases. Scha's noun phrase centred approach suffers from an overgeneration of semantic readings and uses computationally impractical logical forms. Link's verb phrase centred approach can be formulated in a computationally more practical first-order language yet the problem of ambiguity explosion is not solved. A third approach by Verkuyl and van der Does (1991) tries to give plural noun phrases just one semantically weak representation that encompasses the other readings. The approach, however, suffers from empirical shortcomings and the approach is also formulated in a computationally impractical language. Finally, there are approaches that attribute the ambiguity not to single elements but to global factors. For computational applications I argue this to be the most plausible strategy, however, the existing approaches are not formulated in a computationally practical way. In chapters 4 and 5 of this thesis I will therefore develop a computationally suitable formalization of a global strategy.

3.2 Basic Concepts of Noun Phrase Semantics

3.2.1 Generalized Quantifier Theory (GQT)

3.2.1.1 Motivation

In first-order logic noun-phrases are represented either as constants that denote individuals from the individual domain (proper nouns like *John*) or noun phrases lead to existential (*a man*) or universal (*every man*) quantification over the individual domain. For example the sentences

(1) a. John disappears.
b. Every man disappears.

are represented as

(2) a. disappear(john)
b. $\forall$X(man(X) $\rightarrow$ disappear(X))

Montague (1974) was one of the first to argue that a representation of noun phrases in first-order logic is not fully satisfactory. One of the reasons is methodological: The syntactic structure of quantified formulae in predicate logic is completely different from the syntactic structure of quantified sentences in natural language. Whereas in formula (2)a there is a single expression for the noun phrase *John*, in formula (2)b there is no such expression that corresponds to the noun phrase *every man*. The meaning of the noun phrase *every man* cannot be given in isolation, but is introduced syncategorematically, i.e. in the context of the whole sentence. The comparison of (2)a and (2)b shows furthermore that different types of noun phrases lead to different types of logical representations. This means that the interpretation of noun phrases does not comply with the principle of compositionality in formal languages according to which every expression of a given syntactic category should receive the same type of interpretation.

A second, more principled criticism is, that there are forms of quantification in natural language that are not expressible or definable in terms of the first-order logical quantifiers. For example, the noun phrase *more than half of the men* is not expressible in terms of just first-order quantifiers, since its interpretation requires a one-to-one mapping between two finite or infinite sets dependent on a well-ordering by cardinality (see Barwise and Cooper 1981 for a proof).

Montague (1974) therefore initiated a higher-order analysis of natural language quantification according to which all noun phrases are categorematically interpreted as so-called *generalized quantifiers*. There is a functional and an equivalent relational formulation of the semantics of determiners. According to the functional version every noun phrase (including proper nouns) denotes a function from sets of individuals to truth-values. For example the subject noun phrase *every man* in (1)b denotes a function that takes as an argument the interpretation of the verb phrase *disappears* (the set of individuals who disappear). The sentence is true if the set of individuals who disappear is a member of the family of sets interpreting *every man*. Formally, this idea is represented by λ-abstraction over predicates, such that the noun phrase *every man* can be compositionally translated into a higher-order term. I will ignore here the intensional components of Montague's analysis.

(3) $\lambda P[\forall X(\text{man}(X) \rightarrow P(X))]$

The determiner *every* itself can be represented by further abstraction as:

(4) $\lambda Q \lambda P[\forall X(Q(X) \rightarrow P(X))]$

A higher order analysis applied to the proper noun *John* yields the following semantic representation:

(5) $\lambda P[P(\text{john})]$

Instead of taking determiners like *every* as functions from sets to sets of sets, determiners can equivalently be represented in a flattened relational way, viz. as relations between sets. A determiner (*Det*) is then analysed as a relation between the set of individuals interpreting the common noun (CN) of the subject noun phrase – the restrictor, and the set of individuals interpreting the verb phrase (VP) – the scope. This leads to the following general definition (adapted from Partee et. al. 1990, p. 374) where E is the set of entities and [[.]] is the compositional interpretation function:

(6) A determiner is a function D in a model $\mathbf{M} = \langle E, [[.]]\rangle$ assigning to the domain of entities E a binary relation between subsets A and B.

$$[[\text{Det } [[CN]]\,[[VP]]\,]]_E = [[\text{Det}(A,B)]]_E = D_E AB$$

Different determiners are assigned different relations. For example, *every* is associated with the subset-relation. Thus sentence (1)b is true if the set of men is a subset of the set of objects that disappear.

(7) $[[man]] \subseteq [[disappear]]$

More generally, a sentence of the form *every A B* is true if the denotation of *A* is a subset of the denotation of *B*:

(8) $every\ AB \Leftrightarrow A \subseteq B$

The interpretation of the noun phrase *every A* can be then given as (where *E* is the domain of entities):

(9) $every\ A = \{B \subseteq E: A \subseteq B\}$
$= \{B \subseteq E: A \cap B = A\}$

The interpretation of just the determiner *every* in the relational format yields:

(10) $every = \{\langle A, B\rangle \in E \times E: A \subseteq B\}$

Semantically, the relational view on determiners does not differ crucially from the functional one (van der Does 1992, p. 4). Functional determiners (D^{fun}) can be tied uniquely to relational determiners (D^{rel}) via the following equivalence:

(11) $B \in D^{fun}(A) \quad \Leftrightarrow \quad \langle A, B\rangle \in D^{rel}$

In the following I will mainly use the relational notation for ease of readability.

3.2.1.2 Generalized Quantifiers

Representing the meaning of quantifiers as relations between sets not only allows to deal with standard quantifiers (i.e. the first-order existential and universal quantifier) compositionally, it can also be used to represent non-standard quantifiers like *most*, *few*, *more than seven* in an analogous manner. Many of the non-standard quantifiers introduce plural noun phrases and are thus of particular interest here. The extension of the field of quantification to non-standard quantifiers is the subject of "Generalized Quantifier Theory" (GQT). GQT was introduced by Barwise and Cooper (1981) into the semantics of natural language. The term itself dates back to the mathematician Mostowski (1957). In GQT the focus of investigation changes from the noun phrase to the determiner of the noun phrase.

The suggested interpretation of some non-standard quantifiers in relational perspective is given in Table 1 (adapted from Westerståhl 1989, p. 44). Note, that in Table 1 cardinal numbers like

$every_E\, AB = all_E\, AB$	$\Leftrightarrow$	$A \subseteq B = A \cap B = A$
$some^{sg}{}_E\, AB$	$\Leftrightarrow$	$A \cap B \neq \emptyset$
$some^{pl}{}_E\, AB$	$\Leftrightarrow$	$\lvert A \cap B\rvert \geq 2$
$no_E\, AB$	$\Leftrightarrow$	$A \cap B = \emptyset$
$most^1{}_E\, AB$	$\Leftrightarrow$	$\lvert A \cap B\rvert > \lvert A - B\rvert = \lvert A \cap B\rvert / \lvert A\rvert > 0.5$
$both_E\, AB$	$\Leftrightarrow$	$every_E\, AB\ \&\ \lvert A\rvert = 2$
$neither_E\, AB$	$\Leftrightarrow$	$no_E\, AB\ \&\ \lvert A\rvert = 2$
one	=	$some_{sg}$
$two_E\, AB$	$\Leftrightarrow$	$\lvert A \cap B\rvert \geq 2$
$n_E\, AB$	$\Leftrightarrow$	$\lvert A \cap B\rvert \geq \mathrm{n}$
$exactly\ n_E\, AB$	$\Leftrightarrow$	$\lvert A \cap B\rvert = \mathrm{n}$
$at\ most\ n_E\, AB$	$\Leftrightarrow$	$\lvert A \cap B\rvert \leq \mathrm{n}$
$more\ than\ half\ the_E\, AB$	$\Leftrightarrow$	$\lvert A \cap B\rvert / \lvert A\rvert > 0.5$
$the^{sg}{}_E\, AB$	$\Leftrightarrow$	$all_E\, AB\ \&\ \lvert A\rvert = 1$
$the^{pl}{}_E\, AB$	$\Leftrightarrow$	$all_E\, AB\ \&\ \lvert A\rvert > 1$
$the\ seven_E\, AB$	$\Leftrightarrow$	$all_E\, AB\ \&\ \lvert A\rvert = 7$
$n\%\ of\ the_E\, AB$	$\Leftrightarrow$	$\lvert A \cap B\rvert = \mathrm{n}/100 * \lvert A\rvert$

Table 1 Relational Interpretation of Some Non-Standard Quantifiers

two are interpreted as *at least two*, although it can be argued that cardinal numbers sometimes have an "exactly"-reading. We will discuss this below.

To avoid potential terminological confusion here I want to point at the distinction between the concepts 'determiner' and 'quantifier'. There are contexts in which the term 'quantifier' is used to denote the whole noun phrase (e.g. *every man*) whereas 'determiner' only refers to the article (e.g. *every*). In other contexts the concept 'quantifier' is used only for the article and not for the whole noun phrase. I am using the concept 'quantifier' here rather loosely and, depending on the context, relate it to whole noun phrases or only to determiners.

3.2.1.3 Types of Quantifiers

Quantifiers are subclassified according to a number of criteria. For a good discussion and summary see Hess (1989, section 4) or Partee et. al. (1990, ch. 14). I will only list the most important concepts.

Cardinal Quantifiers. Many of the quantifiers listed above impose a cardinality condition for the intersection $A \cap B$, that is the intersection of the noun meaning and the verb phrase meaning. For example, *exactly two N* says that the number of elements in the intersection must equal two. These quantifiers are called *cardinal* or *intersective quantifiers*, as they depend solely on the cardinality of the intersection (e.g. whether it consists of no elements, of one element, of more than two elements etc.). A typical test to distinguish cardinality quantifiers are so-called existential *there*-insertion sentences (Keenan 1987) like

(12) There are some/several/a few/no/many/at least three/not more than seven books.

Thus in the cardinality reading of *two* the sentence

(13) Two men came to the party.

means that the set of objects that are men and came to the party contains two different elements. The cardinality interpretation is plausible if the cardinal *two* is used in an unstressed version.

Proportional Quantifiers. For proportional quantifiers not only the cardinality of the intersection of A and B but also the cardinality of the noun denotation A (or some other contextually given set based on A) is relevant. Proportional quantifiers are not felicitous in *there*-insertion sentences. Examples are *most*, *all*, *every*, *more than half the*, *25 percent of the* etc. Proportional quantifiers presuppose the existence of a base set with respect to which the intersection of A and B is compared. This comparison with the base set can be explicit as in

(14) 25 percent of the students/ most of the students passed the exam.

where the explicit comparison is expressed by *25 percent of* and *most of*, respectively. The comparison can also be implicit as in

(15) Many men came to the party.

In the proportional reading of *many* the sentence expresses that the number of men who came to the party is considerably large with respect to some contextually given base set of men, some of which, presumably, didn't come to the party. There is no explicit ratio that compares the cardinalities of the men who came to the party with the set of all men. We simply determine the cardinality of the subset with respect to the implicitly given base set to be many. Note that the proportional reading of *many* occurs when *many* is used in a stressed version.

Vague, Context-Dependent and Ambiguous Quantifiers. Very often neither proportional nor cardinal quantifiers can be reduced to *precise* relations between sets. That means, we cannot give precise conditions for when a quantifier relation actually obtains. These quantifiers are called *vague quantifiers*. There are vague cardinality and vague proportional quantifiers.

Examples for *vague cardinality quantifiers* are *many*, *several*, *a few*, *a lot of*, *few*. If no theory of vagueness is incorporated, vague determiners are sometimes given an idealized precise version which of course only approximates their meaning, e.g.

(16) *several* = *(at least) three*
a few = *some*

The determiners *many* and *few* are not only vague but also *context-dependent* in the sense that the standard of comparison may vary from context to context. For the cardinality reading of *many* and *few* the standard is often dependent on the size of the domain. For example, the sentence *Many men disappear.* is true when there are 6 objects in our domain and 5 of them are men who disappear. The sentence is, however, not true when the domain contains 1000 objects and 5 of them are men who disappear. To capture this interpretation the cardinality that counts as many has to be calculated from the size of the domain E:

(17) $$many_{card}\,AB \quad \Leftrightarrow \quad |A \cap B| \geq f(E)$$

Here the expression $f(E)$ calculates the cardinality that counts as many for the domain E.

There are also *vague proportional quantifiers*. Interestingly, *many* and *few* are *ambiguous* between a vague cardinal and a vague proportional reading. In the vague proportional reading *many* and *few* express a vaguely specified proportion. For example, six students with a broken leg in a class of twenty might be considered many, whereas six out of twenty people that are right-handed is not considered to be many. That means, the standard to which an amount is compared varies from context to context which can be represented as:

(18) $$many_{prop}AB \quad \Leftrightarrow \quad |A \cap B| > \mathrm{c}|A|$$

This expresses for example that the cardinality of students (As) that are right-handed (Bs) is high compared to a contextually given fixed ratio (c) of students. Here c is the contextual factor that might, for the example be 1/4 for students with broken legs, but 3/4 for students that are right-handed. There are various proposals to include different contextual parameters in the interpretation of context-dependent determiners (see for example Partee 1990, pp. 395 for a discussion). The problem for automatic applications is that the contextual parameter is implicit and has to be externally determined which would require additional non-linguistic knowledge.

These brief remarks about vague and context-dependent determiners indicate that it is very hard, if not impossible to determine the relevant contextual factors automatically. Since for practical applications not all “imprecise” determiners can be simply ignored the meaning of these determiners will have to be approximated with the aim of reconstructing relevant inferences associated with these quantifiers, ideally without requiring the semantics to determine the contextual factors. I will discuss in section 4.5 how such an approximation can be formalized.

There are many other types of quantifiers that are studied within Generalized Quantifier Theory, e.g. definites like *the men* or *John’s cats*, partitives like *at least five of the men*, comparatives like *more female than male students* etc. Sometimes quantifiers that come with a particular condition on their restrictor are called *presuppositional quantifiers*. Examples are *the*, *both*, *neither*, *the seven* etc. I will discuss some of these determiners in section 4.5 where I introduce the representation developed in this thesis.

Classification. A wide variety of terms has been proposed to distinguish the two functions of quantifiers, the cardinality function and the proportional function. Suggested oppositions are “quantificational” as opposed to “cardinal”, “strong” vs. “weak”, “definite” vs. “indefinite”, finally “relative” vs. “absolute”. Hess (1989, pp. 108) gives a good comprehensive discussion with relevant references. We have seen that many determiners are ambiguous between a cardinal and a proportional reading. Often the ambiguity is resolved by the stress pattern – which is, however, not accessible in written text. If the ambiguous determiners are used in an unstressed variant they tend to be interpreted as cardinal quantifiers, if they are used in a stressed variant they get a proportional interpretation. Table 2 summarizes the proposed classification of deter-

	cardinal		**proportional**
precise	*a*, *no*		*every*, *the*, *each*, *all*, *both*, *neither*, *the seven*, *20 percent of the*
	← (unstressed)	*seven*, (*exactly*) *3*, …	→ (stressed)
vague			*most, more than half the, the majority of, less than seven*
	← (unstressed)	*several*, *some*, *many*, *few*, bare plurals	→ (stressed)

Table 2 Classification of Determiners in GQT

miners in GQT. In section 3.2.2.3 I will briefly come back to the observation that many determiners are systematically ambiguous between a cardinal and proportional reading

3.2.1.4 Monotonicity Properties of Generalized Quantifiers

A further important line of research within Generalized Quantifier Theory is the study of different mathematical properties of natural language determiners. Determiners show a regular behaviour across different domains and it is one topic of GQT to study these uniformities. Here I will only briefly address the role of determiners and quantifiers in inference. In particular, I will address how truth values of sentences are affected when the cardinality of the arguments is increased or decreased. This has become known as the *monotonicity* behaviour of a determiner. There are two relevant sets that can be increased or decreased, the CN-interpretation (here A) or the VP-interpretation (here B). If we, for example, have the information that some men walk, adding more walkers to the domain will not change the truth of this statement. If however we have the information that no man walks adding more walkers may well change the interpretation, because some of the walkers may turn out to be men. The following monotonicity properties are distinguished:

(19) **left monotone increasing** $(D\,AB \,\&\, A \subseteq A') \rightarrow D\,A'B$
e.g. If *some* valid cards disappear then *some* cards disappear.
Examples: *some*, *several*, *a few*, *at least 4*

right monotone increasing $(D\,AB \,\&\, B \subseteq B') \rightarrow D\,AB'$
e.g. If *every* card disappears quickly then *every* card disappears.
Examples: *some*, *every, each*, *several*, *a few*, *at least 4*

left monotone decreasing $(D\,AB \,\&\, A' \subseteq A) \rightarrow D\,A'B$
e.g. If *every* card disappears then *every* valid card disappears.
Examples: *every*, *each*, *at most 4*, *not more than 4*

right monotone decreasing $(D\,AB \,\&\, B' \subseteq B) \rightarrow D\,AB'$
e.g. If *no* card disappears then *no* card disappears quickly.
Examples: *no*, *at most 4*, *not more than 4*

Often the term 'increasing' alone is used for 'right monotone increasing', and the term 'decreasing' for 'right monotone decreasing'; the term 'persistent' is then used for 'left monotone increasing' and 'anti-persistent' for 'left monotone decreasing'. There are quantifiers that are neither increasing nor decreasing. For example, neither can (20)a be deduced from (20)b nor vice versa.

(20) a. Exactly three students were singing loudly.
b. Exactly three students were singing.

Other right non-monotone quantifiers are *an odd number of*, *between two and seven* etc. Examples for determiners that are not left-monotone are *most* and *the*.

For a number of natural language quantifiers the inferential properties are difficult to judge. This is especially true for vague, context-dependent and ambiguous determiners. Can we conclude from the fact that many valid cards disappeared the information that many cards disappeared? One reason why we cannot give a clear answer is that *many* is ambiguous between a proportional and cardinal reading. Furthermore, judgments are difficult when collective/distributive effects are taken into account. In particular the monotonicity properties of many collectively read quantifiers are not at all clear. Is the inference from the collectively read (21)a to (21)b really not valid as the non-monotonicity of *exactly n* in GQT would predict?

(21) a. Exactly three students shared an extra large pizza. They were very hungry.
b. Exactly three students shared a pizza.

Since intuitions about admissible inference patterns are not in all cases stable I suggest that the logical reconstruction should not be more precise than our intuition in the problematic cases. This implies a "modest" inference strategy, meaning that perhaps not all intuitively felt inference patterns can be reconstructed in all cases, but that the inferences that can be drawn definitely correspond to our intuitions. In a way one has to admit a certain incompleteness of the system but guarantee correctness.

3.2.1.5 Advantages and Limitations

The aim of Generalized Quantifier Theory is to give noun phrases a uniform compositional semantics. The formalism introduced is higher-order. This increased expressive power allows us to express the semantics of quantifiers like *more than half the* that are – at least for infinite domains – not first-order definable. However, a higher-order formalization is less suitable for practical applications that for example rely on first-order tools like automatic first-order theorem provers and model generators. A further advantage of GQT is that some generalizations on the inferential properties of non-standard quantifiers can be reconstructed. The relevant question in our context is whether GQT is appropriate to represent the semantics of plural noun phrases adequately.

The formalizations in Table 1 on page 38 above all relied on checking whether individual members of one set stand in a particular relation to individual members of another set. That means the formalizations only capture distributive properties, that is, properties that are true of individuals. As it stands, GQT is therefore not suitable to express collective readings of plural noun phrases. Without modifications the collective reading of *Some men gathered on the square.* can not be represented. Furthermore, the anaphoric potential of noun phrases like *many men*, *exactly 3 men* etc. is difficult to explain. Of course, these problems have not gone unnoticed and several extensions to GQT have been proposed to deal with collective readings. In section 3.3.2 I will present one such proposal by Scha (1981) who represents collective, cumulative and other readings of plural noun phrases within the framework of GQT.

Furthermore, as we will see in the next section, a uniformly quantificational analysis of all determiners leads to problems especially for definite and indefinite determiners. This point has been stressed by Löbner (1987). As an example he shows that definite and indefinite articles and quantifiers behave differently under negation.

(22) a. All/both/many/few men came to the party.
b. Not all/both/many/few men came to the party.
c. The/some men came to the party.
d. *Not the/some men came to the party.

What is more, Heim (1982, 1983) and Kamp (1984) observed that a uniform treatment of all types of quantifiers leads to inadequate predictions in so-called "donkey sentences" such as

(23) Every farmer who owns a donkey beats it.

If both noun phrases *every farmer* and *a donkey* are treated as generalized quantifiers we get as the denotation of the restrictor *farmer who owns a donkey* a set of objects such that there is a donkey that is owned by that object. Furthermore, as the denotation of the verb phrase *beats it* (assuming *it* is existentially bound to *a donkey*) a set of objects such that there is a donkey that is beaten by this object. The semantics of *every* then says that the former set is a subset of the latter. This, however, gives inadequate truth-conditions since the second set contains objects that beat any donkey not only the donkeys that they own. This would allow sentence (23) to be true in a situation where every donkey owning farmer beats only donkeys that he doesn't own. The problem is that the extensions of restrictor and scope are determined independently. The point is that in its original formulation GQT cannot adequately represent the semantics of donkey sentence. To solve problems concerned with donkey sentences a rather different approach to quantification was therefore proposed. In this approach definite and indefinite noun phrases do not have a quantificational force on their own. I will introduce this approach in the next section 3.2.2.

3.2.2 Discourse Semantic Approaches

3.2.2.1 Motivation

In the early 1980s Heim (1982, 1983) and Kamp (1984) were concerned with giving an adequate analysis of what they called "donkey sentences". An example was (23) here repeated as

(24) Every farmer who owns a donkey beats it.

Donkey sentences present at least two problems. One is concerned with the interpretation of the pronoun *it*. The other problem has been explained in the previous section 3.2.1.5 and concerns the claim of GQT that all noun phrases should receive a uniform interpretation and that this interpretation is independent of the context where the noun phrase occurs.

To explain the first problem we turn to representations in first-order logic. If the noun phrase *a donkey* is represented as an existential quantifier in first-order logic the pronoun *it* remains unbound:

(25) $\forall X([\text{farmer}(X) \wedge \exists Y(\text{donkey}(Y) \wedge \text{own}(X,Y))] \rightarrow \text{beat}(X,Y))$

In (25) the variable Y in the consequent is unbound.

If, on the other hand, we use a universal quantifier for *a donkey* and extend the scope of the universal quantifier as in

(26) $\forall X \forall Y([\text{farmer}(X) \wedge \text{donkey}(Y) \wedge \text{own}(X,Y)] \rightarrow \text{beat}(X,Y))$

we get an intuitively correct semantic representation, but now we have to explain how *a donkey* should sometimes be represented as a universal quantifier, whereas in other examples such as (27) it should be represented as an existential quantifier:

(27) John owns a donkey.

Kamp's Discourse Representation Theory (DRT) and Heim's File Change Semantics (FCS) solved this problem by viewing the meaning of a sentence in terms of the impact the utterance has on the discourse; this impact is often called the "context change potential" of an utterance. The approach suggests that indefinite NPs are not considered as quantificational on their own but that indefinite NPs are treated as introducing a new entity into the discourse and this entity is represented as a free variable. Definite NPs must – in a first approximation – relate to entities already introduced into the discourse. Semantic rules for interpreting the entire discourse then determine the interpretation of these free variables with the result that the variables are implicitly quantified by the sentential context. When the free variables occur in a simple sentence like (27) they receive an existential quantification (the discourse is true if there is a sequence of individuals for which all conditions in the discourse are true). This view also allows to represent pronouns that have to be bound across sentence boundaries (which could not be straightforwardly described in classical predicate logic):

(28) John owns a donkey. It is hungry.

If an indefinite noun phrase occurs in the scope of a quantifier like in (24) it is not bound by the discourse but by the quantifier itself. In discourse oriented semantics quantification breaks down into a tripartite structure, the quantificational operator, the restrictor and the scope. In a simplified approximation we can say that if the indefinite noun phrase occurs in the restrictor of a quantificational structure then it inherits the quantification of the operator that binds it. In (24) this leads – as desired – to a universally quantified interpretation of the indefinite noun phrase since it occurs in the restrictor of the quantifying operator *every*. We will discuss more technical details of discourse representation theory in chapter 4.

3.2.2.2 Individual Denoting vs. Quantificational Noun Phrases

Discourse related approaches abandon the uniform treatment of noun phrases as generalized quantifiers. Instead, two basic types of noun phrases are distinguished. On the one hand there are noun phrases like indefinites or definites that do not have a quantificational force on their own but are either bound by the entire discourse or, if they are under the scope of another operator, they inherit its quantificational force. On the other hand there are genuine quantificational noun phrases. These noun phrase introduce a quantificational element into the representation of the sentence in which they occur. Roberts (1987) called the two types of noun phrases *individual denoting* and *quantificational NPs*, respectively. Usually, only individual denoting plural noun phrases (e.g. *three men*, *the men*, *some men*) can get a collective reading. Individual denoting singular noun phrases (e.g. *a man*, *the man*) refer to atomic individuals and do not allow for collective readings. Quantificational noun phrases (e.g. *every man*) force a quantificational reading over the domain of atomic individuals. This quantificational force is intrinsic; therefore quantificational NPs never get a collective reading.

Individual denoting NPs are divided into definite and indefinite NPs. Definite NPs point out to an object or a group of objects relatively independently of the rest of the sentence. Among the definite NPs are proper nouns, personal pronouns, and common noun phrases preceded by the definite article *the*, demonstrative pronouns like *those*, possessive pronouns like *yours*, or NPs in the genitive, at least if the NP itself is definite. There are NPs where it is unclear whether they should be classified as definite or quantificational, like *all*. The determiner, *both*, is often taken as quantificational since it enforces a distributive reading. Definite plural noun phrases naturally allow for collective readings.

Indefinite noun phrases, like *two men*, also point at an object or a collection of objects, but they are less specific in determining which individual or collection is meant. An indefinite NP puts more or less constraints on the object it points out to, e.g. *two men* expresses a precisely specified cardinality, *several men* or *a few men* does not. Conjunctions of definite and indefinite NPs, e.g. *John and two customers*, can be read collectively as well. In general, indefinites are more likely to have distributive readings than definites.

It is not always clear which NPs should be classified as indefinite NPs and where to draw the borderline between indefinite and quantificational NPs. Some noun phrases are easily read collectively and easily classified as indefinite NPs, e.g. *a man and a woman*, *two men*, other noun phrases are easily classified as quantificational, e.g. *every/each/no man*. However, there is a large group of NPs which fall in between in that they are often interpreted distributively but under certain circumstances allow for a collective reading, *many/at least two/few/at most four/most/more than half the/at most half the men*. There are differences in how difficult it is to give these NPs a collective reading. According to Lønning (1987) the order of the introduction above corresponds roughly to an ordering from possible to impossible. A set of tests and criteria have been developed to classify noun phrases (e.g. Roberts 1987, pp. 194, Lønning 1987). I list a few of these tests.

Individual denoting noun phrases can serve as an antecedent for discourse anaphora, e.g.

(29) a. The/two/at least two/at most five/many men walk in the park. They are from London.
b. Few men lifted a piano. ?They were very strong.

Furthermore, individual denoting plural noun phrases can occur felicitously as the subject of a predicate with a floated quantifier like in

(30) a. The/some students/three/at least three students each lifted a table.
b. ?Exactly three/many/at most five students each lifted a table.

c. *Few students each lifted a table.

Another test says, that individual denoting plural noun phrases can occur together with collective predicates like *gather* or *meet*.

(31) a. Many/some/at most 50/exactly 25 students gathered in the university to protest.
b. ?Few men gathered to discuss the results.

Furthermore, individual denoting plural noun phrases occur with prepositions such as *among* and *between* that seem to take a group denoting complement.

(32) a. John found a Visacard among some credit cards.
b. *John found a Visacard among few credit cards.

The tests given above lead Roberts (1987) to the classification of determiners given in Table 3.

Individual Denoting	**Quantificational**
a	each
$\text{some}_{sg/pl}$	every
1,2,3,…	$\text{no}_{sg/pl}$
$\text{the}_{sg/pl}$	most
this, that	few, many
these, those	both, neither

Table 3 Robert's (1987) classification of determiners

Implicit in Robert's classification is the hypothesis that determiners are unambiguous. There are a few cases, for examples numerals and *many* where this hypothesis is questionable. I will come back to this issue in the next section 3.2.2.3. Related to this problem is that Robert's classification has the disadvantage that noun phrases with *many*, *few* etc. do not naturally get a collective reading and cannot serve as discourse anaphora. I will come back to this problem in sections 3.3.2.2 and 3.3.2.3 below.

Kamp and Reyle (1993) therefore propose a slightly different classification of noun phrases. They take up a proposal of Partee (1989) and distinguish *proportional quantifiers* from *cardinality quantifiers*. In contrast to proportional quantifiers and definites, cardinality quantifiers and indefinites allow for "existential *there*-insertion". Constructions with NP-final *each* also support this distinction. The following examples are taken from Kamp and Reyle (1993, pp. 452). Examples for the behaviour noun phrases in existential *there* insertions are:

(33) a. There is an/one apple in the basket.
b. There are two/several/many/few/at least two/at most three apples in the basket.
c. *There is the/this/every apple in the basket.
d. *There are all/most apples in the basket.

Example for the behaviour in NP-final *each*-constructions are:

(34) a. The boys received two/several apples each.
b. The boys received few/at least two/at most three apples each.
c. *The board members read all/most application dossiers each.

The semantic function of *there*-insertion sentences is to assert that an individual, or group of individuals, with certain specified properties exists. The floated quantifier *each* presupposes an existing object over which distribution takes place. Both observations indicate that – like indefinite noun phrases – cardinality noun phrases have to introduce a discourse referent into the domain of discourse. In contrast to proportional quantifiers like *every* or *each* cardinality quantifiers make assertions about the size of a group (e.g. whether it consists of no element, one element, at least three elements or at most four elements, etc.). To capture the meaning of cardinality quantifiers and to explain observations with discourse anaphora Kamp and Reyle introduce the technique of antecedens construal by abstraction. I will briefly explain the problems of this approach below in 3.3.3.2.

3.2.2.3 Quantificational vs. Non-Quantificational Use of Indefinites

As discussed earlier, in the GQT tradition an ambiguity of noun phrases between a cardinal and a proportional reading has been observed. Also in the tradition of discourse related approaches it has been argued that indefinites and other individual denoting noun phrase are systematically ambiguous between a quantificational and a non-quantificational use.

A referential approach to the interpretation of indefinites introduces new objects into the representation. In discourse semantic approaches these objects correspond to free variables, or discourse referents. In a predicate logic reconstruction these objects correspond to existentially quantified variables – if they are not bound by other operators. The semantic content of indefinite elements (numerals, *some*, *several* etc.) is seen as providing additional constraints on possible assignments to this variable. That means the indefinite elements occur like adjectives with an intersective meaning. In case of the noun phrase *four men* the set of men is intersected with sets of objects consisting of four members. The interpretation where indefinites are taken as mere cardinality attributes, i.e. as one-place first-order predicates, is incompatible with a quantificational approach where numerals are two-place second order predicates. Löbner (1987) explains that this incompatibility could be traced to the fact that indefinites are ambiguous in that they can be taken either as cardinality predicates or as quantifiers proper. This observation was also stated by Milsark (1977). One example is

(35) Some salesmen walked in.

Milsark argues that *some* is ambiguous between an unstressed variant *sm* and a stressed form *some*, the former being a (vague) cardinality predicate, the latter a quantifier proper. Under the non-quantificational reading the sentence means that something walked in, namely salesmen, in fact some. This interpretation would match the interpretation of *some men* as an individual denoting noun phrase, i.e. as introducing a discourse referent or an existential quantifier, respectively. In the second reading, some out of a certain set of salesmen under consideration walked in. This interpretation is quantificational in the sense of GQ theory, and cannot be adequately handled by a referential interpretation. The ambiguity is associated with several differences. The quantificational interpretation requires that a set of salesmen is already introduced in the context or else independently determined, otherwise the partitive reading (some of the salesmen) would not be available. On the non-quantificational reading the indefinite NP introduces a new object. This interpretation arises in different contexts. The following example by Löbner (here slightly modified) creates a context where *some* has a non-quantificational reading:

(36) When I entered the store, nobody was there. After a couple of minutes, some salesmen walked in.

In contrast, a quantificational reading is the most plausible in the following context.

(37) When I entered the store, nobody was there. As usual the salesmen meet in the cafeteria at lunchtime. At 1 o'clock, some salesmen walked in, but the rest stayed in the cafeteria.

Löbner (1987, p. 192) gives an extensive list of differences which I summarize in Table 4.

	non-quantificational		quantificational
a.	Head noun referentially new.		Head noun referentially given.
b.	Partitive paraphrase impossible.		Partitive paraphrase possible.
	–	*Some of the salesmen walked in.*	+
c.	Prenoun counts the whole denotation of the head noun.		Prenoun counts part of the denotation of the head noun.
d.	Subsequent anaphora refers to the whole denotation of the head noun.		Subsequent anaphora refers to part of the denotation of the head noun.
e.	Head noun stressed, obligatory.		Head noun unstressed, omissible.
	–	*Some walked in.*	+
f.	Prenoun unstressed, omissible.		Prenoun stressed, obligatory.
	+	*Salesmen walked in.*	–
g.	VP stressed or unstressed.		VP stressed.

Table 4 Quantificational vs. Non-Quantificational Use of Indefinites

Löbner states that the non-quantificational readings of indefinites are by far the more frequent ones. A count of about 1000 subsequent occurrences of indefinites in five different tests rendered a portion of quantificational ones of less than 10 per cent, and most of them, in fact, were explicit partitives.

Assuming two different semantic representations for the referential and the quantificational use of indefinites would result in a further explosion of ambiguity and should therefore be avoided. I will discuss possible pragmatic solutions to this problem in section 3.3.3.2.

3.2.2.4 Advantages and Limitations

Discourse related approaches allow for a straightforward representation of collective readings – as will be further detailed in section 3.3.3. Furthermore, discourse related approaches solve problems occurring with donkey sentences and also solve the problem of cross-sentential anaphora. Furthermore, the anaphoric potential of noun phrases like *at least two men* or *exactly 3 men* can be solved if these noun phrases are assumed to introduce discourse referents. However, problems occur if noun phrases are assumed to be unambiguous, i.e. either individual denoting or quantificational. Then one runs into problems with the systematic ambiguity of indefinites between a quantificational and a non-quantificational reading. For example, the "exactly"-effect of numerals in certain contexts cannot be easily explained. Furthermore, some authors have expressed concern about the lack of compositionality in the DRT and FCS approaches. Instead they propose dynamic semantics – a modification of traditional predicate logic – which interprets donkey sentences and discourse anaphora correctly (Groenendijk and Stokhof 1991).

3.3 Strategies to Represent Plural Ambiguities

3.3.1 Overview

Taking the above introduced central concepts of noun phrase semantics as a background the next sections will give an overview of important existing approaches within formal semantics to represent the different readings associated with plural noun phrases. This look into the literature on (formal) plural semantics is also helpful to estimate the approach that will be developed in this thesis.

The existing strategies to deal with plurals and their associated ambiguities can be distinguished according to the following criteria:

- Where does the ambiguity come from (NP, VP, elsewhere)?
- How many readings are assumed (collective, distributive, cumulative, other readings)?
- What is the underlying ontology (set theoretic or algebraic modelling)?
- How are the readings represented (first-order, higher-order)?
- How suitable is the approach for computational applications?

I will introduce two "prototypical" approaches that were of great influence for the development of plural semantics: Scha (1981) and Link (1983, 1991). Scha developed his proposal in the tradition of Generalized Quantifier Theory (GQT) according to which all noun phrases are semantically treated as quantifiers. Link's proposal on the other hand follows a tradition according to which there are two classes of noun phrases that get a semantically different treatment, viz. quantificational vs. individual denoting noun phrases. Related to these different traditions is that Scha and Link locate the source for the ambiguity of plural sentences differently.

According to Scha the ambiguity between collective, distributive and possibly other readings is located in the plural noun phrase or more precisely in the determiner (*Noun Phrase Strategy*). According to Link noun phrases are unambiguous and the readings should be generated within the verb phrase by (c-)overt modification (*Verb Phrase Strategy*). I will furthermore mention an approach by Verkuyl and van der Does (1991) according to which one should grant sentences a single weakest meaning which encompasses the others (*No Ambiguity Strategy*). Although Verkuyl and van der Does discovered later that their approach cannot be sustained their ideas are interesting in that they try to drastically reduce the number of postulated ambiguities. A fourth strategy (*Global Strategy*) combines some of the other strategies, in that the readings of complex sentences are a result of the whole structure or as Roberts (1987, p. 100) puts it: "Distributivity is a property of predications, combinations of a subject and a predicate." The readings can be triggered by different elements of a sentence, there is a functional interplay between the different categories. A similar strategy is also pursued by vanþder Does (1992). I will argue in chapters 4 and 5 of this thesis that this is the preferred strategy of my approach although the cited proponents have not given a computationally suitable formalization of their ideas.

3.3.2 The Noun Phrase Strategy: Scha (1981)

3.3.2.1 Basic Ideas

The distinctive features of Scha's approach are: (i) the ambiguity is located in the noun phrases, i.e. noun phrases are ambiguous, (ii) noun phrases have three readings, a distributive, a collective and a neutral reading, additionally a cumulative reading for sentences with multiple plurals is postulated, (iii) a set-theoretic modelling is assumed, meaning that (iv) readings are represented in a higher-order language, which is (v) not immediately suitable for practical applications. I will comment on these features in the sequel.

Set-Theoretic Modelling of Plurals

Scha developed his theory in the tradition of Generalized Quantifier Theory. The generalized quantifiers introduced in section 3.2.1 quantify only over the domain of individuals. The existence of collective readings shows, however, that there are noun phrases that cannot be reduced to quantifiers over the individual domain. Several modifications and extensions to the traditional GQT treatment of noun phrases are thus necessary to handle plural phenomena. First of all, denotations for collections, i.e. groups of objects, have to be found. Scha and others proposed to use sets for this purpose. The domain E (a set of individuals) of a model is replaced by its powerset $\wp(E)$ (a set of sets). The objects talked about are now sets. Properties become properties of sets instead of properties of individuals. Furthermore, a uniform treatment of sets and individuals within the VP denotation requires a uniform ontological type which means that individuals in the verb phrase denotation are no longer represented as elements of E but are as singletons, i.e. sets containing just the individual as an element. Under this perspective a determiner denotes a relation between a set (the noun denotation) and a set of sets (the verb phrase denotation). As in the singular fragment the noun denotation is a set of individuals. The verb phrase, however, is now seen as a property of collections, i.e. it denotes a set of sets. This allows to represent the collective reading as predication of a property to a collection. The collective reading of

(38) Two men lifted a table.

can be represented as

(39) $\exists Z(Z \subseteq [[man]] \wedge |Z| = 2 \wedge [[lift\ a\ table]](Z))$

Formula (39) expresses that there is a collection of two men that lifts a table. Using λ-notation the collective meaning of a numeral *n* can be given as:

(40) $\lambda X \lambda \boldsymbol{Y}.\exists Z(Z \subseteq X\text{: } [|Z| = n \wedge \boldsymbol{Y}(Z)])$

The bold-face letter indicates that $\boldsymbol{Y}$ is of a different type than X. Whereas X is a set of individuals, $\boldsymbol{Y}$ is a set of sets.

The problem now is how to get the distributive readings, i.e. how to assign the property to atoms. Technically, there is some option here depending on where the source for the distributivity is located. In Scha (1981) the source for the distributivity is located in the noun phrase or more precisely in the determiner. Thus in Scha's setting there are e.g. collective and distributive interpretations of numerals. How this is technically realized will be discussed in the next sections.

Noun Phrases are Ambiguous

According to Scha, the collective-distributive ambiguity resides inside the noun phrase, not in the predicate or the whole sentence. More precisely, Scha attributes the ambiguity to lexical features of determiners. Numerals, the null determiner, *all*, plural *some* are ambiguous between three types of readings: a distributive (D), a collective (C) and a neutral (N) one. The distributive reading means we quantify over atomic individuals, the collective reading means that we quantify over collections and the neutral reading expresses that we quantify over objects that take part in certain collections. In the neutral reading we remain open as to the precise structure and size of these collections. Only the plural definite article is unambiguously forcing a collective reading. Meaning postulates for verbs are used in this case to derive other readings – which is problematic since the meaning postulates must be optional to generate both, collective and distributive readings of mixed predicates (see Roberts 1987 for a discussion).

Representation of Readings

In the following I will mainly show how the readings of the two-place plural sentence

(41) Four men lifted three tables

is represented in Scha's approach. Scha proposes that an ambiguous determiner like *(exactly) n* can have the following three denotations. Note that for ease of readability, variations of Scha's original notation and abbreviations have been chosen to represent the readings (see also van der Does 1993).

(42) D $\lambda X \lambda \boldsymbol{Y}.|\{d \in X: \boldsymbol{Y}(\{d\})\}| = n$
C $\lambda X \lambda \boldsymbol{Y}.\exists Z(Z \subseteq X: [|Z| = n \wedge \boldsymbol{Y}(Z)])$
N $\lambda X \lambda \boldsymbol{Y}.|\cup\{Y \subseteq X: \boldsymbol{Y}(Y)\}| = n$

We give some examples. The reading where both subject and object noun phrase receive a distributive reading (abbreviated as DD) amounts to:

(43) DD
$|\{d \in [[man]]: |\{d' \in [[table]]: [[lift]](\{d\})(\{d'\})\}| = 3\}| = 4$

This reading expresses that the number of men each of which lifted each of three possibly different tables equals 4. The reading does not allow that there are more than four table lifting men, i.e. it expresses the "exactly"-reading of numerals. The reading allows for three to up to twelve different tables.

As a further example we give the NN interpretation of (41):

(44) NN
$|\cup\{X \subseteq [[man]]: |\cup\{Y \subseteq [[table]]: [[lift]](X)(Y)\}| = 3\}| = 4$

This reading expresses that there are four men that can be divided into subcollections such that for each subcollection M there are three tables from which subcollections can be formed that are each lifted by M. Here again the cardinality of the men may not exceed 4.

The doubly collective reading (CC) of (41) yields

(45) CC
$\exists X(X \subseteq [[man]]: |X| = 4 \wedge \exists Y(Y \subseteq [[table]]: |Y| = 3 \wedge [[lift]](X)(Y)))$

Formula (45) expresses that there is a collection of four men that lifted a collection of three tables. Due to the existential quantifier this reading allows for other groups of men or individual men who lift tables, i.e. it expresses the "at least"-reading of numerals.

For ease of readability we can shorten the notation $\exists X(X \subseteq [[man]]: |X| = 4 \wedge \Phi)$ to $\exists X \in [[four\ man]]: \Phi)$ and write (45) as

(46) CC (notational variant)
$\exists X \in [[four\ man]]\ \exists Y \in [[three\ table]]: [[lift]](X)(Y)$

In the next subsection we will see that for two-place sentences Scha assumes yet another reading, the cumulative reading.

How Many Readings?

The ambiguity of numerals between a C, D and N reading predicts that there are nine different readings for (41) since each of the two noun phrases can get a collective, distributive or neutral reading. If additionally scope ambiguities are considered the readings multiply by two yielding eighteen readings for sentence (41). For numerals some of the readings that Scha predicts turn out to be redundant but this need not be the case in general. Scha argued furthermore that sentences with multiple plurals have – apart from the above introduced collective, distributive and neutral readings – additionally a *cumulative reading*. The cumulative reading of sentence (41) states that the number of men who lift a table is four and the number of tables that are lifted by men is three. According to the cumulative reading the interpretation of the subject and the object noun phrase are mutually independent, i.e. the noun phrases do not have scope over each other. In Scha's setting the cumulative reading cannot be obtained from any combination of collective, distributive or neutral readings of the two noun phrases. Scha's original example to argue for the cumulative reading was (47)a which can be paraphrased as (47)b.

(47) a. 600 Dutch firms have 5000 American computers.
b. The number of Dutch firms which have an American computer is 600, and the number of American computers possessed by a Dutch firm is 5000.

Scha considers only distributive variants of the cumulative reading, i.e. single Dutch firms own single American computers. However, (47)a can also be true if groups of firms own groups of computers. A cumulative reading is accordingly possible for (41). The formalisation of the more general cumulative reading of (41) can be given by the formula (48).

(48) $|\cup\{X \subseteq [[man]] : \exists Y(Y \subseteq [[table]] : [[lift]](X)(Y))\}| = 4 \wedge$
$|\cup\{Y \subseteq [[table]] : \exists X(X \subseteq [[man]] : [[lift]](X)(Y))\}| = 3$

The formula expresses that we count all table lifting men (no matter whether the men act alone or together with others) and we count tables lifted by men (whether the tables were lifted alone or together with other tables). We are just interested in cardinalities of men and table and not in the precise lifting constellation. This cumulative reading often occurs in sentences with large numbers. The technical problem with this interpretation is that its structure does not immediately suggest a way to derive it (compositionally) from the surface syntactic structure of (41) since the noun phrases have to be interpreted independently of each other.

With his N reading of numerals in (42) Scha comes close to the neutral reading considered by Link (1983a) who suggested that sentences with large numerals often have an intermediate reading which is neither distributive nor collective. The sentence

(49) Half a million children gathered throughout the country.

can be true in a situation where the children did not gather as a whole, but there could be many subgatherings of collections of children that may or may not overlap. However, Scha's N reading requires that only groups consisting solely of children gathered, which may be too strict since the sentence probably remains true when some adults joined the children. To remedy this intuition van der Does (1992, pp. 34) proposed two additional neutral interpretations that express that a certain group only *participates* in the action to be described:

(50) N2g $\lambda X \lambda \boldsymbol{Y}.\exists Z(Z \subseteq X\text{: } [|Z| = n \wedge Z \subseteq \cup \boldsymbol{Y}]$
N2 $\lambda X \lambda \boldsymbol{Y}.|\cup\{Z \subseteq X\text{: } Z \subseteq \cup \boldsymbol{Y}\}| = n$

Note, that neither reading restricts the VP denotation ***Y*** e.g. only to collections of children. On both readings adults may have joined children and gathered with them. The difference is that on the N2g reading – due to the existential quantifier – there could have been more than *n* children gathering whereas on the N2 reading the amount of children gathering has to exactly equal *n*. Take for example a situation where children are denoted by c1 to c8, men are denoted by m1 to m4 and *gather* has a denotation given as follows:

(51) $[[children]] = X = \{c1,c2,c3,c4,c5,c6,c7,c8\}$,
$[[men]] = \{m1,m2,m3,m4\}$,
$[[gather]] = \boldsymbol{Y} = \{\{c1,c2,m1\},\{c3,c4,c5,m2,m3\},\{c6,m4\}\}$.

In this situation the sentence

(52) Five children gathered.

is predicted to be false in Scha's N reading since adults joined the children. The sentence is true in the N2g reading, since there is a group $Z = \{c1,c2,c3,c4,c5\}$ such that $Z \subseteq X$ and $|Z| = 5$ and, since $\cup\boldsymbol{Y} = \{c1,c2,c3,c4,c5,c6,m1,m2,m3,m4\}$, it is true that $Z \subseteq \cup\boldsymbol{Y}$. Sentence (52) is, however, not true on the N2 reading since $\cup\{Z \subseteq X: Z \subseteq \cup\boldsymbol{Y}\} = \{c1,c2,c3,c4,c5,c6\}$ and the cardinality of this set equals 6, not 5.

3.3.2.2 Problems and Possible Solutions

Scha's noun phrase centred approach generates collective, distributive, neutral and cumulative readings of plural noun phrases. Apart from the problem of a combinatorial explosion of readings two main objections to this noun phrase centred approach have been raised.

Coordinate VP Structures ("Dowty-Sentences")

Dowty (1986) mentions sentences like

(53) a. Four men went to the bar together and had a beer each.
b. John and Mary won a lottery drawing and then developed insomnia worrying about the money. (Roberts 1987, p. 122)

where one part of the predicate is true of the whole collection (collective reading) whereas the other part of the predicate is true of each individual member of the collection (distributive reading). For example in (53)b the preferred reading is that John and Mary jointly won a lottery drawing, then they each developed insomnia. If NPs have different denotations in their collective vs. distributive interpretation – as is assumed within the NP-strategy – then what do these NPs denote? There is no straightforward way in which this observation can be handled in terms of determiner denotations since for different verbs different determiner denotations for the same occurrence of a determiner would have to be chosen. Examples of this type have provided the primary motivation for verb phrase centred accounts of collection and distribution, as I will discuss in the next section. In Dowty's view coordinate VP structures even constitute a "knock-down" argument against noun phrases centred approaches. This objection can be countered by the view that what is involved in mixed cases of conjunction is a kind of an elided anaphoric element in the second conjunct which could be interpreted either collectively or distributively independently of its antecedent (see van der Does 1992, pp. 83).

Anaphoric Potential of Plural Noun Phrases

A related problem of noun phrase centred approaches concerns the treatment of discourse anaphora. The treatment of distributive numerals as (42), reading D, implies that all that counts is the number of objects that are involved in a certain relation expressed by a sentence. This analysis however doesn't bring out the fact that a cardinal plural noun phrase can and in most cases will be used referentially when a speaker has a particular group of four men in mind as in (53)a, no matter whether there were other men ordering beer beside that. This referential use is obvious when the sentence (53)a is continued with an anaphoric pronoun (here *they*) that refers back to the group of four men:

(54) They talked about the football match.

To explain the anaphoric potential of distributively interpreted NPs with determiners like *exactly three* or *at most two* Kamp and Reyle (1993, pp. 452) propose the so-called *antecedent construal by abstraction*. Informally, a quantificational scheme *at most n X are Y* would be taken up by the pluralic abstraction term *those X that are Y*. That means the quantificational structure triggers the introduction of a new plural discourse entity *Z* that is identified with the sum of all the individual *x* that satisfy the union of the restrictor and the scope. Krifka (1996) criticizes this operation to construct discourse referents as antecedents. He sees it as quite *ad hoc* and what is more he addresses the problem when to introduce the new discourse referent *Z*. Either it is introduced whenever a quantificational structure is introduced, since *Z* may be needed later which is seen by Krifka as a “generalization to the worst case” (Krifka 1996, p. 559) and which is neither attractive from a theoretical nor a practical point of view. The second option is to introduce *Z* at the time when it is needed, e.g. when a pronoun occurs and an antecedens has to be found. But his involves reprocessing previous expressions and this again is not satisfactory to explain human text processing and definitely not suitable for automatic text processing. A further criticism concerns the fact that the rule of abstraction is much too powerful and not adequately constrained. For further details of the criticism I refer the interested reader to Krifka's article. What I would like to add as a problem is that antecedens construal by abstraction does not explain the occurrence of collective readings with determiners like *exactly three*, or *at most two* since the abstraction operation only abstracts a discourse referent from a distributive reading of the sentence.

3.3.2.3 Evaluation

The advantage of the noun phrase approach is that most readings can be represented by introducing appropriate readings for the determiners. Furthermore, the fine-grained analysis forces us to think about different possible situations described by plural sentences. However, the disadvantage is that the flexibility of the NP approach leads to a massive overgeneration of readings. Does it make sense to assume 10 or more readings for sentences like (41)? An approach that generates so many readings is not only theoretically implausible but it is also highly unsuitable for natural language processing systems. To restrict a combinatorial explosion one would have to find constraints that restrict the admissibility of the readings. These constraints are not only hard to find theoretically and but also difficult to integrate into natural language processing systems. Scha does not propose such a theory of disambiguation. Scha's approach also has a technical disadvantage. It relies on quantification over sets, i.e. it produces formulae of second-order predicate logic that the first-order inference tools I am going to use will not be able to process. A further disadvantage is that the set-theoretic modelling does not easily generalize to the mass domain as will be shortly discussed in the next section.

3.3.3 The Verb Phrase Strategy: Link (1983)

3.3.3.1 Basic Ideas

The distinctive features of Link's (1983, 1991) theory are: (i) the ambiguity is located in the verb phrase, whereas noun phrases are unambiguous. There are two types of noun phrases: individual denoting and quantificational NPs – only individual denoting noun phrases allow for a collective reading; (ii) Link distinguishes collective and distributive readings, cumulative readings are subsumed under the collective reading, (iii) Link proposes an algebraic approach where the domain has a lattice-structure; (iv) this allows Link – in principle – to represent plurals in a first-order language, (v) which makes it more suitable for computational applications. Link's approach assumes less readings than Scha's, nevertheless the approach also suffers from a combinatorial explosion of readings. A main criticism towards Link's approach says that he fails to explain the readings of non-monotone increasing cardinality quantifiers like *exactly n*.

Algebraic Modelling of the Domain

To accomplish that individual denoting plural noun phrases can refer directly to objects in the domain of interpretation Link uses a domain that has a lattice structure. A simple lattice structure which has the three atomic elements a, b and c can be illustrated by Figure 2.

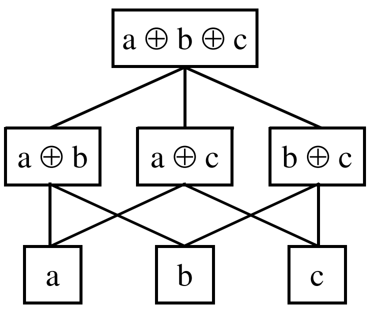


Figure 2 Lattice Structure for Atomic Objects a, b and c

The term a ⊕ b denotes the individual sum (i-sum for short) consisting of the atomic objects a and b. Ontologically, this individual sum is of the same type as its constituting atomic members; it is an object in its own right. While atomic objects can be referred to by individual denoting singular noun phrases like *a man*, i-sums can be referred to by an individual denoting plural noun phrases like *three men*. The individual sums in the lattice are partially ordered by a part-whole relation which is a reflexive, anti-symmetric and transitive relation. In Figure 2 this relation is indicated by the lines to be read from bottom to top as “is a part of” (symbolically $\leq$). For a formalization of a lattice-structure see section 4.4.4. In Link’s system one-place predicates P can be turned into pluralized predicates by the *-operator yielding $*P$. The extension of the starred predicate $*P$ is the set of all i-sums constructed on the basis of the extension of the non-starred predicate P, i.e. on the basis of the atomic individuals a, b, c. If for example [[*table*]] = {a, b, c} then [[**table*]] = {a, b, c, a ⊕ b, a ⊕ c, b ⊕ c, a ⊕ b ⊕ c}. The extension of the pluralized predicate $*P$ has a uniquely specified supremum (least upper bound), in this case the i-sum a ⊕ b ⊕ c. Intuitively, the supremum of some set X ordered by the part-of relation is the smallest thing in X such that all other elements of X are a (proper) part of the supremum.

In models with more than three atomic tables, the object a ⊕ b ⊕ c would be just one of the possible three-membered i-sums. In our model the i-sum a ⊕ b ⊕ c is the denotation of the individual denoting plural noun phrase *three tables*. The denotation of the noun phrase *two tables* could be any of the i-sums a ⊕ b, a ⊕ c or b ⊕ c.

The assumption of a lattice structured domain has the advantage that the analogy between the count domain and the mass domain can be straightforwardly described. The semantics of mass nouns like *water*, *gold*, *sugar* is similar to the semantics of plural nouns. The sum of two mass expressions is again a mass expression and not an entity of a different category. Link (1983) suggests the following examples to illustrate the analogy:

(55) a. If a is water and b is water then the sum of a and b is water.
b. If the animals in this camp are horses, and the animals in that camp are horses, then the animals in both camps are horses.

The common property of mass and plural count nouns is often called the “cumulative reference property”: if a predicate P applies to two expressions then P also applies to the sum of the two expressions. The cumulative reference property applies to plural as well as mass nouns. The only difference between the count and the mass domain is that the latter does not contain atomic bottom elements. Mass nouns do not have minimal parts (at least not in our conceptualization of the world).

Verb Phrases Are Ambiguous between a Collective and a Distributive Reading

In the approach developed by Link (1983, 1991) noun phrases are not ambiguous between a collective and a distributive interpretation. Every noun phrase gets just one interpretation. The ambiguity is triggered by the verb phrase. Collective readings are only possible in combination with individual denoting (referential) plural noun phrases (e.g. *three men*, *the men*, *some men*). The collective reading can be represented in the traditional way as predication of the verb phrase denotation over the noun phrase denotation without changing the type of the predicate extension. For example, the collective reading of

(56) John lifted three tables

can be represented as

(57) $\exists X(*table(X) \wedge |X| = 3 \wedge lift(j,X))$

Here the one place functional cardinality symbol '$|\cdot|$'means that the atomic parts of the i-sum X are counted. The existential quantifier simulates the referential interpretation of the noun phrase *three tables*.

To obtain a distributive interpretation Link needs an operation to "get at" the atomic individuals of the i-sum denoted by *three tables*. To do this Link introduces a distributivity operator **D** which can be attached to one-place predicates, here *P*, and allows one to quantify over atomic individuals. The distributivity operator $^{\mathbf{D}}$ is defined as (58). In the formula the symbol $\leq_i$ expresses the individual-part-of-relation:

(58) $^{\mathbf{D}}P := \lambda X.\forall Y[Y \leq_i X \rightarrow P(Y)]$

The distributivity operator may also apply to a verb phrase derived by λ-abstraction, so that e.g. direct and indirect objects of verb phrases can be interpreted distributively. The logical form of the distributive reading of (56) can be given as:

(59) $\exists X(*table(X) \wedge |X| = 3 \wedge {}^{\mathbf{D}}\lambda Z.lift(j,Z)(X))$
$\Leftrightarrow \exists X(*table(X) \wedge |X| = 3 \wedge \forall Y[Y \leq_i X \rightarrow lift(j,Y)])$

For ease of readability, distributivity operators $^{\bullet\mathbf{D}}$ and $^{\mathbf{D}\bullet}$ are defined (Link 1991) that operate on two place predicates, here *Q*, and indicate which argument is distributed over. The operator $^{\bullet\mathbf{D}}$ makes a two-place predicate distributive in its second argument and leaves the first argument unaffected. $^{\bullet\mathbf{D}}Q$ is defined as

(60) $^{\bullet\mathbf{D}}Q := \lambda V\lambda W.\forall U[U \leq_i W \rightarrow Q(V, U)]$

The notation $^{\mathbf{D}\bullet}Q$ means that *Q* is only distributive in its first argument place. $^{\mathbf{D}\bullet}Q$ is defined as:

(61) $^{\mathbf{D}\bullet}Q := \lambda V\lambda W.\forall U[U \leq_i V \rightarrow Q(U,W)]$

Link argues that in sentence (41) here repeated as

(62) Four men lifted three tables.

each argument place can be read collectively or distributively amounting to four readings. Furthermore, the distributivity operator – implicitly introducing universal quantification – can interact in scope with other quantifiers thus multiplying the readings for (62) by two. Since for the doubly collective reading scope makes no difference Link gets altogether seven readings for (62). These readings correspond to the scoped readings on the left part of Figure 1 on page 18. The cumulative and the neutral reading that were introduced in Scha's approach as a separate reading are subsumed under the collective reading.

Representation of Readings

Following are the logical forms for the readings that Links assumes for (62). The notation D_MC_T means that the noun phrase *four men* (M) has wide scope and is interpreted distributively (D) whereas the noun phrase *three tables* (T) has narrow scope and is interpreted collectively (C). Accordingly, the notation D_TC_M means that *three tables* is interpreted distributively and has wide scope over the collectively interpreted noun phrase *four men*. For easier comparison, the labels (R1), (R2) etc. correspond to the labels in Figure 1 on page 18.

The doubly collective reading $C_MC_T = C_TC_M$ expresses that there is a group of four men and a group of three tables that is lifted by the men:

(63) $C_MC_T = C_TC_M$ (R1, R5)

$$\exists X \exists Y({*man}(X) \wedge |X| = 4 \wedge {*table}(Y) \wedge |Y| = 3 \wedge lift(X,Y))$$

The reading C_MD_T expresses that each table is related individually to the group of four men:

(64) C_MD_T (R2)

$$\exists X \exists Y({*man}(X) \wedge |X| = 4 \wedge {*table}(Y) \wedge |Y| = 3 \wedge {}^{\bullet\mathbf{D}}lift(X,Y))$$
$$\Leftrightarrow \exists X \exists Y({*man}(X) \wedge |X| = 4 \wedge {*table}(Y) \wedge |Y| = 3 \wedge \forall U[U \leq_i Y \rightarrow lift(X, U)])$$

In reading D_MC_T each individual man relates to three (possibly different) tables allowing for up to twelve different tables that are lifted. This interpretation could also represent the situation in which the same three tables were lifted by each of the men on different occasions.

(65) D_MC_T (R3)

$$\exists X({*man}(X) \wedge |X| = 4 \wedge {}^{\mathbf{D}}\lambda U.\exists Y[{*table}(Y) \wedge |Y| = 3 \wedge lift(U,Y)](X))$$
$$\Leftrightarrow \exists X({*man}(X) \wedge |X| = 4 \wedge \forall U[U \leq_i X \rightarrow \exists Y({*table}(Y) \wedge |Y| = 3 \wedge lift(U,Y))])$$

The reading D_MD_T expresses that each of the three men lifted each of three (possibly different) tables. Intuitively, this means if there are only three tables each of the three men lifts the tables at different locations. If there are twelve tables each of the three men can lift "his" three tables in the same interval as the other men.

(66) D_MD_T (R4)

$$\exists X({*man}(X) \wedge |X| = 4 \wedge {}^{\mathbf{D}}\lambda U.\exists Y[{*table}(Y) \wedge |Y| = 3 \wedge {}^{\bullet\mathbf{D}}lift(U,Y)])$$
$$\Leftrightarrow \exists X({*man}(X) \wedge |X| = 4 \wedge \forall U[U \leq_i X \rightarrow \exists Y({*table}(Y) \wedge |Y| = 3 \wedge \forall V[V \leq_i Y \rightarrow lift(U,V)])])$$

The reading C_TD_M expresses that there is a group of three tables such that each of the three men lifted that group of tables

(67) C_TD_M (R6)

$$\exists Y \exists X({*table}(Y) \wedge |Y| = 3 \wedge {*man}(X) \wedge |X| = 4 \wedge {}^{\mathbf{D}\bullet}lift(X,Y))$$
$$\Leftrightarrow \exists Y \exists X({*table}(Y) \wedge |Y| = 3 \wedge {*man}(X) \wedge |X| = 4 \wedge \forall U[U \leq_i X \rightarrow lift(U,Y)])$$

The reading D_TC_M expresses that there is a group of three tables such that for each atomic member of the group of tables there is a group of four men who together lift the table.

(68) $D_T C_M$ (R7)

$\exists Y(*table(Y) \wedge |Y| = 3 \wedge {}^{\mathbf{D}}\lambda U.\exists X[*man(X) \wedge |X| = 4 \wedge lift(X,U)](Y))$

$\Leftrightarrow \exists Y(*table(Y) \wedge |Y| = 3 \wedge \forall U(U \leq_i Y \rightarrow \exists X[*man(X) \wedge |X| = 4 \wedge lift(X,U)]))$

Finally, the reading $D_M D_T$ expresses that for each atomic member of a group of four tables there are three (possibly different) men who each lift the table. The number of men can vary between 3 and 12, i.e. there can be 12 different liftings.

(69) $D_T D_M$ (R8)

$\exists Y(*table(Y) \wedge |Y| = 3 \wedge {}^{\mathbf{D}}\lambda W \exists X[*man(X) \wedge |X| = 4 \wedge {}^{\mathbf{D}\bullet}lift(X,W)](Y))$

$\Leftrightarrow \exists Y(*table(Y) \wedge |Y| = 3 \wedge \forall V[V \leq_i Y \rightarrow \exists X(*man(X) \wedge |X| = 4 \wedge$
$\forall U[U \leq_i X \rightarrow lift(U,V)])])$

3.3.3.2 Problems and Possible Solutions

The approaches of Scha and Link crucially differ in the treatment of distributive interpretations of numerals and indefinites. In Link's approach indefinites are treated as unequivocally individual denoting whereas Scha treats indefinites as ambiguous between three readings. Van der Does (1992) argues that Link's approach fails to explain data concerned with non-monotone increasing quantifiers and therefore cannot be sustained. More concretely the following three objections have been raised.

Maximality Effect of Numerals

Numerals (*two*, *three*, *four*, …) can have an exactly-reading that cannot be captured by starting with an existential quantification over non-atomic individuals. Existential quantifiers are monotone increasing; therefore the maximality effect with numerals has to be otherwise explained.

We take sentence (70)a (see Lønning 1991) to clarify this first objection. Sentence (70)b is the paraphrase of the collective reading. The paraphrase for the distributive reading is less clear, it could be (70)c or (70)d.

(70)
a. Two men lifted a table.
b. There is an object y and y consists of two different men and y lifted a table.
c. There are two objects x_1, x_2 and for each of these objects: x_i is a man and x_i lifted a table.
d. The number of objects x_i such that x_i is a man and x_i lifted a table, equals two.

Link's representation (71) of the distributive reading of (70)a captures paraphrase (70)c.

(71) $\exists X(*man(X) \wedge |X| = 2 \wedge {}^{\mathbf{D}}\lambda U.\exists Y[table(Y) \wedge lift(U,Y)](X))$

$\Leftrightarrow \exists X(*man(X) \wedge |X| = 2 \wedge \forall U(U \leq_i X \rightarrow \exists Y[table(Y) \wedge lift(U,Y)]))$

Scha's distributive interpretation (72) of sentence (70)a captures paraphrase (70)d:

(72) $|\{d \in [[man]] : \exists X(X \subseteq [[table]] \wedge |X| = 1 \wedge [[lift]](\{d\})(X)\}| = 2$

Due to the existential quantification in paraphrase (70)c and its corresponding representation

(71) it is not excluded that there were other men who were lifting tables, i.e. Link's representation entails an upward monotone *at least*-interpretation of numerals. In contrast, paraphrase (70)d formalised as (72) states that all men who individually lift a table are counted, i.e. all that counts is the precise number of men lifting a table. This corresponds to the *exactly*-interpretation of numerals.

As I have discussed before both interpretations do in fact occur: the "referential" *at least*-interpretation, for example as an answer to *wh*-questions, and the "quantificational" *exactly*-reading, for example as an answer to *how-many*-questions in a context where we count the objects in a database that have certain properties. As to the collective reading of numerals Link's and Scha's formalizations turn out to be equivalent. In both approaches the collective reading is "about" an unspecified group consisting of exactly two men who together lifted a table, i.e. a referential interpretation is proposed that does not exclude that other groups or individuals also lifted a table.

Especially for large numerals it is often felt that they have an exactly reading. Does this observation force us to assume one or even two additional *exactly*-readings for numerals as proposed by Scha and van der Does? In the referential tradition to indefinites (and numerals) one often seeks a pragmatic solution to explain the exactly-effect of numerals. That there are not more than the specified number of tables is seen as a pragmatic implicature, i.e. as something that can be inferred from the utterance context but is not a logical entailment. More precisely, the exactly-effect is seen as a so-called *scalar implicature*. Scalar implicatures are one type of *conversational implicatures*. Conversational implicatures result from the hearer's assumption that the speaker will behave cooperative in an utterance situation ("cooperative principle") and therefore will follow certain rules of conversation. Grice (1975) investigated these rules and summarized them as so-called *maxims of conversation*: maxim of quantity ('Be as informative as required.'), maxim of quality ('Do not say that for which you lack adequate evidence.'), maxim of relevance ('Be relevant.'), maxim of manner ('Be clear, unambiguous, brief, and orderly.').

Scalar implicatures (Horn 1972, Gazdar 1979, Levinson 1984) are conversational implicatures that result form the application of the maxim of quantity. For example in the utterance *Some of the boys went to the party.* the word *some* triggers a scalar implicature to *Not all of the boys went to the party*. Scalar implicatures are triggered when terms are ordered within a so-called *implicational scale*. An impliational scale is a set of lexical items that are of the same syntactic category and that are ordered in terms of their informativeness. Examples for implicational scales are {*all*, *most*, *many*, *some*} or {*always*, *often*, *sometimes*}. An item is more informative if it *entails* other (lower) items in the scale, e.g. *all* entails *many*. Vice versa, less informative elements just *implicate* by the maxim of quantity that the speaker did not mean the more informative element of the pragmatic scale, e.g. *many* implicates *not all*.

The "exactly"-interpretation of a numeral can similarly be explained as a scalar implicature if one assumes that numerals have lexically associated with them a scale of numbers. For example Landman (1998, p. 242) suggests the following scale:

(73) n. n boys went to the party
…
5. 5 boys went to the party
4. 4 boys went to the party
…
1. One boy went to the party.

If sentence

(74) Four boys went to the party.

is uttered we calculate the *exactly*-interpretation as a scalar implicature triggered by the pragmatic scale in (73). The utterance of sentence (74) implicates that level 4 is the highest (most informative) in this scale for which (74) is true. If the speaker had known there were more objects she should have said so, she didn't hence the conversational implicature is there aren't more. Hence (74) implicates that there were exactly four boys at the party. See also Landman (1998) for a detailed discussion of calculating scalar implicatures.

Explicitly Non-Monotone Increasing Determiners

Link's approach does not directly generalize to *explicitly* non-monotone increasing determiners like *at most two* or *exactly two*. Distributive readings of non-monotone increasing cardinality quantifiers impose an upper bound on the cardinality of the noun denotation. However, representing distributive readings of non-monotone quantifiers by introducing an existential quantifier over groups and then distributing over this group entails upward-monotonicity. The explanation that the maximality effect stems from pragmatics cannot be maintained since the existence of the upper bound is made explicit, e.g. by adding *exactly* in front of a numeral. To avoid obviously wrong entailments the "exactly"-effect has to be part of the semantic representation itself and cannot be shifted to pragmatics.

The semantic representation of distributively interpreted non-monotone increasing quantifiers has to solve two problems: first, the maximality-effect (which was solved in Scha's approach), second the possibility of anaphoric references (which was not solved by Scha). The possibility of anaphoric references is illustrated by the following example (Link 1998a):

(75) At most four squatters were left in the building. They were dragged out by the police.

The discourse means that afterwards all squatters are gone. The interpretation has to guarantee that *at most four squatters* refers to the maximal set of squatters so that the anaphoric pronoun *they* points to all of the squatters left in the building.

To achieve this maximal interpretation with a referential (i.e. existential) interpretation of *at most four squatters* the semantic representation has to be augmented with an explicit maximality condition. I will show in section 4.5.2.5 how in my approach such a maximality condition can be integrated into the interpretation of non-monotone increasing quantifiers. Also Link himself suggests the addition of explicit maximality conditions in later work (Link 1998a). The representation to be developed will express the maximality effect of the distributive reading, it will nevertheless allow for collective readings and for discourse anaphora with determiners like *exactly 3*, *at most 5* etc. This counters van der Does' objection that Link's approach fails on non-monotone increasing quantifiers.

We have addressed above that in Discourse Representation Theory (Kamp and Reyle 1993) the phenomenon of anaphoric references to cardinality quantifiers is explained by so-called antecedent construal by abstraction. The problem is that the DRT approach does not generate collective readings of quantifiers like *at least n* or *exactly n*.

Quantificational and Non-Quantificational Use of Indefinites

A third related objection to Link's approach is that not only numerals but all indefinite noun phrases are in fact ambiguous between a referential and a quantificational reading where the quantificational reading cannot be adequately captured by an existential quantifier. If there is indeed a systematic ambiguity the pragmatic explanation given above is not sufficient. However, as I have discussed above in section 3.2.2.3 Löbner argues that non-quantificational readings of indefinites are by far the more frequent ones.

3.3.3.3 Evaluation

Neither Link's nor Scha's approach are immediately suitable for automatically processing plurals since both approaches generate too many readings. Scha assumed nine readings for a simple transitive sentence with two plural noun phrases. Taking scopal changes into consideration he postulated 18 readings (some of which are redundant). Link reduces these readings: there are four readings without scopal change and eight readings with scopal change (which is reduced to 7 because some are logically equivalent). Link does not introduce extra logical forms for cumulative or neutral readings thus reducing the number of readings the semantics generates for plural sentences. Link's approach has the advantage that – due to his lattice-theoretic ontology – he can formulate distributive readings in a system of first-order logic with λ-abstraction and a set of additional symbols and operators, like the *-, $^{\mathbf{D}}$- or ⊕-operators. The lattice theoretic approach also allows to treat the plural and the mass domain analogously. Furthermore, Links VP-centred approach can solve problems concerned with VP-conjunction because noun phrases are not considered to be ambiguous. "Dowty-sentences" like

(76) Four men went to the bar together and had a beer each.

can now be represented with just one reading for the subject noun phrase:

(77) $\exists X(*man(X) \wedge |X| = 4 \wedge go\_to\_the\_bar(\mathrm{X}) \wedge {}^{\mathbf{D}}have\_a\_beer(X))$

Furthermore, the distinction of individual denoting and quantifying noun phrases makes it easier to treat the anaphoric potential of individual denoting noun phrases.

The question is, however, whether Link's approach undergenerates because he neglects intermediate situations and cumulative readings. I will discuss this problem with respect to my approach in section 4.3. In van der Does (1993) several technical details of Link's approach are criticized, e.g. how to prevent the distributivity operator to be applied twice. Van der Does also suggests improvements. Also van der Does (1992, pp. 31) points out that formally Link is in fact committed to sixteen possible readings (again with some reduction). Since I only wanted to present the basic ideas of Link's approach I refer the interested reader to van der Does' articles.

Link himself suggests in a later article (Link 1998a) a solution to the representation of explicitly non-monotone increasing quantifiers. However, the maximality-effect that occurs with "normal" numerals only has a pragmatic explanation, furthermore the systematic ambiguity of indefinites between a quantificational and a non-quantificational interpretation is not represented. Also subsumption of the cumulative reading under the collective may turn out to be not precise enough to derive the intended inferences. However, with respect to computational applications overprecision often has the effect that it is impossible to find a realistic implementation of the theory. And even if a system does represent these additional readings the problem remains how to choose the relevant reading. Neither Link nor Scha propose an elaborate system of disambiguation.

3.3.4 No Ambiguity Strategy: Verkuyl and Van der Does (1991)

3.3.4.1 Basic Ideas

The main idea of the "No ambiguity Strategy is" that plurals are not considered to be ambiguous but have just one reading that is indeterminate with respect to collective, distributive or other readings. Similar to Scha, Verkuyl and van der Does assume a set-theoretic modelling of the domain. Their language allows for quantification over sets, and therefore uses a higher-order language. The idea that one logical representation encompasses all possible constellations sounds appealing at first sight. However, there are several empirical arguments against the one-reading-hypothesis that show that this approach cannot be sustained.

One Reading Hypothesis

The massive explosion of ambiguity that the systems of Scha and Link predict is countered by a proposal of Verkuyl and van der Does (1991, later published as 1996). They propose that a sentence like

(78) Four men lifted three tables.

is not ambiguous but that it has just one indeterminate reading which underinforms the hearer of the precise way in which the liftings took place in a particular situation. They call their approach the *one-reading hypothesis*. The approach claims that the use of plural noun phrases is genuinely indeterminate as to which interpretations are intended. The semantics should therefore strive to remain neutral in that respect. To realize this idea they propose a logically weak interpretation for the noun phrases that captures the range of possible interpretations from a completely distributive reading to a completely collective reading.

Representation of Reading

Technically, Verkuyl and van der Does extend the representation of NPs like *four men* so as to contain the information that the set of four men in (78) is subject to some kind of covering. A cover of a set X is defined as a collection $\boldsymbol{C}$ of non-empty subsets of X such that $\cup\boldsymbol{C} = X$ (in the following abbreviated as $\boldsymbol{C}$ *covers* X). The cover structure is obtained by constructing from a set X being a subset of the noun denotation [[*man*]] the collection of sets $\boldsymbol{C}$ such that each cell of the collection participates in the property expressed by the VP. This is expressed in (79) in a preliminary version disregarding the plural object noun phrase:

(79) $\exists X \subseteq [[man]] \wedge |X| = 4 \wedge \exists \boldsymbol{C}\ covers\ X : \boldsymbol{C} = \{[[man]] \cap Z : Z\ lifts\ three\ tables\}$

The representation expresses that there is a set X of four men and that X can be divided into a collection $\boldsymbol{C}$ such that for each of the elements ("cells") of $\boldsymbol{C}$ it is the case that this "cell" participates in lifting three tables. Van der Does (1992) shows that the notation (79) is equivalent to the "participatory" reading N2g introduced in (50) on page 54 above.

The following examples show the predictions of formula (79) with respect to some constellations where m_i stands for a man and w_i stands for a woman.

(80) a. **Situation A**
[[*man*]] = {m1,m2,m3,m4,m5,m6,m7,m8}
[[*lift three tables*]] = {{m1,m2},{m3,m4},{w1,w2}}
Formula (79) is true in situation A since there is an X = {m1,m2,m3,m4} such that $X \subseteq$ [[*man*]] and $|X| = 4$ and there is a $\boldsymbol{C}$ = {{m1,m2},{m3,m4}} such that $\boldsymbol{C}$ *covers* X and each element of $\boldsymbol{C}$ takes part in lifting three tables.

b. **Situation B**
[[*man*]] = {m1,m2,m3,m4,m5,m6,m7,m8}
[[*lift three tables*]] = {{m1,m2},{m3,m4},{m5},{w1,w2}}
Formula (79) is true in situation B since there is an X = {m1,m2,m3,m4} such that $X \subseteq$ [[*man*]] and $|X| = 4$ and there is a $\boldsymbol{C}$ = {{m1,m2},{m3,m4}} such that $\boldsymbol{C}$ *covers* X and each element of $\boldsymbol{C}$ takes part in lifting three tables.

c. **Situation C**
[[*man*]] = {m1,m2,m3,m4,m5,m6,m7,m8}
[[*lift three tables*]] = {{m1,m2,c1,c2},{m3,m4,c3,c4},{w1,w2}}
Formula (79) is true in situation C since there is an X = {m1,m2,m3,m4} such that $X \subseteq$ [[*man*]] and $|X| = 4$ and there is a $\boldsymbol{C}$ = {{m1,m2},{m3,m4}} such that $\boldsymbol{C}$ *covers* X and each element of $\boldsymbol{C}$ takes part in lifting three tables.

d. **Situation D**
[[*man*]] = {m1,m2,m3,m4,m5,m6,m7,m8}
[[*lift three tables*]] = {{m1,m2},{m3,w1},{w1,w2}}
Formula (79) is false in situation D since there is no $X \subseteq$ [[*man*]] such that $|X| = 4$ and such that there is a $\boldsymbol{C}$ that covers X such that each element of $\boldsymbol{C}$ takes part in lifting three tables.

Note, that formula (79) is true in situation B although there are more than 4 table lifting men. That means reading (79) does not cover the exactly-reading of numerals. This was already noted in the discussion of the N2g reading in (50). Situation C illustrates that formula (79) treats the participatory reading, i.e. it is sufficient for the men to take part in the table lifting event. We get the collective "extreme" of the range of possible situations if the cover $\boldsymbol{C}$ equals {{m1,m2,m3,m4}}. If the cover $\boldsymbol{C}$ contains only singletons {{m1},{m2},{m3},{m4}} we get a distributive "extreme". All other configurations lie in between. What other approaches call a "distributive" or a "collective" interpretation is now only a special case of a more general weak interpretation. In particular, distributive and collective interpretations now depend on the nature of the verb phrase denotation in a particular context.

Verkuyl and van der Does generalize this treatment to two-place predicates such that each argument can receive a weak interpretation. This allows Verkuyl and van der Does to propose the single reading (81) for sentence (78).

(81) $$\exists X(X \subseteq [[man]] \wedge |X| = 4 \wedge \exists \boldsymbol{C}\ covers\ X\ (\boldsymbol{C} = \{[[man]] \cap V : \exists Y(Y \subseteq [[table]] \wedge |Y| = 3 \wedge \exists \boldsymbol{D}\ covers\ Y\ (\boldsymbol{D} = \{[[table]] \cap W : [[lift]](\mathrm{V},\mathrm{W})\}))\}))$$

Formula (81) says that there is a set X of four men and that X is divided into a collection $\boldsymbol{C}$ such that for each of the elements ("cells") of $\boldsymbol{C}$, i.e. the subsets V of X, there is a set Y of 3 tables where Y is divided into the subsets of Y lifted by V such that each of the cells W of the collection $\boldsymbol{D}$ is involved in the lift-predication. Or in other words, there is a set X that is covered so that for each element of the cover there is a set Y that is also covered so that there is some (yet unidentified) relationship R between the covered subsets of X and Y.

The idea is that formula (81) captures all so-called "readings" of Scha and Link at once. The covering of the NP denotation expresses that sentences like (78) fundamentally underinform about what happened with four men and three or more tables in a particular model. If no further information is available the sentence does not reveal what actually happened. The sentence in particular does not reveal how many tables were in fact lifted since for each possible covering of the denotation of the subject noun phrase there may be four tables. Note furthermore, that two place verbs remain indeterminate as to whether the individuals which make up their arguments are involved in the predication strictly individually or in groups.

3.3.4.2 Problems and Possible Solutions

The idea that one logical representation encompasses all possible constellations sounds attractive in the light of the massive explosion of ambiguity in Scha's and Link's proposals. There are however several arguments against the one-reading-hypothesis. Most of these arguments are raised by Verkuyl and van der Does themselves in succeeding works (e.g. Verkuyl 1999, van der Does 1992, 1995). Also Lønning (1991) pointed out at several problems. I will summarize these objections here.

Split Subject Interpretation

One problem addressed by Lønning is that the weak neutral interpretation allows a splitting of noun phrase denotations. This can cause undesired readings when the subject noun phrase gets such a split interpretation. For example the neutral reading (81) predicts that sentence (78) is true in a situation where a single man lifted three tables and a set of three men also lifted three tables, i.e. we choose the set {{m1},{m2,m3,m4}} as the relevant cover of the denotation of the subject *four men*. Formula (81) predicts that each element of the cover takes part in lifting three tables, i.e. we end up with the possibility of six tables being lifted. Judgments differ whether sentence (78) allows such an interpretation. The same problem occurs in an example due to Lønning (1989)

(82) Three boys bought a boat.

where the object noun phrase is an indefinite singular noun phrase. Accepting a split subject interpretation allows sentence (82) to be about two boats. This interpretation is rejected by many speakers. They can interpret (82) as pertaining to one boat or to three boats but not to two boats. Thus, for transitive predicates giving the wide-scope subject a weak neutral interpretation leads to unacceptable truth conditions. On the other hand, for one place-predicates and for direct objects of transitive sentences the cover readings seems to predict acceptable truth-conditions.

To meet this "split subject" problem one could assume that there are (possibly lexical) restrictions on the structure of the cover of the external argument NP (the subject in a transitive sentence). One could restrict the possible coverings of [[*three men*]] to the "collective" and "distributive" extremes. The distributive cover would be a set of singletons for each of which there is a set of three tables. The collective cover would only contain the set X itself leading to the "collective" interpretation of sentence (78). There would be no restrictions on the possible coverings of the internal argument *three tables*. This solution may work technically for a number of cases but seems rather *ad hoc*.

Unconstrained Covers

A further problem with the one-reading hypothesis occurs because formula (81) allows for unconstrained covers leading to the prediction that altogether more than twelve tables can be involved in sentence (78). More concretely, in formula (81) the possible covers of the set X range from the poorest cover $\{X\}$ via intermediate alternatives to its richest cover $\wp(X)$. The cardinality of the poorest cover is 4, the cardinality of the richest cover is 2^4. For each element of the richest cover there can be three tables being lifted leading to the possibility of up to $2^4 \times 3 = 48$ tables being lifted. This however is an inappropriate reading for the subject of a transitive sentence. Intuitively, sentence (78) does not allow us to speak about more than $4 \times 3 = 12$ tables. In general, under the normal scoping we expect in sentences with iterated numerals n and m that the number of tables lies between n and $n \times m$. But quantification using unrestricted covers allows an upper bound of $2^n \times m$ tables. The problem seems to be that in (81) the cardinality of the subject cover may exceed the cardinality of the underlying set.

This problem could be solved by strengthening the notion of a cover. The unrestricted cover condition $X = \cup \boldsymbol{C}$ seems too weak. It has been suggested to use pseudo-partitions, i.e. require that ***C*** covers X and additionally $|\boldsymbol{C}| \leq |X|$. This has the effect that at least four and at most twelve tables may be lifted in (78). A stronger restriction is to use minimal covers, i.e. require that no cell of the cover is fully included in another cell. The strongest restriction is to use partitions, i.e. require non-overlapping cells.

Van der Does proposes pseudo-partitions as the best compromise. But this choice is still debatable, since it allows for example for the following situation (van der Does 1992, p. 53):

(83) a. Richard and Harry each lifted two tables
b. Richard and Ellen together lifted two tables.
c. Three people lifted two tables.

From (83)a and (83)b it follows that $\boldsymbol{C}$ = [[*lifted two tables*]] = {{r},{h},{r, e}}. Since $X = \cup \boldsymbol{C}$ = {r,h,e} and $|\boldsymbol{C}| \leq |X|$, sentence (83)c would be true in the situation described by (83)a and (83)b. Semanticists have varying judgments on whether (83)c can be so used. The kind of verifying situation may be easier to get having sentences that are explicitly modified such as

(84) Three people lifted two tables alone or with others.

However, for texts that aim at describing situations clearly and precisely neither (83)c nor (84) would express the information described (83)a and (83)b in a cooperative way. Furthermore, the additional constraints on covers do not prohibit split subject interpretations, i.e. the first problem is not yet solved by strengthening the notion of a cover.

Cumulative Readings and Maximality-Effect

Van der Does (1992) argues that formula (81) does not correctly handle the cumulative reading of sentence (78). The cumulative reading says that the total number of men lifting tables is four and that the total number of tables being lifted by men is three. This cannot be obtained by representation (81) which is inherently asymmetric, i.e. one existential quantifier is in the scope of the other existential quantifier. Formula (81) says that there is a cover ***C*** and for each element of that cover there is a *Y* such that … In contrast, the cumulative reading requires the scope of the two NPs to be independent. According to van der Does, the cumulative reading cannot be represented by giving both noun phrases the trivial cover $\{\{X\}\}$ and $\{\{Y\}\}$, respectively, and thus making scope phenomena irrelevant. The reason is that treating the cumulative reading as a doubly collective reading leads to two existential quantifiers which – due to the upward monotonicity of existential quantifiers – does not exclude that there are more men who lift tables in a particular model. We have seen a similar effect already in (80)b. The maximality effect cannot be captured by mere existential quantification. If it turns out that the exactly-effect of cumulative readings cannot be satisfactorily explained then the one-reading-hypothesis has to be replaced by at least a two-reading hypothesis.

I have discussed possible solutions to the maximality problem in 3.3.3.2 above. One solution lies in a pragmatic explanation for simple numerals and in adding additional maximality conditions with explicitly non-monotone quantifiers.

Since Verkuyl and van der Does found no satisfactory solution to the problems mentioned above the one reading hypothesis was finally abandoned. Van der Does instead suggests a minimum of three readings whereas Verkuyl suggests two readings.

3.3.4.3 Evaluation

The main advantage of the no-ambiguity approach is of course a massive reduction of ambiguities with plural noun phrases. However, giving noun phrases a (contextually dependent) weak cover interpretation is problematic since its constraints are too weak, i.e. it allows for two many verifying situations that are intuitively not adequate. A further disadvantage for practical applications is that the proposed weak reading cannot be formulated in first-order logic, since quantification over members of a cover (i.e. sets) is higher-order.

Verkuyl and van der Does tried to formalize the correct observation that plurals are in many cases indeterminate as to the precise realization of a certain relation. The problem is that their formalization tries to describe what is going on in a particular situation instead of relying on lexical entailments triggered by verbs or other elements of the sentence. To formalize the contribution of the verb to possible verifying situations is – of course – equally difficult, however, it is very often simply not necessary to spell out these possible verifying situations.

3.3.5 Global Strategy: Roberts (1987), Van der Does (1992)

The “Global Strategy” to represent plural ambiguities can be seen as a combination of the other strategies. The ambiguity is seen as a result of the whole structure, or – as Roberts (1987)þputs it:

> Distributivity is a property of predications, combinations of a subject and a predicate. [...] Distributivity may be triggered either by a quantificational determiner in the subject NP or by the presence of an explicit or implicit adverbial distributivity operator on the predicate. (Robertsþ1987,þpp. 100)

A similar strategy is also pursued by van der Does (1992).

The global strategy aims at localizing the ambiguity not at a single lexical element but attribute it to the global structure of the sentence. Whether a sentence is interpreted distributively, collectively or otherwise depends on the constituting categories, and possibly on pragmatic, contextual or other factors. A reduction of ambiguity is achieved by restricting the possible combinations that are compatible with each other.

The basic idea of the global approach is interesting in that it is open to take into account many factors that determine different readings. The approach faces the same problem as the other approaches, viz. to decide which readings are the basic ones, how to represent the readings and how to formalize the factors that determine one or the other reading. Technically the “Global Approach” can be formalized in different formats, e.g. GQT or DRT.

In this thesis I will follow a global strategy and will discuss one possible formalization and implementation of a global strategy. The main idea is that – in general – lexical items like verbs, determiners or adjectives themselves are not considered to be ambiguous between a collective or a distributive reading. Which of the readings is generated is determined by a complex interplay of features encoding possible disambiguation factors.

3.4 Evaluation of Existing Approaches for Natural Language Understanding

In the previous sections I have looked at a number of prototypical examples within formal semantics to represent the meaning of plurals in natural language. The plural phenomena addressed are only part of a larger field of problems, furthermore the approaches discussed are only a selection among many other existing approaches. In particular, I have not introduced event related approaches to the representation of plurals. For an overview see articles in Rothstein (1998), or work by Krifka (1989b), Eberle (1998), Moltmann (1997), Schein (1993) and others. Many of the event related approaches discuss the same problems and offer similar solutions like Link or Scha – unless they are concerned with event specific problems like VP modification. Since my approach will introduce events in the semantic representation some references to these approaches will be made in the respective sections.

Both, Scha's and Link's approach suffered from a combinatorial explosion of readings. Neither of them offers a theory of disambiguation. Verkuyl and van der Does reduced the number of readings but their approach could not be maintained for empirical reasons.

What have we learned for the representation of plurals in practical applications? A semantic component for natural language understanding will have to answer the same basic questions as the formal semantic approaches and it will partly use similar techniques. The basic questions are repeated here:

- Where does the ambiguity come from?
- How many readings are assumed?
- What is the underlying ontology?
- How are the readings represented?

All these topics have to be dealt with taking into account the following question:

- How suitable is the approach for computational applications?

Concerning the first question I have already suggested that a "global approach" is most suitable for a flexible practical approach, i.e. neither determiners, nor verbs, nor other lexical elements are – in general – ambiguous between a collective or a distributive reading but the ambiguity (and the disambiguation information) has to be attributed to global properties of a structure. Sometimes the lexical semantics of verbs disambiguates, sometimes the type of the determiner, sometimes additional syntactic markers etc. In my approach I will categorize the different disambiguation factors and weigh them according to their "disambiguation strength" (see chapter 5).

As to the second question a practical system has to avoid a combinatorial explosion of readings for efficiency reasons. On the other hand "real" ambiguities have to be detected to prevent a natural language system to perform the wrong actions or to draw the wrong conclusions. That means one has to carefully draw the borderline between different readings on the one hand and different verifying situations of one and the same reading on the other hand. One main problem in giving an adequate semantics to plural phenomena is that plural expressions are notoriously imprecise in their reference – which serves the efficiency of natural language. Many people often do not want to be more explicit than necessary, they are simply not interested in the precise constellations of a certain action if it is not relevant for what they want to convey. Formal semantic representations on the other hand aim at a high degree of precision. Therefore the task is to find representations that come close to this empirical level of accuracy. That means it is not only important to find semantic representations for plurals but also to avoid unnecessary overprecision. Link (1998a, p. 21) stresses this point with the following example

(85) The Romans built the aqueduct. They were excellent architects.

In (85) it is not at all clear what the plural noun phrase *the Romans* refers to. Obviously not all Romans built the aqueduct, nor are all of them excellent architects; perhaps even the ones who erected the aqueduct were not the same as the architects.

In particular Scha's but also Link's approach have been criticized not only because of the combinatorial explosion of readings but also because of the "overprecision" of their approaches. With his neutral and cumulative readings Scha tried to describe what was going on in the situations verifying the sentences rather than relying on judgments of entailment. For example, if we analyse a sentence like

(86) Four boys shared three pizzas.

in its non-distributive reading we just don't know what was going on apart from the fact that there were four boys and three pizzas and that there was some indeterminate sharing-relation between the group of boys and the group of pizzas. The meaning of the verb *share* allows for many possible constellations but it is not necessarily the task of the formal semantic representation to spell out these possible constellations. Of course, this view shifts the problem to the lexical semantics of verbs. But for natural language understanding tasks it is not always necessary to spell out the lexical semantics precisely. On the one hand, many inferences can be drawn even if one accepts a certain level of imprecision or indeterminacy. On the other hand, many inferences that come out as logically valid in "precise" formal semantic approaches (e.g. monotonicity properties of quantifiers) turn out to be less self-evident than the theory predicts. Therefore, missing some of these inferences by opting for a "modest" inference strategy and not over-burdening the semantic representation is not necessarily a disadvantage.

As to the underlying ontology I will argue that a lattice-theoretic approach is to be preferred. Two major advantages are that it allows to formalize the semantic theory in a first-order language, second it can be easily extended to treat both the count *and* the mass domain.

Concerning the representation of readings I will opt for a discourse related approach since not only isolated sentences but mainly anaphorically related text will have to be processed. Furthermore, I will argue for a computationally suitable first-order formalization of the readings and use a similar approach like Link (without using many specialized operators though).

In the next chapter 4 I will discuss that formal semantics and computational semantics often have a different view on what constitutes the meaning of an expression. Whereas formal semanticists view the meaning of a sentence in terms of its truth-conditions (i.e. the expression is interpreted with respect to a real world situation), computational semanticists often have a proof-theoretic view of meaning, i.e. they view the meaning of the sentence in terms its entailments with respect to a formal knowledge-base. In developing a semantic representation for natural language understanding tasks one therefore has to keep in mind that the representations must lend themselves to automatic theorem proofing tasks. Devising specialized inference procedures for specialized languages turns out to be complex and difficult. Therefore it is often suggested to use off-the shelf first-order theorem provers and model generators for this task. This restricts the suitable representations considerably as I will discuss in the next chapter.

4 Representing Plurals for Natural Language Understanding

4.1 Overview

This chapter presents a semantic representation of plurals suitable for natural language understanding applications. In developing the representations I had to deal with the same questions as the formal approaches in the previous sections, viz. where to locate the ambiguity, how many readings to assume, which underlying ontology to adopt and how to represent the readings. All questions have to be answered with a special emphasis on the suitability for computational applications. Section 4.2 will deal with a set of requirements that a representation of plural ambiguities must fulfil to be suitable for the applications pursued in this thesis. The section includes a brief overview of different perspectives on semantic processing, viz. formal vs. computational semantics and truth-conditional vs. proof-theoretic semantics. In this thesis I propose a computational proof-theoretic approach to plural semantics. Section 4.3 will then deal with the question how many readings are assumed, section 4.4 will introduce the formal setting: the language of discourse representation theory and lattices as the underlying ontology, and in section 4.5 the representations will be introduced, also dealing with the question where to locate the ambiguity.

4.2 A Proof-Theoretic Approach to Plural Semantics

In this section I will introduce a set of requirements that a representation of plural ambiguities must fulfil to be suitable for the applications pursued in this thesis. Since the representation will be used in practical applications that perform automatic reasoning and query answering a trade-off between expressive power and logical tractability has to be found. Considering that in-depth research has been undertaken to explore first-order inference computationally and that a wide range of advanced theorem provers, model builders and other tools are freely available I will propose to reuse these off-the-shelf tools. This decision not only requires the representation language to be first-order but also results in some rather specific requirements for the acceptable syntax of the semantic representation language. Before I list the requirements in section 4.2.3 I will show in sections 4.2.1 and 4.2.2 that there are different perspectives on semantic processing that imply different requirements.

4.2.1 Formal vs. Computational Semantics

A semantic theory for natural language investigates what the meaning of natural language expressions (words, phrases, sentences, texts) is. This includes determining how the meaning of elements of the expressions can be combined to build up complex meanings. The meanings are generally represented by giving a formal meaning representation, e.g. by using a logic language. The logic representation then allows us to simulate the intuitively acceptable interpretations and entailments of sentences and texts. Depending on the final purpose of the semantic representation there can be different priorities and requirements on the logic language. Formal semanticists try to model our intuitions about the readings of sentences as accurately as possible. They search for expressive formalisms as suitable representation languages, aim at explanatory adequacy, and try to map natural language expressions to that formalism in a systematic and general way. Computational semanticists are, beyond that, interested in using the semantic formalism in practical applications. In addition to the tasks of formal semanticists they therefore have to investigate the computational properties of the semantic theory so that it is applicable to real-world applications. They check whether the formalisms can be realistically implemented and how the representations can be automatically post-processed. Further topics of interest include efficiency (performance of the system in terms of time and resource behaviour), coverage (how many phenomena can be described), robustness (how does the implementation deal with input not covered by the theory) and user-friendliness (how useful is the computational semantic system for real applications). See also Monz and de Rijke (2000) for an introduction to these questions.

Chapter 3 focused on the formal aspects of plural semantics, in the following sections I will investigate how the results of the previous sections have to be modified to be suitable for practical implementations. Before I focus on the computational properties of the representation language in section 4.2.3 I want to discuss different attitudes towards the nature of meaning that formal and computational semanticists have.

4.2.2 Truth-Conditional vs. Proof-Theoretic Semantics

Formal and computational semanticists tend to have a different overall picture of how to define the concept of meaning and how to determine the meaning of natural language expressions. Formal semanticists often follow a truth-conditional model-theoretic meaning theory whereas computational semanticists focus on proof-theoretic approaches. The two approaches are not always clearly distinguished. Since in this thesis I will follow a proof-theoretic approach to plural semantics (see section 4.2.4) I will introduce the general differences between the two approaches.

Truth-Conditional Model-Theoretic Semantics

Within formal semantics Montague (1973, reprinted in 1974) is a typical representative of a *truth-conditional* view of meaning where the meaning of sentences is determined via *model-*

theoretic techniques. A *truth-conditional* view proposes that to know the meaning of a sentence corresponds to knowing the necessary and sufficient conditions for a sentence to be true, that is, to correspond to a state of affairs in the real world. To calculate the meaning of a natural language sentence a *model-theoretic* approach is proposed. The sentence is translated into a formal language and additionally a formal mathematical model of the world is assumed. Expressions of the formal language are then systematically mapped to objects of this model via an interpretation function. The interpretation function assigns e.g. objects from the domain to the constants of the formal language, truth-values to the representation of declarative sentences, truth-functions to the connectives, other functions to the function symbols, and extensions to the predicates. Though an intermediate formal model is established, the final foundation of semantics is the real world: a sentence is interpreted with respect to a formal model of the real world. According to a truth-conditional model-theoretic view a sentence is true if corresponds to a state of affairs in the real world. That means 'truth' is considered to be a relationship between language and an independently existing reality. It is also important that classical model-theory can be understood to deal with *static* relationships among individuals. To explain intensionality in natural language, i.e. statements not only about the world as it actually is, but also about the world how it is imagined to be, how it ought to be etc. the concept of 'possible worlds' is introduced. The truth of a sentence is then relative to a chosen possible world and point in time.

Truth-conditional semantic approaches are, however, not directly suitable for practical natural language understanding systems for several reasons. First, they tend to be computationally impractical. Montague's formulation of semantics for example "throws around huge sets, infinite objects, functions of functions, and piles of possible worlds with great abandon" (Hirst 1987, p. 32). A second disadvantage is that truth-conditional semantics is not what we aim at in computational applications like text understanding, query answering or dialogue-systems. In these applications our purpose is not to interpret the sentences with respect to a *real* world: whether objects exist in the real world or whether declarative sentences are true with respect to an existing situation is not checked. Rather we are interested in the state of affairs itself. And this state of affairs is "constructed" by the natural language text itself. Also, in its original formulation, truth-conditional semantics really only deals with the interpretation of declarative sentences: the interpretation of a declarative sentence consists of its truth-value. However, a natural language understanding system also needs to able to deal with questions or commands that cannot be assigned a truth-value. Blackburn and Bos (2000a, introduction to chapter 4) hint at a further disadvantage of truth-conditional model-theoretic approaches. In these approaches the concepts of validity and logical inference are purely semantically defined. A sentence is valid if it is true in all models, and a valid inference is an argument such that whenever all premises are true in some model, the conclusion is true in that model also. Both concepts – validity and logical inference – are defined in terms of models, i.e. in terms of semantic objects. Moreover, the definitions refer to the class of *all* models which is a very large and abstract object. It is certainly not realistic to put all those models in a computer and check whether an argument is valid by checking all premises and all conclusions in all of those models (which can be infinitely large). To perform logical inference on a computer it is therefore necessary to look for other approaches where the concept of inference is defined on a syntactic not on a semantic basis. This is done by proof-theoretic approaches that I will look at next.

Proof-Theoretic Semantics

For computational approaches a truth-conditional view to semantics is often replaced by a proof-theoretic approach where sentences are not *interpreted* with respect to the real world but *evaluated* with respect to a formal *knowledge base* that is constructed from the natural language text. The idea is that (declarative) sentences are assumed to describe a true state of affairs and thus their formal representation is constituting a part of the knowledge base. The knowledge base is then some collection of sentences in an appropriate logical form. The meaning of a sentence is accordingly no longer described in terms of truth-conditions but in terms of the inferences that can be drawn from that sentence with respect to the knowledge base, or as Hirst puts it with reference to Tarnawsky (1982): "The meaning of a sentence depends on the knowledge of the interpreter and includes the propositions, possibly infinite in number, entailed by the sentence with respect to that knowledge" (Hirst 1987, p. 39). Determining the meaning of a sentence involves translating it – purely syntactically – into an appropriate formal representation and evaluating its logical consequences (with respect to other sentences). Under this perspective it possible to deal not only with declarative sentences but also with questions and commands. The evaluation of declarative sentences consists of adding a formal representation of the sentence to the knowledge base (plus considering the entailments of the addition), the evaluation of a question consists of finding an answer with respect to the knowledge base and the evaluation of commands consists of executing operations on the knowledge base. Thus the knowledge base is not a static object, that is completely given in advance, but rather viewed dynamically because it can be permanently extended when texts are interpreted. Under this perspective truth is a relationship between sentences of the knowledge base and new sentences that are checked with respect to the knowledge base. Note, that the formal knowledge base – consisting of the logical representation of natural language sentences – can also be seen as a "model" of the world, but this model is only dynamically created by the description of the world via sentences and not statically given in advance as an abstraction of the real world. These "models" have to be distinguished from models in the original model-theoretic meaning in that the former are purely syntactical objects. They make no assumptions about the existence of individuals, functions and relations apart from those projected by the vocabulary of the language. The models thus have to be understood "pseudo-model-theoretically in terms of Herbrand interpretations". (Kowalski 1994, p. 3). Under a "pseudo-model-theoretic" re-interpretation of knowledge bases a sentence is true if it is represented in the knowledge base or is provable from the knowledge base. A sentence is false if its negation is true.

In model-theoretic approaches possible-world semantics was introduced to explain intensionality in natural language. Several proof-theoretic approaches offer purely syntactical alternatives that employ a syntactically rich vocabulary of terms representing time, events, situations, theories etc. (Hobbs 1985, Kowalski 1994). Furthermore, in proof-theoretic approaches the concept of logical inference and validity are defined only on the basis of syntactic structures of sentences; models play no role. Of course, as Blackburn and Bos (2000a, introduction to chapter 4) emphasize, proof-theoretic methods must always be *justifiable* in semantic terms. That means they may not be arbitrary symbol manipulations but should always be grounded in "real" intuitively valid logical inferences. Despite that, proof-theoretic methods require no appeal to semantic concepts, only syntactic manipulation of formulae is required. And this syntactic manipulation is much easier to perform on a computer.

4.2.3 Advantages of First-Order Logic for A Proof-Theoretic Approach

The shift from a truth-conditional to a proof-theoretic semantics also involves considerations concerning computationally suitable formal languages that can serve as knowledge representation languages. Language understanding consists of various stages like parsing, computing a representation and performing inferences. Inferences are made on the basis of the natural language text, of meaning postulates or of general world-knowledge. An essential requirement is therefore that the representation language supports all these types of inferences and is also suitable to express world knowledge. That means the choice of the representation is central. Both the mapping of representations to natural language input and performing inferences depend on the representation used. This centrality of representation in natural language understanding has e.g. been stressed by Schubert and Hwang (2000).

Two major classes of representations have been used: knowledge structures like semantic nets, frames or scripts, and logical representations like predicate logic, higher-order logic or various forms of intensional logics (see Hirst 1987, p. 33).

The advantage of using logic representations is that they are widely accepted as general knowledge-representation languages, that they can therefore be easily reused in different applications and – since there is research across different fields on foundations and applications of logic languages – a wide variety of off-the-shelf tools to process logic languages is available. There are several logic languages that have been proposed as suitable semantic representation languages. The most widely used, studied and implemented version of logic is classical first-order logic. Other languages are for example modal logic, higher-order logic, typed logics etc. (Sowa 2000). The languages differ mainly in expressive power, in readability, in their logical and proof-theoretic properties. First-order predicate logic for example allows quantification over individuals, but not over predicates. Higher-order logics are more expressive in that they allow quantification over first-order predicates as well. However, the computational complexity of higher-order inference increases and – what is more – desirable theoretical properties get lost: while first-order logic is complete, second-order logic is not. Sowa (2000) puts the strength of first-order logic as follows: "besides expressive power, first-order logic has the best-defined, least problematic model theory and proof theory …".

The key to overcome the limited expressive power of first-order logic is to assume richer ontologies. New entities are introduced into the domain of discourse and additional axioms are defined that constrain how these entities behave. For example, in the realm of natural language semantics, possible worlds or situations are introduced into the domain to describe modalities, events are introduced to describe temporal and aspectual relations, and groups are introduced to describe pluralities. As long as the additional constraints describing the objects can be stated in first-order logic we can speak of a full reduction of a formalisms to first-order logic. Many representation formalisms can be fully reduced to first-order logic. These include formalisms which at first glance seem to lie beyond its reach, e.g formalisms that make use of modal or temporal operators. Reduction means that the formalisms are pure notational variants that are equivalent to first-order logic and from which exactly the same theorems can be derived. This is important in that the well-understood logical properties of first-order logic (e.g. its correctness and completeness) apply as well to the notational variants. As we will see later, there is for example a simple reduction of discourse representation theory (DRT) to first-order logic.

There are, however, limitations to the technique of ontological enrichment. If the additional constraints describing the entities need to be formulated in a higher-order language then the first-order modelling is only an approximation. For example Lønning (1989) showed that formulating plural semantics in a first-order language using a lattice-theoretic approach requires constraints that are not first-order definable. More concretely, the full notion of completeness in a semilattice (existence of suprema for arbitrary non-empty sets) is not first-order definable. See also Link (1998b, pp. 147) for a brief summary. But despite these limitations very often the first-order approximations are "good enough" from a practical point of view.

In general, there are formalisms that can only partially be reduced to first-order logic. Partial reduction means that there are minor changes, e.g. loss of expressive power. Blackburn and Bos (2003, section 2.4) mention as an example the quasi-reduction of higher-order logic to first-order logic by introducing extra entities into models and constraining them to act like higher-order functions. Not all the required constraints can be written in a first-order way (otherwise there would be no difference between first-order and higher-order logic) but in many cases it is possible to achieve a practically sufficient first-order approximation to higher-order logic by formulating appropriate first-order postulates.

In general, for different applications, different logic languages are suitable. This applies as well to finding a suitable knowledge representation formalism for natural language semantics in general and plural semantics in particular. If a semantic formalism is mainly used to sharpen our intuition about the readings of a sentence a formalism with a high expressive power and good readability will be preferred and its logical and proof-theoretic properties will be of minor interest. If, however, a semantic formalism is used in practical computational applications not only expressive power but well-understood logical properties, implementability and computational efficiency will have to be considered as well.

In this thesis I will follow a proof-theoretic approach to plural semantics and the logic representations will be used in practical applications that perform automatic reasoning and query answering. This makes first-order logic the best choice as a semantic representation language.

4.2.4 Proof-Theoretic Plural Semantics – Goals and Methods

The following subsections briefly summarize the main goals, methods and results of my approach to represent the semantics of plurals.

4.2.4.1 Basic Assumptions

The overall goal of my proof-theoretic approach to plural semantics is to practically implement the semantics of a comprehensive set of plural constructs including the inferences that are associated with them. This also involves providing algorithms to reduce the ambiguities associated with plural constructs. My approach will answer the four basic questions concerning plural semantics as follows:

Location of Ambiguity. Readings of complex sentences containing plurals cannot in general be traced back to individual elements of a sentence, but are a result of the whole structure. This for example means that noun phrases are – in general – not ambiguous between collective, distributive or other readings.

Number of Readings. Ambiguity has to be carefully distinguished from indeterminacy. As a result I will only distinguish collective and distributive readings, no separate logical forms for cumulative or mixed readings are introduced.

Underlying Ontology. I assume an algebraically structured domain of discourse that is not given a set-theoretic representation but rather characterized by axioms that describe the behaviour of objects axiomatically. For the plural domain I assume a lattice-theoretic structure. An algebraic view involves that there is no commitment as to the “real nature” of the objects considered – which is in line with the proof-theoretic perspective of my approach.

Representation. I will use a “flat” DRS language that is a variant of first-order predicate logic to represent plurals.

As to the additional question of computational suitability I will show in the sections and chapters to follow that the approach can be used in practical applications performing inferences on plural sentences.

4.2.4.2 A Proof-Theoretic Approach to Handle Plurals

The ideal goal is to automatically perform those inferences on plural sentences that are felt to be intuitively valid by humans. This is of course an unrealistically ambitious goal since natural language interpretation is based on complex reasoning mechanism, in particular it is – in general – more than *logical* inferencing in the strict sense of the word. Furthermore, it involves many knowledge sources that are computationally hardly tractable. Still, important contributions to the interpretation of discourses containing plurals can be made. In my thesis I follow two main goals: First, I will develop techniques to automatically perform *logical* inferences on sentences containing plurals (section 4.5 and chapter 6). Second, I will develop techniques to automatically disambiguate plurals (chapter 5). Whereas the first goal is mainly realized by applying logical inferencing techniques, the second goal is realized by parameter based calculations involving non-standard reasoning techniques. I will deal with the logical aspects of my approach in this chapter and explain the disambiguation algorithm in the next chapter 5.

For the first goal, logical inferencing with plurals, I will choose a proof-theoretic approach.

This is in contrast to many of the existing approaches within plural semantics. I have summarized the main arguments for a proof-theoretic approach already in section 4.2.2. In particular, proof-theoretic approaches rely only on syntactic manipulations and are thus easier for computational implementations. Also, proof-theoretic approaches are more suitable to describe dynamically changing states of affairs. In discourse interpretation this is a more realistic situation since we cannot hope to have a complete static description of the world a priori. Furthermore, proof-theoretic approaches can easily deal with different types of sentences. Specifically, concerning the treatment of plurals I see a further advantage of a proof-theoretic approach not mentioned so far. According to my opinion, a proof theoretic approach is more suitable to deal with the indeterminacy of many plural sentences. To represent the non distributive reading(s) of sentences like

(87) Four men lifted three tables.

a truth-conditional model-theoretic approach tries to explain how the world can look like for the sentence to be true, that means it has to explain a whole lot of possible combinations between men and tables that make the sentence true. Under a proof-theoretic perspective we are not directly interested in possible constellations, but only in the inferences we can draw from the sentences. And if a sentence is not very informative (e.g. if the constellations are not explicitly spelled out) we simply cannot draw certain inferences about the constellations. This is, however, not a disadvantage but in fact an advantage, since the sentence may be deliberately indeterminate and certain inferences are therefore simply not necessary or intended. I will discuss this point in more detail in section 4.3.2.

The guiding principle in designing logical representations for plural sentences is always to perform the empirically observed inferences automatically within a logical framework. In my proof-theoretic approach the semantic representations are therefore seen as a mere tool to achieve this goal. The representation language is seen very pragmatically and the main criterion is its adequacy for this practical task. In particular the language and the semantic theory is not burdened with further commitments such as ontological statements, or explanations of syntactic facts. Also, I do not attempt to make psychological assumptions that "explain" how human reasoning "really" works. See Hobbs (1985) and Kowalski (1994) who also stress this "pragmatic" view of semantic representations.

The decision to choose a proof-theoretic approach, of course, still leaves great leeway concerning how the representations actually look like. In the following sections a systematic modular definition of the logical forms generated by natural language sentences will be given, and it is shown that the representations are suitable for the intended goals.

4.2.4.3 Architecture of Discourse Interpretation

In a proof-theoretic approach to semantics logical inference plays a crucial role. Hobbs (1985) explicitly associates discourse interpretation with inferential processes:

> Discourse interpretation processes, as I see them, are inferential processes that manipulate or perform deductions on logical expressions encoding the information in the text and on other logical expressions encoding the speaker's and the hearer's background knowledge. (Hobbs 1985, p. 61)

Hobbs addresses that during discourse interpretation several sources of information are used for (logical) inferences. In my architecture I will use two main sources of information: (i) information explicitly expressed in natural language texts or discourses, (ii) external knowledge sources that are given in addition to the explicit information in the text. The external information can – in general – consist of world-knowledge including for example mathematical knowledge, specific background knowledge about a particular situation or domain, but also more abstract linguistic knowledge, for example syntactic, semantic (lexical or structural) and pragmatic knowledge. Concerning linguistic knowledge typically structural semantic knowledge is used for logical inferences. Note, however, that not all information sources enter *logical* inferences, e.g. pragmatic inferences are in general not logical inferences in the strict sense of the word. Also, information about disambiguation defaults are not treated as *logical* inferences in my framework. The rest of this chapter 4 contains my suggestions how to encode additional information for logical inferences (mainly structural semantic knowledge and simple mathematical knowledge), whereas the treatment of disambiguation defaults will be dealt with in chapter 5.

Both knowledge sources – explicit and external information – are finally represented in the same logic representation language. To extract explicit textual information a systematic modular translation from natural language to the logical representation will be defined. Other information sources have to be predefined directly in the logic representation language.

4.2.4.4 Logical Basis for the Proof-Theoretic Approach

The above sketched proof-theoretic perspective results in a number of requirements for the representation language.

Flexibility. The representation language has to be suitable to integrate both, explicit knowledge derived from natural language text, and the more general external knowledge sources given in advance via axioms.

Syntax-Semantics Interface. It should be easy to automatically and systematically generate the logic representation from English surface structure.

Expressive Power. The representation language should have sufficient expressive power to represent plural phenomena as detailed as necessary for the considered applications, and to express additional external information needed for inferences.

Logical Properties. The representation language should have well-understood logical properties that are suitable to perform automated reasoning.

Simplicity and Tractability. The representation language should use a simple syntax so as to ease syntactic manipulations. In particular, the language should be suitable for automatic computational manipulations.

Generality. The representation language should use not only a simple but also a general syntax so that it can make use of standard computational tools, in particular off-the-shelf automatic theorem provers. This will result in the requirement to use standard first-order syntax as far as possible. The definition of additional operators (e.g. ι-, λ- or *-operators) or non-standard quantifiers should be avoided.

The following sections will show how these logical requirements will be satisfied. The approach is based on a first-order language that is considered to be most suitable for a proof-theoretic approach. First-order logic has well understood *logical properties*, has – as I will show in section 4.5 – sufficient *expressive power* to represent a major part of the plural constructs, and first-order logic is *general* and *reusable* since off-the-shelf theorem provers and model generators exist for first-order logic. More concretely, natural language texts are first translated into specifically designed discourse representation structures (DRSs) that are a variant of first-order logic. Using DRSs allows for an easy *syntax-semantics interface* (section 4.4.1). The DRSs generated from natural language texts are finally automatically translated into standard first-order logic (section 4.4.3) which is also the formal language to express auxiliary axioms that formalize additional external knowledge sources. This shows the *flexibility* of the representation language(s). The DRSs are constructed so as to be maximally informative, that means representations are selected that allow us to get logical inferences “for free” or almost “for free”. This for example means that in section 4.5 the use of additional operators like the *-operator or the ι-operator – that would need extra auxiliary axioms – are avoided. Also non-standard quantifiers are avoided. This method has the side-effect that the syntax of the representation language is *general* and easily *reusable*. As will be exemplified in sections 4.4.4 and section 4.5 some intended deductions cannot be performed on the basis of the DRSs alone since the inferences require additional knowledge, for example general linguistic knowledge about the lattice structure of plural objects. Also reasoning about identity must be possible. Furthermore, reasoning with plurals often requires simple mathematical knowledge for example to make numeric comparisons, or to perform simple arithmetic. DRSs and the additional auxiliary axioms that represent these external knowledge sources use a simple “flat” notation (section 4.4.2) that allows us express even those formulae in first-order logic that normally require second-order logic. The flattening of the notation makes the language very *flexible*. Furthermore the *simple* syntax eases computational manipulations and thus makes the language *tractable*. Moreover, it is possible to access Prolog predicates from logical representations derived from both explicit and additional external knowledge sources (chapter 6). This *flexibility* is important for instance to concisely and efficiently include arithmetic operations.

4.3 How Many Readings?

4.3.1 Overview

In this section I will first motivate how many readings I assume in my approach before I show in sections 4.4 and 4.5 how these readings are represented.

Both Scha's and Link's treatment predicted that sentences involving (one or more) plural NPs have a fairly great amount of readings. Both complex classifications were rather driven by "a logician's view" (Link 1998a, p. 32). The goal was to show the various quantifier structures that a plural sentence gives rise to when treated precisely in a formal framework. Due to the combinatorial explosion of readings the fine-grained distinctions are not manageable for practical applications. I will therefore reconsider a number of examples from a practical viewpoint and decide how many *possible* readings I will assume. The question of reducing possible to *plausible* readings will be postponed to chapter 5, also the discussion of possible and plausible scope ambiguities will be dealt with there. Section 4.3 discusses in particular the question whether we need separate cumulative and mixed readings. In this discussion I will adopt some of the ideas formulated in the "No Ambiguity Approach" by Verkuyl and van der Does. Also, distinctions drawn by Link and Schütze (1991) have influenced the treatment. The notion of indeterminacy as introduced in section 2.4 will be a crucial concept. The main argument is that in some constructions the occurrence of plural noun phrases leads to ambiguity whereas in other constructions we only observe indeterminacy. I will distinguish only two interpretations, the distributive and the non-distributive (the rest). The various types of non-distributive interpretations are seen as a result of indeterminacy. The non-distributive interpretation comprises collectivity in a narrow sense (one collective relation with a plural object) and collectivity in a broad sense (the actual realization of the relation is not further specified). I will furthermore assume that noun phrases and verbs are, in general, not intrinsically ambiguous but that the distributive reading is triggered by a larger context in the sentence. The distributive interpretation will be represented by a universal quantification over the individual members making up the plural object, whereas the non-distributive interpretation is represented by a collective reading. Cumulativity effects are seen as a side effect of the cumulativity of certain types of predicates.

4.3.2 Are Plurals Ambiguous or Indeterminate?

Collective and Distributive Readings

Most researchers agree that there should be a logical distinction between distributive and non-distributive readings. Several arguments support this view.

The basic ambiguity definition 4 in section 2.4 tests for alternate truth value judgments. According to this test the sentence

(1) Three men lifted a table.

is ambiguous between a collective and a distributive reading. In a situation where none of the men lifts a table on his own but the men together lift one table we have to deny sentence (1) in its distributive reading but affirm it in its collective reading. In Roberts (1987, pp. 100) further evidence for the ambiguity between collective and distributive readings is given. Her key evidence is the anaphoric potential of the noun phrase *a table*. In the collective reading sentence (1) can be continued with

(2) It was heavy.

This continuation is impossible in the distributive reading since the distributive reading leads to a universal quantificational force of the noun phrase *three girls*. Roughly speaking this has the effect that the indefinite noun phrase *a table* is in the scope of the distributively interpreted – hence universally quantified – *three girls* thus making it impossible for a pronoun to refer to *a table*. Since in the collective reading the indefinite noun phrase *a table* is not in the scope of a universal quantifier anaphoric reference of *it* to *a table* is possible. A further evidence for the assumption of a distributive (quantificational) reading is that an interaction with the scope of other quantifiers can be observed. The following sentence taken from Link and Schütze (1991)

(3) Applicants have to know at least one programming language.

commonly means that every applicant has to know at least one programming language but this language can be different for each of them. If we do not assume a distributive interpretation of the plural subject noun phrase *applicants* the scope of *at least* may or may not include *applicants*, but in either case we do not get the intended reading where there can be different languages for every applicant.

Like other authors I will therefore assume two incompatible semantic representations for collective and distributive readings, i.e. there is no systematic relationship between the two readings in form of entailments. In discussing the alternative "No Ambiguity Approach" by Verkuyl and van der Does in section 3.3.4 we have seen that the assumption of a weak representation that encompasses both a collective and a distributive reading is problematic for several reasons. Most importantly it allowed for too many verifying situations, e.g. sentence (1) could be true in a situation where two men together lift one table and another man lifts a second table, i.e. there would be two tables involved which is implausible for (1). A further argument for giving distributive readings a separate logical form is that distributive readings can be clearly marked in natural language, e.g. by the floated quantifier *each*. The distributive reading induces universal quantificational force and is thus equipped with a precise logical interpretation. By contrast – as we will see below – the collective mode is mostly vague an indeterminate, even when adverbs like *together* are added.

The examples above that supported the assumption of a separate distributive reading all contained a plural subject noun phrase as external argument and an indefinite direct object as internal argument. Several authors (Verkuyl 1999, Lønning 1987 and 1991) have discussed whether the ambiguity assumption can legitimately be generalized to other structures and argument positions. In particular, it has been claimed that an asymmetry between external and internal arguments of verbs can be observed. For example, in sentence

(4) A man carried three tables upstairs.

an individual denoting plural noun phrase occurs in internal argument position (here as the direct object of the transitive verb *carry*). Sentence (4) can mean that a man lifted three tables one-by-one, as a whole or in any other constellation. Employing ambiguity tests sentence (4) does not seem to be ambiguous between a collective and a distributive reading. For example, applying the ambiguity definition 4 from section 2.4 we find that in a situation where a man first carried two tables upstairs and then another table sentence (4) is simply true in that situation, it cannot be denied. In any situation where three tables get finally carried upstairs the sentence is true, no matter how the carryings actually took place. This observation would imply that sentence (4) is not ambiguous between a distributive and a non-distributive reading but indeterminate in the sense of definition 5 from section 2.4. Other ambiguity tests also seem to suggest an indeterminate reading. The sentence

(5) A girl lifted three tables and so did a man.

is true in a situation where a girl lifted three tables as a whole and a man lifted three tables one-by-one. This possibility of a "crossed understanding" has been used as an argument for indeterminacy. Furthermore, the sentence

(6) A girl didn't lift three tables.

negates all possible table-lifting constellations. The sentence is true if a girl didn't lift three tables as a whole or if the girl didn't lift three tables one at a time. Again, this could be seen as an argument for the indeterminacy of the sentence.

Similar effects occur with one place predicates like *disappear* in sentence

(7) Five men disappear.

Sentence (7) again shows that the dichotomy between collective and distributive readings is not always clear-cut. It is difficult if not impossible to decide whether (7) should receive a collective or a distributive reading. If the individual disappearing events are wholly unrelated we perhaps prefer a distributive reading, if there is some correlation between the disappearings we may prefer a collective reading. As it stands the sentence is indeterminate as to the possible disappearing constellations.

There is, however, a problem in assuming – in general – that internal arguments of transitive verbs or subjects of intransitive verbs are indeterminate with respect to a collective or distributive interpretation. We have addressed this problem already in section 2.3.1.1 with the following examples:

(8) a. She summarized the proposals .
b. John juggled with six plates.
c. Samantha quickly polished the boots.
d. Fred could not lift three tables.

The examples show that in special contexts the collective/distributive distinction still plays a role. What is more, when event variables are introduced in the representation language as is necessary for (8)c the question arises as to what the temporal relations between the various events expressed by a plural sentence are.

I will therefore not assume in general that internal arguments of transitive verbs and external arguments of intransitive verbs are indeterminate but assume both collective and distributive readings as possible readings. That means I assume (like Link) that – in principle – any argument position can be read collectively or distributively, with the consequence that I also face the problem of a combinatorial explosion of *possible* readings. However, there are regularities as to which readings are more plausible in certain constructions. For example, the observation that collective and distributive readings are indistinguishable for a number of predicates can be explained by assuming a collective reading and applying meaning postulates that distribute the predicate down to individual members. Further regularities to reduce possible readings will be dealt with in chapter 5.

The Indeterminacy of Collective Readings

What is common to the collective readings of sentences (4) or (7) is that they describe the situation incompletely. For example in (4) there is no indication allowing one to decide whether the three tables are carried one-by-one, as a whole or in any other constellation. Different constellations do not change the truth-value of sentence (4). In any case, to make the sentence true three tables get finally lifted in a possibly complex event. Thus, sentence (4) fundamentally underinforms the hearer about how the actual situation was realized. This indeterminacy can be intended by the user who does not want to be more precise than necessary in a particular situation – a feature that adds to the efficiency of natural language. For an extensive discussion of this topic I refer the interested reader to articles in Verkuyl (1999).

The indeterminacy can be represented by giving the sentences a collective reading (in Link's sense) and assuming that the actual realization of the relation is dependent on various factors, on the lexical semantics of the verb, on the context, on our world-knowledge etc. For example, the observation that collective and distributive readings are often indistinguishable can be explained by the lexical properties of certain types of verbs. Dowty (1986) for example assumes for certain types of verbs "distributive subentailments" that have the effect that if a predicate applies to a group it also applies to each part of that group. Lønning (1987, pp. 225) calls these predicates "downwardly closed", and Krifka (1989b, p. 78) introduces the concept "divisivity" (among a number of other reference types of predicates). "Divisivity" can be defined as follows (where ≤ means that Y is a part of X).

(9) $\forall p(\mathbf{DIVISIVE}(p) \leftrightarrow \forall X \forall Y(p(X) \wedge Y \leq X \rightarrow p(Y)))$

Divisive predicates prefer a collective reading in the respective argument place. What actually had to happen in the situation is left implicit. Assuming divisivity allows us to reconstruct that sentence (10)a entails sentence (10)b.

(10) a. John carried 4 tables upstairs.
b. John carried each of the 4 tables upstairs.

If we start with a collective reading for (10) and then apply the meaning postulate (9) sentence (10)b can be inferred.

Krifka (1996) demonstrates further interesting examples that show that the level of participation in a collective reading is dependent on the predicate or on the context.

(11) a. I returned to the house because I thought I had left the windows open.
b. But when I came back I found that the windows were closed.

The preferred interpretation of (11)a is that I thought that I had left one or more of the windows open, whereas the preferred interpretation of (11)b is that I found out that all the windows were closed. Starting with a collective reading and assuming the following meaning postulates for *open* and *closed*, resp. one could explain the difference.

(12) a. $\forall X(\text{open}(X) \leftrightarrow \exists Y(Y \leq X \wedge \text{open}(Y)))$
b. $\forall X(\text{closed}(X) \leftrightarrow \neg\exists Y(Y \leq X \wedge \text{open}(Y)))$

However, that lexical information alone cannot be the only reason for the different entailments is demonstrated by the example

(13) a. I could reach the safe because the doors were open.
b. I could not reach the safe because the doors were closed.

In the particular context, (13)a expresses the fact that all the doors were open, whereas (13)b expresses the fact that at least some of the doors were closed. The examples show that total vs. partial participation in the interpretation can shift with the context, it therefore cannot be just a lexical property as suggested by (12). Krifka (1996) proposes a pragmatic rule to explain which level of participation is most plausible. His hypothesis is that in predications on sum individuals, the logically stronger interpretation (an interpretation is logically stronger if it logically entails the weaker interpretation) is preferred unless explicit or implicit information enforces a particular interpretation (e.g. the presence of a floated quantifier like *each*, or lexically enforced or situationally preferred interpretations of the predicate). I will not go into further details here.

The above examples showed that in certain situations – without additional knowledge – it is not possible to spell out the meaning of the collective predication totally in terms of what happened to the individuals. The “dimensions of indeterminacy” (Link and Schütze 1991, p. 350) concern the way how the individuals are involved in the action. Link and Schütze propose three dimensions of indeterminacy for collective readings: (i) “Separate vs. Common Involvement” expresses that objects can participate in the relation alone or together with others. For example, in (10)a tables can be carried alone or together with others. (ii) “Singular vs. Multiple Involvement” expresses that the same individual can be involved more than once in a certain predication. For example in

(14) The men were writing songs. (Gillon 1987)

some individuals can be involved in many song writing events (alone or together with others). (iii) “Few vs. Many Involvement” expresses that depending on the predicate or the context total or partial participation of the group can make the collective reading true. For example, to make (11)a true it is sufficient that at least one window is open, whereas (11)b requires all windows to be closed. Different dimensions of indeterminacy can co-occur in one sentence.

Of course, the assumption of an indeterminate collective reading pushes the problem down to giving an appropriate lexical semantics of verbs. However, depending on the intended granularity of the semantic representation and the corresponding inferences it is very often not necessary to spell out the lexical semantics of verbs and one can do with a coarse analysis in the form of an indeterminate collective reading.

The indeterminacy of the collective reading also explains mixed or partitional readings that have been introduced for sentences like

(15) 6000 students gather all over the country.

Possible gathering constellations are not considered as separate interpretations but we assign one collective, i.e. non-distributive interpretation which is indeterminate as to the possible subgatherings.

Cumulative Readings

Scha (1981) argued that sentences containing two numerical noun phrases like

(16) a. Four men lifted three tables.
b. 600 Dutch firms own 5000 American computers.

can – in addition to a collective and a distributive reading – express a cumulative reading. For example, sentence (16)a can be true in a situation where two men lift two tables and the other two men together lift one table. This verifying situation is different from a genuinely collective interpretation where three men jointly lift three tables as a whole. Similarly, the cumulative reading of (16)b says that the number of Dutch firms that own at least one American computer is 600 and the number of American computers that are owned by at least one Dutch firm is 5000, which again is different from a genuinely collective owning. Cumulative interpretations can also occur with other types of plural noun phrases like

(17) a. John and Mary own three houses.
b. At least three boys bought at most six books.
c. The students were reading the newspapers.
d. Exactly three boys visited exactly four girls.

Sentence (17)a doesn't tell us anything about how the houses distribute to John and Mary, neither does (17)b tell us how the books distribute to the boys. Common to the sentences in (17) is that subject and object noun phrase are autonomously referring terms but the sentences are indeterminate as to the specific relation between the objects denoted by the argument NPs. That means that sentences with a "cumulative reading" allow for different verifying situations, or, in other model-theoretic words, a cumulative reading can have more than one model that satisfies it.

That cumulative readings allow for different verifying situations has been taken as one argument to assume a separate logical form representing cumulative readings. This is, however, not the only solution to explain cumulativity effects. One can as well assume that a cumulative reading has the same semantic representation as a collective-collective reading, the latter being a "special case" of a cumulative reading. I will come back to this below.

According to my opinion, there is no "absolutely" true or false answer to the question whether it is necessary to assume separate logical forms for collective and cumulative readings. If we apply the ambiguity tests of section 2.4.3 to decide whether the difference between collective and cumulative readings is a matter of ambiguity or of indeterminacy we find that the tests are less cogent than for the distinction between distributive and non-distributive readings. Therefore a decision is "relative" in that it depends on what you want to do with the semantic representation. I will address different possible applications after discussing some ambiguity tests.

One ambiguity test checks whether the sentences

(18) a. Four men lifted three tables, so did three women.
b. John and Mary own three houses, so do Bill and Sue.

can have crossed understandings. According to my intuition crossed understandings are possible. For example, (18)b can mean that John and Mary jointly own three houses, and Bill owns one house and Sue owns another two houses. The negation test is less intuitive. Does sentence

(19) It is not the case that John and Mary own three houses.

negate all possible (non-distributive) owning-constellations between John and Mary and the three houses? Also, the truth assignment test does not give a clear answer. Assume a situation where two men lift one table together, and two other men each lift a different table. Can we truly deny sentence (16)a (in its non-distributive reading)? It is difficult to answer this question "neutrally" since when being asked the question we already implicitly elaborate on possible constellations that then are logically incompatible and therefore, if made explicit, would require different logical forms. However, in its original formulation (16)a is indeterminate and most often we do not bother to think about possible constellations. Possible elaborations for (16)a concern the way how men are distributed to tables and vice versa. For example,

(20) Four men jointly lifted three tables as a whole.

elaborates on (16)a and assigns a group of men to a group of tables (the genuinely collective reading). Sentences like

(21) a. In total, four men lifted three tables.
b. Altogether, four men lifted three tables.

indicate that the sentence is verified by a "cumulative situation", but gives no more detail, Whereas

(22) Four men jointly lifted three tables one-by-one.

expresses that the same four men lifted one table after the other, in sum three tables.

The sentences (20) to (22) all entail the indeterminate sentence (16)a, but there is no relationship of entailment between the elaborated sentences (20) to (22). Logically this observation can be modelled by simply giving (16)a a doubly collective reading (in Link's sense) and adding elaborations in the logical form for the genuinely collective reading as in (20) and the "cumulative" reading as in (21).

Typically, spelling out the precise constellations in natural language turns out to be rather clumsy and one often uses tables or diagrams instead. Natural language on the other hand is superior when we deliberately want to leave out unimportant details and express just the information that is appropriate or known in a given situation. For example sentence (16)a simply states that there is a lifting relation and that there are two groups that stand in this lifting relation. How this lifting relation is realized can either be made more precise by elaborating it in the text, or we infer plausible constellations from the meaning of the verb, from our world-knowledge or from other contextual information.

As mentioned above from a practical point of view there are arguments for both positions: assuming extra representations for collective and cumulative readings, or assuming just one collective reading and relying on judgements of entailment. The arguments depend on the final purpose of the logical representation. For example, encoding knowledge from a natural language source in a knowledge base poses different problems than using natural language to express knowledge base queries that operate on an already existing unambiguous database.

Encoding knowledge expressed in natural language in a formal knowledge base (one part of the proof-theoretic view of natural language semantics) requires that for each plural noun phrase one has to decide whether it is interpreted distributively, collectively (or cumulatively). Distinguishing cumulative from collective readings explodes the number of readings. Logically independent readings should only be assumed if there are logical entailments that cannot be otherwise explained. Thus the relevant question is whether there are structural entailments that are triggered by a cumulative interpretation but not by a collective and vice versa. What does sentence (16)a entail in its non-distributive reading? Intuitions vary as we have seen in section 3.3. I agree with many authors that (16)a simply entails that there is a group of four men and a group of three tables and that the two groups stand in the lifting relation, i.e. there is something which brings the four men together in one collection. The following examples indicate that whatever more can be said about plausible constellations depends on our world knowledge, our lexical knowledge, on the context or on other factors all of which are not part of the logical form generated by the sentence taken in isolation.

(23)
a. Four men lifted three pianos.
b. Three boys ate four cakes.
c. Three boys ate four apples.
d. There are three boys and four cakes. The boys ate the cakes.
e. Two students own three cars.
f. The students were reading the newspapers. (Link and Schütze 1991)
g. The guards were watching the prisoners. (Langendoen 1978)

h. The guards were holding the prisoners. (Langendoen 1978)

Assigning a doubly collective reading in Link's sense explains the desired entailments. Cumulative readings have been introduced to explain additional entailments, viz. maximality effects and the participation of all members of a group in a certain relation. However, it has been doubted by many researches that these should be *logical* entailments. As has been argued for example by Krifka (1989a, 1989b, 1999) maximality effects of numerical noun phrases should not – in general – be part of the logical form but be derived as a scalar implicature (see section 3.3.3.2). To be maximally informative the speaker chooses the highest values for the numbers so that the sentence is true. One argument for a pragmatic explanation is that the maximality effect can be cancelled, as is characteristic for implicatures in general. An example for a cancellation is (Krifka 1999):

(24) Three boys ate seven apples, perhaps even eight.

This suggests that maximality should only be part of the logical representation if it is explicitly expressed as in

(25) Exactly three boys ate exactly seven apples.

Furthermore, examples with non-numeric noun phrases like

(26) a. The guards were holding the prisoners. (Langendoen 1978)
b. The ten players scored three goals. (Carpenter 1995)
c. The women released the prisoners. (Langendoen 1978)

demonstrate that assuming a cumulative logical form that requires participation by everyone to make the sentence true is too strong. Whereas the most likely interpretation of (26)a is that each guard is holding at least one prisoner, sentence (26)b does not necessarily involve participation by everyone. Also, (26)c can be true if not every woman participated in releasing one or more prisoners.

For applications that derive knowledge from a natural language source I therefore consider it to be appropriate to assume a (doubly) collective reading in the sense of Link. In this I follow Roberts (1987), Lønning (1987), Link (1991), Verkuyl (1999), Krifka (1989a, 1989b) and others in that the cumulative case does not constitute a separate reading but is a special case of the collective (= non-distributive) reading. In (16)a there is a group of four men and a group of three tables and the two groups stand in the lifting relation, i.e. there is something which brings the four men together in one collection. How the relation can actually be realized depends on many factors (lexical knowledge, context, world-knowledge) that go beyond the structural semantics of the sentence. Thus the logical form of sentence leaves the precise interpretation indeterminate. The elaboration of the particular constellations either has to be made explicit in the natural language source or has to be inferred from lexical or contextual knowledge. With this view I "inherit" all the problems that were discussed in section 3.3.3.2 for Link's approach, in particular upward-monotonicity of the logical representation, but I will show how some of the problems can be solved.

Verkuyl (1999) argues that it is misleading to use the concept “collectivity” for both genuine collectivity and cumulativity, he therefore introduces the concept “totalizing reading” instead. Roberts uses the concept of a “group-group” reading. I will stick with the term “(doubly) collective reading” and distinguish collectivity in a narrow sense (genuine collectivity) and collectivity in a broad sense (cumulativity). The collective reading will be indeterminate as to the possible verifying constellations.

Cumulativity in Knowledge Base Queries

There are, however, other applications where inferences about maximality and full participation might be considered relevant. Take for example an application where the description of the world does not have to be derived from a natural language source but where it already exists in form of an unambiguous database that contains facts of individuals and/or groups. When answering a knowledge base query stated in natural language the task may be simply to find out whether the query is true on any of its possible readings and then, if needed, give an informative answer indicating in which sense the query is true, perhaps by simply listing the facts that satisfied it. Suppose there is a database which contains facts about students and cars which they own. The database may contain the information that John is a student and owns one car, Mary is a student that owns another car, and John and Mary together own a third car. A user who is just interested in the number of cars and students that stand in the owning-relation may query the database with the following question:

(27) How many students own how many cars?

The expected answer is derived from the database by adding up the number of students and the number of cars that stand in the owning relation (not counting the same objects more than once). An appropriate answer will then be

(28) Two students own three cars.

To derive the answer it is not important how each car is owned. A cumulative reading of sentence (28) is sufficient. Only an answer that counts *all* students and cars is appropriate, i.e. maximality is essential. Since there is no *explicit* entry in the database stating that two students own three cars the answer (28) cannot be derived without the assumption of a cumulative reading. To represent knowledge base queries like (28) one might therefore argue that the assumption of a cumulative reading is necessary to derive the expected answers. It has also been argued that to derive a positive answer to the question

(29) Do two students own three cars?

on the basis of the small knowledge base above one needs to assume a separate cumulative reading. However, the stipulation of a cumulative reading is not the only possibility to derive a positive answer to question (29). A cumulative interpretation can also be derived in a more systematic and independently justified way as has e.g. been noted by Krifka (1989b, pp. 107, 1999) and Landman (1998). Krifka assumes for a number of verbal predicates a general meaning postulate which he calls “cumulativity”. For one place predicates cumulativitiy means that if a predicate p applies to an object X and p also applies to another object Y, then p also applies to the sum individual consisting of X and Y, denoted as X⊕Y. For example if *John runs.* is true and *Mary runs.* is true we can infer (via the cumulativity of *run*) that *John and Mary run.* is true. Two place predicates behave similarly, e.g. if *John owns a Mini.* is true and *Mary owns a Porsche.* is true then we can deduce that the sentence *John and Mary own a Mini and a Porsche* is true as well (in its non-distributive reading). Cumulativity can be generalized to n-place relations, in the following way,

(30) $\forall p(\textbf{CUMULATIVE}(p) \leftrightarrow$
$\forall X_1 \ldots \forall X_n \forall Y_1 \ldots \forall Y_n(p(X_1, \ldots, X_n) \wedge p(Y_1, \ldots, Y_n) \rightarrow p(X_1 \oplus Y_1, \ldots, X_n \oplus Y_n)))$

Cumulativitiy allows for inference patterns like the one we had in our database example. Assume s_1 and s_2 are students and c_1, c_2 and c_3 are cars and the database contains the following facts:

(31) s_1 owns c_1, and s_2 owns c_2, and $s_1 \oplus s_2$ own c_3

Then we can infer via (30) that

(32) $s_1 \oplus s_2$ own $c_1 \oplus c_2 \oplus c_3$.

If we assume e.g. Link's representation of plurals, we see that (32) entails

(33) $\exists X(*student(X) \wedge |X| = 2 \wedge \exists Y(*car(Y) \wedge |Y| = 3 \wedge own(X,Y)))$

which corresponds to the representation of the doubly collective reading of the sentence

(34) Two students own three cars.

That means cumulative interpretations can be seen as a side-effect of the cumulativity of predicates and need not be stated as a separate logical form involving quantification over partitions of sets or other complicated formalisms. Syntactically, representation (33) can be straightforwardly derived from (34). A further advantage is that the two noun phrases do not take scope with respect to each other since the two existential quantifiers can be permuted leading to equivalent logical formulae – which was one important property of cumulative interpretations. Furthermore, cumulative interpretations are fairly unspecific under this analysis: From the interpretation given, we cannot infer which student(s) own which car(s). This is as it should be, since cumulative interpretations allow for a wide range of scenarios in which they can be true. If a sentence explicitly expresses genuinely "collective constellations" like in

(35) Two students jointly own three cars.

this has to be made explicit in the logical form by an appropriate semantic analysis of e.g. *jointly*. I will address this below in section 4.5.5. An important difference compared to e.g Scha's representation of cumulative readings in section 3.3.2 is that representation (33) does not exclude that there are more than two students that own cars and more than three cars owned by students. This maximality effect of "cumulative" readings is seen as a result of a scalar implicature.

Theoretically, assuming cumulativity of predicates we can do without a separate cumulative representation for the application in knowledge base queries. The problem is, however, to implement the cumulativity of predicates practically. To perform the following inference based on the cumulativity of *own* we need to know that the owned objects are not identical.

(36) a. John owns one car.
b. Mary owns two cars.
c. Hence, John and Mary own three cars.

This may be plausible for predicates like *eat* or *own*, but less plausible for predicates like *touch* or *see* where it is well possible that the same objects are touched or seen. As long as we have no additional evidence that the owned objects are different we cannot logically infer (36)c. Furthermore, there are practical problem in implementing cumulativity of predicates within an automatic theorem prover or model builder. The application of the cumulative inference rule is highly inefficient. Since rule (30) explicitly generates new sum-objects from existing objects there can occur a combinatorial explosion of new objects which explodes the system.

Since my practical applications mainly aim at deriving knowledge from a natural language source I have neither introduced a separate logical form for cumulative readings nor did I implement the cumulative reference property practically. This does not say that it is not – in principle – possible. Since I combine theorem proving with model building it is possible to use the models generated from a consistent natural language text as a database and give knowledge base queries a logical form that corresponds to a cumulative representation. Covington (1996) gives an introduction how these knowledge base queries can be represented in Prolog.

To give an overview, Figure 3 shows how collective and distributive readings are understood in this thesis.

4.4 The Formal Setting

In the rest of this chapter I will introduce a semantic representation for plural constructions conforming to the set of requirements stated above in section 4.2.4. Before I introduce the concrete representations in section 4.5 I will first provide some formal preliminaries. Since the representations are based on Discourse Representation Theory (DRT) the basic ideas of DRT are repeated. I will then define a "flattened" form of a Discourse Representation Structure (DRS) language used in this study and show how DRSs are translated into first-order logic. The first-order representation is then interpreted with respect to a lattice-structured domain that I will briefly explain.

4.4.1 Basic Ideas of Discourse Representation Theory

In section 3.2.2 I have already discussed some of the basic arguments for introducing discourse semantic approaches. Here I will only give a brief introduction to the basic ideas or Discourse Representation Theory (DRT) which is one representative of discourse semantic approaches.

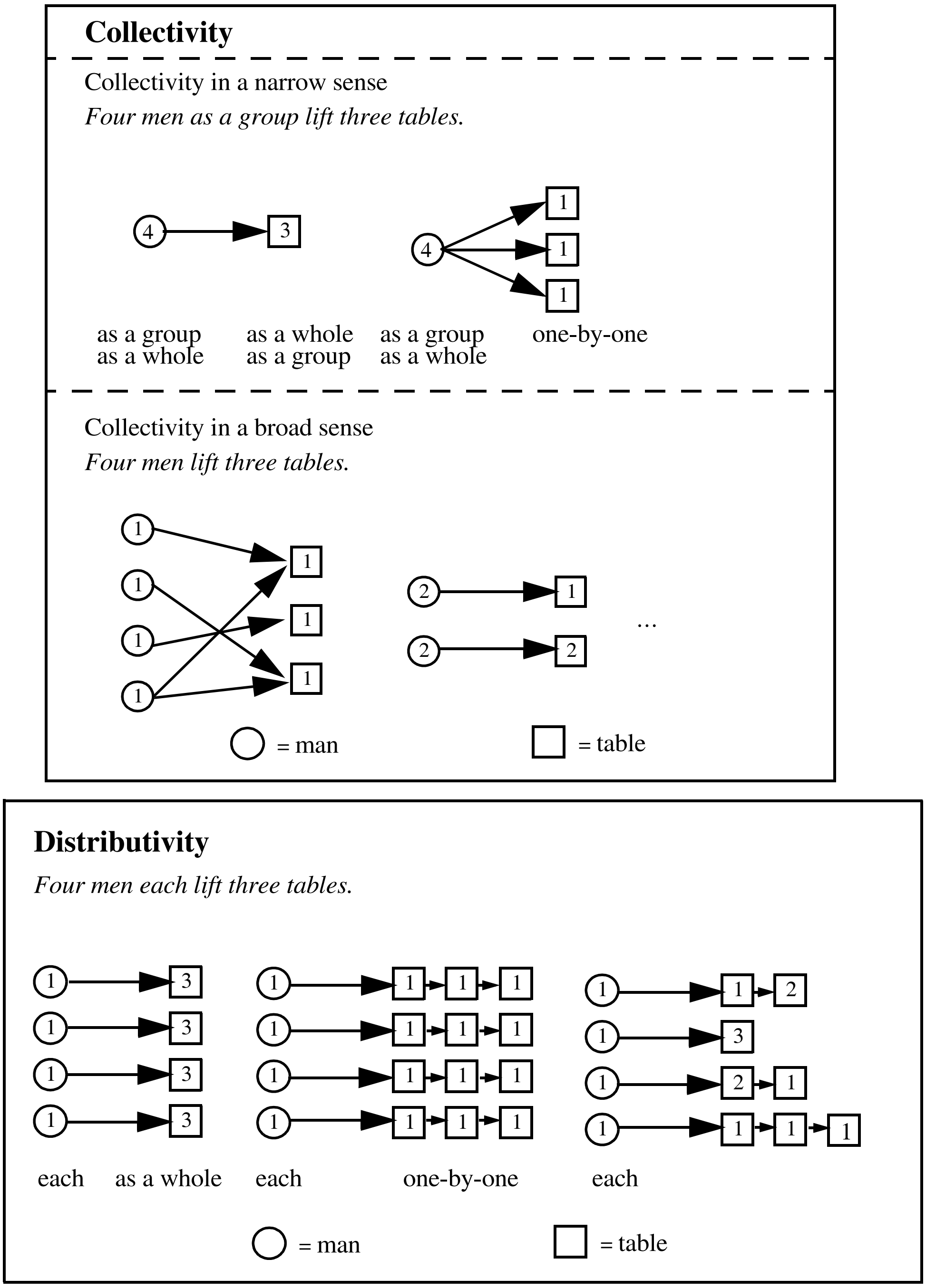


Figure 3 Two Readings of External Argument in '*Two men lift three tables*.'

Correct understanding of natural language texts requires not only processing of individual sentences and their constituents, but also taking into account the way sentences are interrelated to express complex propositional structures. It is well-known that aspects such as anaphoric reference, ellipsis and tense cannot be successfully handled without taking the preceding discourse into consideration. Take for example the discourse

(37) Mary sees a man. He lifts a table.

A naive approach using classical first-order predicate logic would represent the meaning of these two sentences as two separate formulae

(38) $\exists X(\text{man}(X) \wedge \text{see}(\text{'Mary'},X))$
$\exists Y(\text{table}(Y) \wedge \text{lift}(Z,Y))$

The second formula contains a free variable Z for the pronoun *he*. Thus, this representation fails to *resolve* the anaphoric reference of the pronoun *he* in the second sentence to the indefinite noun phrase *a man* in the first sentence. What sentence (37) means would be correctly represented as

(39) $\exists X \exists Y(\text{man}(X) \wedge \text{see}(\text{'Mary'},X) \wedge \text{table}(Y) \wedge \text{lift}(X,Y))$

However, it is hard to define a general post-processing algorithm that converts unresolved formulae like (38) into their final representations like (39). Take for instance the slightly modified discourse

(40) Mary sees John. He lifts a table.

The post-processing algorithm would have to convert

(41) $\text{see}(\text{'Mary'},\text{'John'})$
$\exists Y(\text{table}(Y) \wedge \text{lift}(Z,Y))$

into

(42) $\text{see}(\text{'Mary'},\text{'John'}) \wedge \exists Y(\text{table}(Y) \wedge \text{lift}(\text{'John'},Y))$

The example shows that we would have to define an algorithm that sometimes substitutes a constant for a variable as in (42), while sometimes it has to take into account quantifier scopes as in example (39). These simple discourses indicate that putting information of successive sentences into a single logical unit is a complex process. For anaphora resolution, both quantifier scope and free variable manipulation are necessary. Extending the discourse with further elements like ellipsis, presupposition or tense even more complex algorithms would have to be defined.

A second problem with naive first-order logic approaches to text processing is psychological. Though the first order representations (39) and (42) give adequate truth-conditions for (37) and (40) resp., they do not explain our intuitions how discourse actually works. Uttering the first sentence of (37) we talk *about* a man that is seen by Mary. Thus a new object, a man, is introduced into the discourse and we say that this object becomes *contextually relevant*. The successive discourse will be interpreted taking this changed context into account. This aspect of meaning – not captured by classical first-order predicate logic – is often called *Context Change Potential*. By uttering (37) we do not simply make a claim about the world but we also change the context in which subsequent sentences will be interpreted.

Discourse Representation Theory (DRT) (Kamp 1981, 1984, Kamp and Reyle 1993) was developed as a theory that models the context change potential of utterances in a natural way thus avoiding implausible and unsystematic post-processing steps. Multi-sentential texts are represented as a single logical unit called a discourse representation structure (DRS). Each part of a sentence contributes some logical conditions to the DRS using the preceding sentences as context for example to resolve anaphoric references, ellipsis or presuppositions. Thus information is accumulated in the course of discourse processing.

A DRS is represented by a term `drs(U,Con)` where `U` is a list of so-called discourse referents and `Con` is a list of conditions for the discourse referents. The list `U` stores information about which discourse entities are available, the list `Con` stores information about the properties these entities have and how they are interrelated. In terms of first-order logic the discourse referents correspond to quantified variables and the conditions stand for (open or closed) formulae. Since the beginnings of DRT in 1981 (Kamp 1981) several types of conditions have been introduced. In a commonly used variant the discourse (37) will be translated as

(43) `drs([A,B,C],[A='Mary',man(B),see(A,B),table(C),lift(B,C)])`

graphically represented as:

(44) Mary sees a man. He lifts a table.

```
[A,B,C]
A='Mary'
man(B)
see(A,B)
table(C)
lift(B,C)
```

In the sequel I will use the basic ideas of DRT but will adapt and modify the “standard” types of conditions to meet the above stated requirements of the representation language. The following sections will introduce the modified DRS language.

4.4.2 Definition of a Flattened DRS Language

A discourse representation structure (DRS) `D` is a structure `drs(U,Con)` consisting of a finite set `U` of discourse referents and a finite set `Con` of conditions. A simple term `T` is a constant or a discourse referent. There are primitive conditions and complex conditions. Primitive conditions are formed from a set of predefined relation symbols like `object/2`, `property/2`, or `predicate/4`. Complex conditions are built from other conditions according to the following inductive definition:

Definition 6. Inductive Definition of DRS conditions and DRSs

1. If $X_1, \ldots, X_n$ are discourse referents ($n \geq 0$) and $C_1, \ldots, C_m$ ($m \geq 0$) are conditions then drs([$X_1, \ldots, X_n$],[$C_1, \ldots, C_m$]) is a DRS.
2. If r is a predefined relation symbol of arity n, and $X_1, \ldots, X_n$ are terms then r($X_1, \ldots, X_n$) is a condition.
   The set of predefined relation symbols is defined in Table 15 and Table 16 in the Appendix A.

3. If X_1, …, X_n and Z are discourse referents then sum_of(Z,[X_1,…, X_n]) is a condition.
4. If D is a DRS then -D is a condition.
5. If D_1 and D_2 are DRSs, then D_1 v D_2 is a condition.
6. If D_1 and D_2 are DRSs, then D_1 => D_2 is a condition.
7. Nothing else is a DRS or a condition.

Flattened Representation

The primitive DRS conditions introduced by clause 2 differ from the primitive conditions used in "standard" DRSs like (44) above. Clause 2 states that primitive DRS conditions can only be formed from a predefined set of relation symbols. For example, instead of `man(X)` I introduce the condition `object(X,man)`. The condition uses the predefined relation symbol `object/2` the arguments of which are the discourse referent `X` and the term `man`. Thus `man`, formerly a one-place predicate symbol, is now occurring as a term.

There are several advantages of using a flattened notation. In the flattened notation concepts are treated as individuals meaning that they can now be referred to by terms and do not introduce predicate symbols any more. The most important practical advantage is that terms can be quantified over via first-order quantification whereas quantification over predicates would require higher-order quantification. Thus, general aspects of predicates can be expressed directly in first-order logic. First-order formalizabilty is very important since a natural language understanding system not only generates logical representations for a natural language discourse. It also contains logical axioms that model additional knowledge, and it provides facilities to perform inferences on the basis of the discourse representation and the logical axioms. Both, the formulation of logical axioms and the integration of an efficient deduction module are immensely simplified by the use of a flattened first-order notation. Extensive use of the possibility to quantify over terms representing concepts will for example be made when I define auxiliary first-order axioms to model inferences triggered by plural constructions. Furthermore, the integration of off-the-shelf theorem provers like Otter or Satchmo requires first-order representations. For further arguments concerning the practical advantages of treating concepts as individuals see also Schwitter et. al. (1999).

As an introductory example I give the flattened equivalent of the representation (44). Again I use the graphical representation of the DRS for better readability.

(45) Mary sees a man. He lifts a table. (preliminary)

```
[A,B,C,D,E]
named(A,'Mary')
object(B,man)
structure(C,state)
predicate(C,see,A,B)
object(D,table)
structure(E,event)
predicate(E,lift,B,D)
```

The DRS (45) uses only some of the predefined relation symbols. Other predefined relations will be introduced in the respective sections that discuss the representation of the selected phenomena. A complete list of the predefined relation symbols can be found in Table 15 and Table 16 in the Appendix A.

Note that the DRS (45) will not be the final representation of (37) since additional conditions will be necessary when the fragment is extended with plural constructions. The same holds for the following introductory examples.

Primitive Conditions with Eventualities

Apart from using a flattened notation the DRS (45) differs from (44) in that all verbs are associated with eventualities (events or states). More precisely, each verb introduces an additional discourse referent for eventualities. In our example *see* introduces the state-denoting discourse referent `C`, the predicate *lift* introduces the event-denoting discourse referent `E` (indicated as `structure(C,state)` and `structure(E,event)`, resp.). This treatment of events as kinds of objects that can be talked about goes back to Davidson (1967). A detailed survey of the use of eventualities in the semantics of English can be found in Parsons (1990).

The treatment given here assumes that the distinction between state and event denoting verbs is lexically given. Event verbs denote a change in time whereas state verbs express non-changing properties. Although this strict classification into state and event denoting verbs is a simplification with respect to the semantics of natural language verbs it will be sufficient for the purpose of this thesis (see again Parsons 1990 for a discussion). Introducing discourse referents for eventualities facilitates the representation of verb modification or of the temporal order expressed in a discourse. Take for example the discourse

(46) Mary sees a man. He lifts a table and then he carries a chair into the kitchen.

The discourse expresses modifications of verbs (*into the kitchen*) and temporal order between events (*and then*) which can be represented by interrelating primitive conditions containing discourse referents for eventualities:

(47) Mary sees a man. He lifts a table and then he carries a chair into the kitchen. (preliminary)

```
[A,B,C,D,E,F,G]
named(A,'Mary')
object(B,man)
structure(C,state)
predicate(C,see,A,B)
object(D,table)
structure(E,event)
predicate(E,lift,B,D)
object(E,chair)
object(F,kitchen)
structure(G,event)
predicate(G,carry,B,E)
modifier(G,direction,into,F)   % into the kitchen
temporal_structure(E,before,G) % and then
```

The predefined relation symbol `modifier/4` expresses that the event denoted by `G` is modified by the prepositional phrase *into the kitchen* that expresses the direction of the carrying event. For further modification types see Table 15 in the Appendix A. Furthermore the temporal relation between the lifting and the carrying event can be expressed by the additional DRS condition `temporal_structure(E,before,G)`.

In many event related approaches the arguments of the predicate are pulled out of the predicate and their so-called thematic roles are made explicit, e.g. the condition

```
predicate(C,see,A,B)
```

would e.g. be represented by a coordination of the conditions

```
predicate(C,see) & agent(C,A) & theme(C,B)
```

Each of the thematic roles relates an event to an object. Other postulated thematic roles are e.g. goal, benefactive, experiencer, instrument etc. For a discussion of thematic roles see Parsons (1990, Chapter 5). It is however difficult to determine the appropriate thematic roles for all verbs automatically. If the content of the thematic roles is not needed for inferential processes one could instead replace real thematic roles by “dummy” roles derived from the syntax, e.g.

```
predicate(C,see) & subject(C,A) & direct_object(C,B)
```

The problem with thematic or dummy roles is that it is difficult to distinguish intransitive from transitive verbs in a straightforward way. From the conditions

```
predicate(C,see) & subject(C,A) & direct_object(C,B)
```

it can be logically deduced that

```
predicate(C,see) & subject(C,A)
```

This inference may be desirable for many verbs (e.g. *eat*), it is however not generally valid. From *John enters a card* it cannot be deduced that *John enters*. The difficulty to distinguish transitive from intransitive verbs is also problematic if one wants to state logical axioms that express general properties of verbs, e.g. an axiom that states that if an intransitive predicate that is lexically divisive is true of a group it can be distributed to the parts of that group (from the fact that five men wait we can deduce that each of the five men waits). Such an axiom could tentatively be formalized in first-order logic as:

(48) $\forall P \forall E \forall X(\text{divisive}(P) \wedge \text{structure}(E,\text{event}) \wedge \text{predicate}(E,P) \wedge \text{subject}(E,X) \rightarrow$
$\forall X_1(\text{part\_of}(X_1,X) \rightarrow \exists E_1(\text{structure}(E_1,\text{event}) \wedge \text{predicate}(E_1,P) \wedge \text{subject}(E_1,X_1))))$

Without further additions this axiom – originally intended for intransitive verbs – would also be triggered for transitive verbs classified as divisive yielding inferences that are not generally desirable, e.g. from the fact that five men lift a table one would be able to conclude that each of the five men lifted the table – which is of course not true in the collective reading. The inference could be blocked by classifying certain argument position as divisive but in general the problem does not disappear. To distinguish intransitive from transitive verbs one would have to enrich the logical form with yet another predefined logical relation expressing the syntactic category of the verb, e.g.

```
predicate(C,see) & subject(C,A) & direct_object(C,B) & category(C,tv)
```

Technically, this is of course possible but to avoid an explosion of predefined predicates and a further deviation from the standard familiar notation we prefer to keep the obligatory arguments together with the predicate of the verb. Also, reducing the number of conditions in the DRS has computational advantages with respect to the theorem prover that will be used in this thesis. This decision requires that the we have three predefined predicate relations, viz. `predicate/3` for intransitive, `predicate/4` for transitive and `predicate/5` for ditransitive verbs. The thematic roles are thus kept implicit. If one later finds that it is necessary to express the thematic roles explicitly one could still add the following logical axiom:

(49) $\forall E \forall X \forall Y(\text{predicate}(E,\text{see},X,Y) \rightarrow \text{agent}(E,X) \wedge \text{theme}(E,Y))$

or even more generally the axiom

(50) $\forall E \forall P \forall X \forall Y(\text{predicate}(E,P,X,Y) \rightarrow \text{agent}(E,X) \wedge \text{theme}(E,Y))$

could be formulated.

Complex Conditions

Clauses 4-6 of Definition 6 state that complex conditions can be built from primitive conditions with the logical connectives `-`, `v`, `=>`. The logical operators correspond to standard first-order connectives for negation, disjunction and implication. The logical connective $\wedge$ is implicitly represented by a comma (which is omitted in the pretty printed version of the DRS). Furthermore, in the pretty-printed version we replace the negation `-` by `NOT` for better readability.

(51) If a table is heavy then John does not lift it. (preliminary)

Internal representation

```
drs([A],[named(A,'John'), drs([B,C],[object(B,table),
property(B,heavy),structure(C,state),predicate(C,be,B)]) =>
drs([],[-drs([D],[structure(D,event),predicate(D,lift,A,B)])])])
```

Pretty-printed representation

```
[A]
named(A,'John')
   [B,C]
   object(B,table)
   property(B,heavy)
   structure(C,state)
   predicate(C,be,B)
   =>
   []
   NOT
      [D]
```

```
structure(D,event)
predicate(D,lift,A,B)
```

The connectives `-`, `v`, `=>` lead to “nested” DRSs, i.e. DRSs that are enclosed by other DRSs. For example, the negated DRS is included in the DRS representing the consequent of the implication, and the DRS condition representing the implication is enclosed by the main DRS.

The equivalent of (51) in standard first-order syntax is:

(52) $\exists$A(named(A,‘John’) $\wedge$ $\forall$B$\forall$C(object(B,table) $\wedge$ property(B,heavy) $\wedge$ struture(C,state) $\wedge$ predicate(C,be,B) $\rightarrow$ $\neg\exists$D(structure(D,event) $\wedge$ predicate(D,lift,A,B))))

In section 4.4.3 below I will give a systematic translation of DRSs into first-order logic.

Accessibility

New sentences of a discourse are analysed in the context of preceding sentences. To resolve anaphora a search space for possible antecedents is defined by an accessibility relation among nested DRSs. A discourse referent of a DRS $\mathtt{D}_1$ is *accessible* from a DRS $\mathtt{D}_2$ (i) if $\mathtt{D}_2$ equals $\mathtt{D}_1$, or (ii) if $\mathtt{D}_1$ encloses $\mathtt{D}_2$, or (iii) if $\mathtt{D}_1$ occurs in an implicative condition with $\mathtt{D}_1$ as the antecedent and $\mathtt{D}_2$ as the consequent, viz. $\mathtt{D}_1$ `=>` $\mathtt{D}_2$. Pronouns can only be resolved to accessible discourse referents having the same number and gender restrictions.

If the discourse (51) was extended to

(53) If a table is heavy then John does not lift it$_1$. It$_2$ is not heavy.

the pronoun *it$_1$* in the first sentence can be resolved to *a table* because the discourse referent for *a table* occurring in the antecedent is accessible from the consequent of the implicative condition. However, the pronoun *it$_2$* in the second sentence cannot be resolved since there is no discourse referent with the appropriate gender restriction that occurs in an accessible DRS.

These are only very informal remarks about the concept of accessibility. For more details see Kamp and Reyle (1993) or Blackburn and Bos (2000b).

4.4.3 DRS Interpretation as Translation into First-Order Logic

DRSs can be translated into first-order formula according to the algorithm given in Kamp and Reyle (1993). Here *f* is the translation function that maps DRSs and DRS conditions into their equivalents in first-order logic. Note that a Prolog implementation of this algorithm can be found in Blackburn and Bos (2000b).

Definition 7. Translation of DRSs into First-Order Logic

1. $(\mathtt{drs}([\mathtt{X}_1,\ldots,\mathtt{X}_n],[\mathtt{C}_1,\ldots,\mathtt{C}_m]))^f = \exists X_1 \ldots \exists X_n((C_1)^f \wedge \ldots \wedge (C_m)^f)$
2. $(\mathtt{r}(\mathtt{X}_1,\ldots,\mathtt{X}_n))^f = r(X_1,\ldots,X_n)$

3. $(\texttt{sum\_of(Z,[X}_1\texttt{,}\ldots\texttt{, X}_n\texttt{])})^f = \mathit{sum\_of}(Z,[X_1, \ldots, X_n])$
4. $(\texttt{-D})^f = \neg\, (D)^f$
5. $(\texttt{D}_1 \texttt{ v } \texttt{D}_2)^f = (D_1)^f \vee (D_2)^f$
6. $(\texttt{drs([X}_1\texttt{,}\ldots\texttt{, X}_n\texttt{],[C}_1\texttt{,}\ldots\texttt{, C}_m\texttt{]) => D})^f = \forall X_1 \ldots \forall X_n(((C_1)^f \wedge \ldots \wedge (C_m)^f) \rightarrow (D)^f)$

Clause 1 translates DRSs into formulae of first-order logic. Discourse referents are mapped to existentially quantified variables, and conditions are recursively translated. Clauses 2 and 3 state that primitive conditions are not changed. The list operator used in the term $[X_1, \ldots, X_n]$ corresponds to an n-pace function symbol that forms the sum of its arguments. Complex conditions using negation and disjunction are straightforwardly mapped using clauses 4 and 5, resp. Clause 6 shows how complex conditions using => are mapped. They are different from the other conditions in that discourse referents occurring in the antecedent of an implication are translated into universally quantified variables, the universal quantifiers have scope over both antecedent and consequent. All other discourse referents are existentially quantified. The translation shows that the language of DRS has full first-order strength.

Concerning the semantic interpretation of DRSs we get the result that a DRSs can be satisfied in a given model using a given assignment if and only if its translation into first-order logic can be satisfied in that same model using the same assignment. “It follows that a DRS is valid, consistent or inconsistent if and only if its first-order translation has the same property.” (Blackburn et. al. 2001, p. 19). I will therefore not give a direct model-theoretic interpretation of DRSs but rely on the standard interpretation of first-order logic (see Kamp and Reyle 1993 for more details). The underlying lattice-theoretic ontology of the domain of discourse will be introduced in the next section.

4.4.4 Basic Lattice Theory for Plurals

To represent plural phenomena in DRT we not only have to provide discourse referents that represent individuals but also discourse referents that represent groups of individuals and discourse referents that could represent either. The models with respect to which discourse representation structures (or their first-order equivalents) are interpreted then have to contain both individuals and groups in their domain of discourse. To model this structure I will use a variant of Link’s lattice-theoretical approach (Link 1983) that I have briefly introduced already in section 3.3.3. The lattice-theoretical approach assumes a structured domain of discourse where apart from regular atomic individuals there are complex objects or groups (*individual sums* in Link’s terminology) that serve as the denotation of plural expressions. I have introduced an intuitive explanation of lattices in section 3.3.3 above, here I will give a formal definition of the type of lattice that will be used in this thesis.

Complete Join Semilattices (General)

The semantic representation for plurals assumes a lattice-theoretic structure of the domain of discourse. More precisely, the domain of discourse is structured as a *complete join semilattice*.

Definition 8. Join Semilattice

A join semilattice is a structure ***A*** = <A, part_of> where A is a set of entities, part_of is a partial order of A and any two elements of A have a least upper bound with respect to part_of.

A partial order is a binary relation such that for all X, Y, $Z \in$ A the following axioms hold:

(A1)	$\forall X(\text{part\_of}(X,X))$	reflexivity
(A2)	$\forall X \forall Y \forall Z(\text{part\_of}(X,Y) \wedge \text{part\_of}(Y,Z) \rightarrow \text{part\_of}(X,Z))$	transitivity
(A3)	$\forall X \forall Y(\text{part\_of}(X,Y) \wedge \text{part\_of}(Y,X) \rightarrow X = Y))$	antisymmetry

For any two entities X and $Y \in$ A there has to exist a least upper bound Z $\in$ A with respect to *part_of*:

(A4) $\forall X \forall Y(\exists Z(\text{part\_of}(X,Z) \wedge \text{part\_of}(Y,Z) \wedge \forall V \in \text{A}(\text{part\_of}(X,V) \wedge \text{part\_of}(Y,V) \rightarrow \text{part\_of}(Z,V))))$ Least Upper Bound

In lattice-theory the object Z postulated by (A4) is called *the supremum*. (A3) guarantees that if a supremum exists it is unique. The concept of a supremum is not only applicable to pairs of elements of A but more generally to any subset of A:

Definition 9. Supremum of a Set B

Suppose ***A*** = <A, *part_of*> is a complete join semilattice. Suppose B is a subset of A. Z is *the supremum* of B iff

(A5) $\forall X \in \text{B} (\text{part\_of}(X,Z))$

(A6) $\forall Y \in \text{A} ((\forall X \in \text{B} (\text{part\_of}(X,Y))) \rightarrow \text{part\_of}(Z,Y))$

When such a Z exists it is unique and is denoted as *sup*(B).

We can now define *complete join semilattices*, which is used as the underlying structures to analyse plural phenomena.

Definition 10. Complete Join Semilattices

A join semilattice ***A*** = <A, *part_of*> is called *complete* if for all non-empty subsets B $\subseteq$ A the supremum *sup*(B) exists.

(A7) $\forall \text{B} \exists \text{Z} (sup(\text{B}) = \text{Z})$

Note that the existence of a supremum can only be formalized using a higher order formula.

A lattice-theoretic structure can be used for describing both the semantics of the count domain (*tables*, *chairs*, *books*, …) and the mass domain (*time*, *health*, *money*, …). From a conceptual perspective the denotations of count nouns can be decomposed into indivisible parts, whereas the denotation of mass nouns cannot. The count domain that contains indivisible or *atomic* objects can be characterized by the following definitions.

Definition 11. Atomic Semilattices

Given a complete join semilattice ***A*** = <A, part_of> we call an object $X \in$ A *atomic* if it does not have proper parts.

(A8) $\forall X(\text{structure}(X,\text{atomic}) \leftrightarrow \forall Y(\text{part\_of}(Y,X) \rightarrow X = Y))$

A complete join semilattice ***A*** is called atomic if every object $X \in$ A has atomic parts.

(A9) $\forall X \exists Y(\text{part\_of}(Y,X) \wedge \text{structure}(Y,\text{atomic}))$

The domain of discourse not only consists of objects but also of events. Events are also individuals that can be talked about, individuated or referred to. Events can have a complex structure. For example, playing soccer consists of a number of activities, also lifting a number of tables can be seen as a complex event. The parts of events are, however, more difficult to individuate than the parts of groups. Also, the identity criteria for events are less clear than for objects (Schein 1993, pp. 94). In our ontology the domain of entities is divided into the domain of objects A and the domain of eventualities E, where A and E are disjoint. Like objects, eventualities are structured as a complete join semi-lattice. However, we do not assume that the lattice is atomic (Krifka 1989a, 1989b, Eberle 1998, articles in Rothstein 1998, Schein 1993). The part-structure of events is e.g. important when giving a precise semantics of certain adverbials like *together*, *individually*, *jointly* etc.

Lattice-Theoretic vs. Set-Theoretic Modelling

There are several practical and conceptual reasons motivating a lattice-theoretic approach but I will only give a brief summary here. For a more detailed discussion see Link (1983), Krifka (1989a, 1989b), Kamp and Reyle (1993), Landman (1989), Lønning (1989) and others.

The idea of using complete join semilattices for plural semantics is that apart from atomic individuals (atoms) there are complex individuals (groups) formed by the supremum operation The advantage is that both atoms and groups have the same elementary ontological type; both are individuals of the domain of discourse. While atomic individuals serve as the denotation of non-quantifying singular noun phrases, groups can be denoted by plural noun phrases. No recourse to higher-order objects like sets as denotations of plural expressions is necessary. The categorial uniformity of singular- and plural objects is suggested by many natural language phenomena, for instance a query starting with *who* can be answered by both a singular and a plural noun phrase.

From a formal point of view, an ontologically uniform treatment could also be achieved by using a powerset algebra since each join semilattice <A,part_of> is isomorphic to a powerset structure $<\wp(\text{B}),\subseteq>$ where $\wp(\text{B})$ is the powerset of some given set B and '$\subseteq$' is the subset relation of set-theory. For example, B can be taken to be the set of all atomic elements of A. To achieve categorial uniformity in a powerset structure both singular and plural noun phrases have to denote sets; in particular, singular NPs like *John* or *the book* denote singletons. This categorial "raising" of noun phrase denotations is a technical trick that appears rather counter-intuitive. Furthermore, a technical disadvantage is that sets are higher-order objects that cannot be quantified over in a first-order language. A further motivation for using lattice-theoretic structures is that they explain the structural analogy of the plural and the mass domain. As with plural nouns, the sum of two mass expressions is again a mass expression and not an entity of a different category (often this is called the *cumulative reference property*). The only difference is that lattices modelling the mass domain are not atomic. Furthermore, not only the domain of objects but also the domain of events can be modelled by a lattice-theoretic structure.

Representing Lattice-Theoretic Concepts in the DRS (Practical)

The lattice-theoretic structure of the domain of discourse is partly reflected in a number of DRS conditions. To ease readability in the following section Table 5 on page 108 will give an

Lattice-Theoretic Concepts	**DRS Condition**
domain of objects	`structure(X,dom)`

groups
individuals
mass objects
	`structure(X,group)`
	`structure(X,atomic)`
	`structure(X,mass)`

domain of eventualities	`structure(E,e_dom)`
events states	`structure(E,event)` `structure(S,state)`
part-of relation between objects	`part_of(X,Y)`
proper part of	`proper_part_of(X,Y)`
part-of relation between events	`e_part_of(X,Y)`
equality	`is_equal(X,Y)`
supremum	
$sup(\{X_1,\ldots, X_n\})$ $Z = sup(\{X_1,\ldots, X_n\})$	$[X_1, \ldots, X_n]$ $sum\_of(Z,[X_1, \ldots, X_n])$
maximality (for plural definites)	`maximal(X,context,drs(U1,C1) => drs(U2,C2))`

Table 5 Lattice Theoretic Concepts as DRS Conditions

overview of how the lattice-theoretic concepts introduced above can be “recognized” in the DRS. The domain of discourse is divided into the domain of objects and the domain of eventualities. The domain of objects can be subdivided into groups, individuals and mass objects, the domain of eventualities is subdivided into events and states. Each discourse referent in the DRS is typed according to its ontological status using the predicate `structure/2`. Note that as a technical side-effect this typing of discourse referents will be important for automatic reasoning since some of the theorem provers require variables to be “domain-restricted”. Depending on how much information is available for the respective discourse referents the choice is made between the more general conditions `structure(X,dom)` and `structure(E,e_dom)`, resp., or more specific conditions like `structure(X,atomic)`, `structure(X,group)`, `structure(X,event)` etc.

The ontological hierarchy will be implemented in the inference module (see chapter 6) by assuming additional first-order logic axioms. The ontological hierarchy of the domain of objects includes the following axioms:

(Ax. 1) $\forall X(structure(X,group) \rightarrow structure(X,dom))$

(Ax. 2) $\forall X(\text{structure}(X,\text{atomic}) \rightarrow \text{structure}(X,\text{dom}))$
(Ax. 3) $\forall X(\text{structure}(X,\text{mass}) \rightarrow \text{structure}(X,\text{dom}))$

The domain of eventualities requires the following additional axioms:

(Ax. 4) $\forall X(\text{structure}(X,\text{event}) \rightarrow \text{structure}(X,\text{e\_dom}))$
(Ax. 5) $\forall X(\text{structure}(X,\text{state}) \rightarrow \text{structure}(X,\text{e\_dom}))$

The part-of ordering relation is represented by the two-place relation `part_of/2`. It is a reflexive, transitive and anti-symmetric via the following axioms:

(Ax. 6) $\forall X(\text{part\_of}(X,X))$ reflexivity
(Ax. 7) $\forall X \forall Y \forall Z(\text{part\_of}(X,Y) \wedge \text{part\_of}(Y,Z) \rightarrow \text{part\_of}(X,Z))$ transitivity
(Ax. 8) $\forall X \forall Y(\text{part\_of}(X,Y) \wedge \text{part\_of}(Y,X) \rightarrow \text{is\_equal}(X,Y))$ anti-symmetry

If domain restriction is required the axiom (Ax. 6) has to be formulated for the domain of objects and the domain of eventualities as follows:

(54) a. $\forall X(\text{structure}(X,\text{dom}) \rightarrow \text{part\_of}(X,X))$
b. $\forall X(\text{structure}(X,\text{e\_dom}) \rightarrow \text{part\_of}(X,X))$

The proper part-of relation `proper_part_of/2` is an abbreviation according to the following definition:

(Ax. 9) $\forall X \forall Y(\text{proper\_part\_of}(X,Y) \rightarrow \text{part\_of}(X,Y) \wedge \neg\, \text{is\_equal}(X,Y))$

In the implementation, the reverse direction is not implemented for efficiency reasons.

The following sections show that quantity information for almost all newly introduced discourse referent is made explicit in my representation. Therefore the axiom that groups consist of atomic parts is practically implemented as (Ax. 10)-1. In a number of cases (e.g. for query representation) the cardinality is not made explicit. For these cases the axiom (Ax. 10)-2 is needed:

(Ax. 10) 1. $\forall X \forall C(\text{structure}(X,\text{group}) \wedge \text{quantity}(X,\text{cardinality},C,\text{count\_unit}) \wedge \text{value}(C,\text{geq},2)$
$\rightarrow \exists Y \exists Q(\text{structure}(Y,\text{atomic}) \wedge \text{proper\_part\_of}(Y,X) \wedge \text{quantity}(Y,\text{cardinality},Q,\text{count\_unit}) \wedge \text{value}(Q,\text{eq},1))$
2. $\forall X(\text{structure}(X,\text{group}) \rightarrow \exists Y(\text{proper\_part\_of}(Y,X) \wedge \text{structure}(Y,\text{atomic})))$

The axiom expresses that the domain of countable objects – implicitly restricted by the measurement dimension `cardinality` in `quantity(X,cardinality,C,count_unit)` – is atomic.

The axiom that atoms do not have proper parts is implemented as:

(Ax. 11) $\forall X(\text{structure}(X,\text{atomic}) \rightarrow \forall Y(\text{part\_of}(Y,X) \rightarrow \text{is\_equal}(X,Y)))$

Again, the reverse direction is not implemented for efficiency reasons since it leads to a combinatorial explosion.

Equality is represented by the relation `is_equal/2`. In chapter 6 I will show that practical considerations require to explicitly define the equality relation `is_equal/2` as reflexive, symmetric and transitive via additional first-order axioms:

(Ax. 12)	$\forall$X(is_equal(X,X))	reflexivity
(Ax. 13)	$\forall$X$\forall$Y$\forall$Z(is_equal(X,Y) $\wedge$ is_equal(Y,Z) $\rightarrow$ is_equal(X,Z)))	transitivity
(Ax. 14)	$\forall$X$\forall$Y(is_equal(X,Y) $\rightarrow$ is_equal(Y,X))	symmetry

Again, domain restriction for axiom (Ax. 12) can be achieved by the same technique as in (54) above.

Additionally, for each predefined relation equality substitution axioms have to be formulated. They state that identical entities have the same properties, e.g.

(Ax. 15)
1. $\forall$X$\forall$Y$\forall$O(object(X,O) $\wedge$ is_equal(X,Y) $\rightarrow$ object(Y,O))
2. $\forall$X$\forall$Y$\forall$P(property(X,P) $\wedge$ is_equal(X,Y) $\rightarrow$ property(Y,P))
3. …

Note that equality substitution axioms can be formalized directly in first-order logic due to the flat notation.

For practical reasons I represent the supremum Z of a set of objects B using the list notation, more concretely as `sum_of(Z,B)` meaning `Z` is the sum of the elements of `B`, where `B` is instantiated to a list, e.g. `[X,Y,V,W]`. To model that `Z` is the unique least upper bound I will use list-manipulation operations like flattening and permutation in the practical implementation. Also commutativity, associativity and idempotence of the sum formation are not directly enforced via first-order axioms but more efficiently simulated by list processing operations. The list-notation will be one of the few non-standard first-order language elements. Furthermore, lists – that in Prolog are a special form of functions – will be the only functions allowed in the DRS and their use is highly restricted. Supremum formation will only be triggered by explicit coordination of noun phrases.

For the mass domain we will need an axiom that states that proper parts of mass objects are of the same substance, e.g. a parts of water is still water. This can be encoded by the following axiom:

(Ax. 16) $\forall$X$\forall$Y$\forall$O(structure(X,mass) $\wedge$ object(X,O) $\wedge$ proper_part_of(Y,X)
$\rightarrow$ object(Y,O))

Furthermore, we need an axiom that states that mass objects do have proper parts. This axiom is similar to axiom (Ax. 10) for the count domain:

(Ax. 17) $\forall$W$\forall$D$\forall$Q$_1$$\forall$U(structure(W,mass) $\wedge$ quantity(W,D,Q$_1$,U) $\rightarrow$
$\exists$P$\exists$Q$_2$(structure(P,mass) $\wedge$ proper_part_of(P,W) $\wedge$ quantity(P,D,Q$_2$,U)))

For the practical problems to be solved several other axioms will have to be implemented. We will need axioms for the eventuality domain, axioms that compare the cardinality of objects, and other measurement axioms. Some of these axioms will be introduced in section 4.5, a complete survey is given in chapter 6 where also a discussion of the practical implementation of the axioms is presented. Furthermore, mathematical knowledge will have to be encoded, e.g. the knowledge that if a number `N` is greater than 6 then `N` is also greater than 4. These are non-logical axioms and their practical implementation will be discussed in chapter 6. In the following section mathematical knowledge is assumed to be given.

4.5 A Flat First-Order Representation of Plurals

In the following section semantic representations for important plural phenomena will be introduced using the flat DRS language introduced in the previous sections. A guiding principle in designing the representations was on the one hand to support logically important inferences, on the other hand to avoid that the representations get "too precise" and thus computationally impractical. Furthermore, the representations show how far we can get with flat first-order representations. The current treatment adopts some of the basic ideas of Kamp and Reyle (1993, chapter 4), but also deviates in many aspects for computational and linguistic reasons. The flattened notation will be useful to implement the lattice-theoretic axioms of the previous sections. The representation and processing is based on the following assumptions:

- There are only collective and distributive readings, no separate logical forms for cumulative or mixed readings are introduced.
- Noun phrases themselves are – in general – not ambiguous between collective/distributive or other readings. Each determiner has just one lexical entry.
- Each noun phrase is either individual denoting or quantificational.
- Individual denoting plural noun phrases can occur in sentences with collective readings and can be referred to by anaphora. Quantificational noun phrases do not introduce discourse referents and always trigger universally quantified readings. They do not allow for discourse anaphora.
- Distributive interpretations are achieved by quantifying over the atomic parts of the denotation of individual denoting plural noun phrases.
- Individual denoting noun phrases are classified according to their tendency to be read collectively or distributively.
- Explicitly non-increasing individual denoting noun phrases (e.g. *exactly three men*) require additional maximality conditions in the DRS to prevent upward monotonicity.
- The DRS language does not contain generalized quantifiers. Quantification is always reduced to standard universal and existential quantification.
- DRSs are expressed in a flat first-order language. The list operation is the only function used in the language.
- Each discourse referent is typed according to its ontological status (e.g. `structure(X,atomic)`, `structure(X,event)`).
- Collective and distributive readings are triggered globally.

- Logical inferences about plurals require additional logical axioms.

The assumptions will become more clear in the following sections.

4.5.1 Collective and Distributive Readings – Basics

4.5.1.1 Individual Denoting Noun Phrases

Singular, Plural and Mass Nouns

The representation of individual denoting noun phrases like *a table* or *some tables* is calculated on the basis of the representation for the singular or plural noun and the respective determiner. The basic representation of countable singular nouns like *table* is as follows:

(55) Representation of the singular noun *table*

```
object(X,table)
```

The plural noun *tables* is systematically related to its singular counterpart *table*. The noun *table* is true of individual tables whereas *tables* is also true of all sums that can be formed from individual tables. Recall from section 3.3.3 that in Link's approach the *-operator transforms a predicate `P` that holds of individuals into a predicate that is not only true of those individuals but also of all sums that can be formed from those individuals:

(56) $\forall \mathbf{P}(^*\mathbf{P}(\mathrm{X}) \leftrightarrow \forall \mathrm{U}(\mathrm{structure}(\mathrm{U},\mathrm{atomic}) \wedge \mathrm{part\_of}(\mathrm{U},\mathrm{X}) \rightarrow \mathbf{P}(\mathrm{U})))$

The introduction of a separate pluralization *-operator to represent plural nouns has the disadvantage that the operator does not correspond to standard a first-order syntax. This violates the generality requirement stated in section 4.2.3 according to which the representation should be as close as possible to standard first-order syntax. I therefore represent the pluralization of nouns explicitly in the DRS such that no extra operator and no extra axioms are necessary:

(57) Representation of the pluralized noun *tables*

```
[U]
structure(U,atomic)
part_of(U,X)
=>
[]
object(U,table)
```

Of course, a "flattening" of the *-operator in (56) would also be possible thus avoiding both the introduction of a non-standard operator and the higher-order axiom-schema (56). This flattening could be achieved for example by representing *table* as `object(X,table,sg)` and *tables* as `object(X,table,pl)`. However, the fact that `object(X,table,pl)` denotes an object the atomic parts of which are individual tables (denoted by `object(X,table,sg)`) would still have to be represented by an additional definition along the lines of (56). Representation (57) avoids an additional logical axiom since the desired inference "comes for free".

Mass nouns like *water* or *sugar* are represented as:

(58) Representation of the mass noun *water*

```
[X]
structure(X,mass)
object(X,water)
```

Mass nouns do not have a plural form.

Individual Denoting Noun Phrases

Individual denoting NPs like *a/one table* or *some/two/several tables* introduce discourse referents into the DRS. The discourse referent for a singular NP stands for an atomic object, whereas a plural NP introduces a non-atomic or group discourse referent. In the logical form this difference will be indicated by the conditions `structure(X,atomic)` and `structure(X,group)`, respectively. The preliminary logical forms for *a table* and *some tables* will then be:

(59) a table (preliminary)

```
[A]
structure(A,atomic)
object(A,table)
```

(60) some tables (preliminary)

```
[A]
structure(A,group)
   [B]
   structure(B,atomic)
   part_of(B,A)
   =>
   []
   object(B,table)
```

To obtain a uniform treatment of different types of noun phrases and to simplify logical inferences all types of noun phrases will be associated with quantity information in the final representation. This quantity information can be implicit as for indefinites (*some tables*) or explicit as for cardinality or measurement NPs (*two tables*, *two ounces of gold*). This uniform treatment is a modification of Krifka's (1989a, 1989b) analysis of measure constructions and a variation thereof in Aone (1991). I will discuss the motivation more explicitly when I introduce measurement NPs in section 4.5.2.12 below. Here it is sufficient to know that the quantity information for each NP contains a dimension of measurement (`weight`, `size`, `cardinality`, etc.) and a measurement unit (`ounce`, `cm`, `count_unit`, etc.) which is expressed by the additional DRS condition

(61) `quantity(X,Dimension,N,Count_Unit)`

where `X` and `N` are discourse referents and `Dimension` and `Count_Unit` are constants the value of which is determined by the type of the noun phrase. For measurement phrases like *two ounces of gold* the dimension of measurement (here `weight`) and the measurement unit (here `ounce`) are derived from the lexical entry for *ounce*, yielding the following DRS condition.

(62) `quantity(X,weight,N,ounce)`

For cardinality NPs or indefinites the dimension and count unit is implicitly given by the semantics of the count noun. I name the dimension `cardinality` and the count unit `count_unit` yielding the DRS condition

(63) `quantity(X,cardinality,N,count_unit)`

The discourse referent `X` stands for the object the quantity of which is determined, `N` is a discourse referent denoting a number. Numbers are supposed to be given objects of the domain of discourse and the number-symbols 1, 2, 3, … are assumed to be given as constants in our DRS language. The denotation of `N` is restricted by further conditions. In a standard mathematical notation these additional restrictions would be expressed e.g. as $N = 3.75$ or $N \geq 2$. In our flattened notation these restrictions are encoded by the additional conditions

(64)
```
value(N,eq,3.75)
value(N,geq,2)
value(N,less,2.75)
etc.
```

The third argument of the relation `value/3` always has to be instantiated to a number constant denoting the corresponding number.

We now arrive at the final representations for the indefinite NPs *a table* and *some tables*:

(65) a table
```
[A,B]
structure(B,atomic)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
object(B,table)
```

(66) some tables
```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
   [D]
   structure(D,atomic)
   part_of(D,B)
   =>
   []
   object(D,table)
```

NPs with distributive adjectives modifying the noun are represented as follows.

(67) a young man
```
[A,B]
structure(B,atomic)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
object(B,man)
property(B,young)
```

(68) some young men

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
   [D]
   structure(D,atomic)
   part_of(D,B)
   =>
   []
   object(D,man)
   property(D,young)
```

For collective uses of adjectives see sections 4.5.4.1 and 4.5.6.4 below.

The representation of other types of NPs will be discussed in 4.5.2. To explain the representation of collective and distributive readings one exemplary type of individual denoting noun phrases is sufficient.

At first glance the suggested flat logical representations may look unfamiliar and perhaps unnecessarily complicated. It has to be kept in mind, however, that the representations are not primarily designed for "human consumption" but for the use in natural language understanding systems. This for example involves automatic reasoning on the basis of these representations. As soon as we will discuss these applications the advantages of the proposed logical forms will become more evident.

4.5.1.2 Verbs

In contrast to almost all plural nouns, plural verbs are – in general – not inherently distributive. If a plural verb is true of a group it is not necessarily true of all atomic elements of that group. Therefore the logical representation does not distinguish between singular and plural verbs. Both are assigned a simple DRS condition. We stated above that verbs are associated with events or states which implies that verbs introduce discourse referents into the discourse universe. To distinguish these discourse referents from those introduced by individual denoting noun phrases an eventuality discourse referent is restricted by an additional DRS condition

(69) `structure(E,event)`

or

(70) `structure(S,state)`

The predefined relation symbol for verbs is `predicate/N`, the arity `N` of which depends on whether an intransitive, transitive or ditransitive verb is represented. The first argument of `predicate` always corresponds to the eventuality variable.

(71) Intransitive verbs like *sleep*

```
[A]
structure(A,state)
predicate(A,sleep,X)
```

(72) Transitive verbs like *lift*

```
[A]
structure(A,event)
predicate(A,lift,X,Y)
```

(73) Ditransitive verbs like *give to*

```
[A]
structure(A,event)
predicate(A,give_to,X,Y,Z)
```

The treatment of the copula *be* will be addressed below in 4.5.6.

4.5.1.3 Collective and Distributive Readings

The sentence

(74) Some men lift a table.

containing the individual denoting plural noun phrase *some men* is ambiguous between a collective and a distributive reading. The collective reading expresses that there is a group of men that together lift a table. This meaning can be captured by predicating the verb of the whole group of men yielding the representation:

(75) Some men lift a table. (collective reading)

```
[A,B,C,D,E]
structure(B,group)                     % some
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
  [F]                                  % men
  structure(F,atomic)
  part_of(F,B)
  =>
  []
  object(F,man)
structure(D,atomic)                    % a
quantity(D,cardinality,C,count_unit)
value(C,eq,1)
object(D,table)                        % table
structure(E,event)                     % lift
predicate(E,lift,B,D)
```

Tense information is ignored for the moment. The distributive reading of (74) expresses that there is a group object consisting of individual men, and for every atomic part of that group (that is for every individual man in the group) there is a (possibly different) table that is lifted by that individual man. To represent this reading it is not necessary to assume that the indefinite NP *some men* is ambiguous. The distributive reading can be derived from the collective reading by assuming universal quantification over the atomic parts of the group denoting noun phrase. This renders the following DRS for the distributive reading of (74):

(76) Some men (each) lift a table. (distributive reading)

```
[A,B]
structure(B,group)                       % some
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
```

```
    [C]                                   % men
    structure(C,atomic)
    part_of(C,B)
    =>
    []
    object(C,man)
    [D]                                   % each
    structure(D,atomic)
    part_of(D,B)
    =>
    [E,F,G]
    structure(F,atomic)                   % a table
    quantity(F,cardinality,E,count_unit)
    value(E,eq,1)
    object(F,table)
    structure(G,event)                    % lift
    predicate(G,lift,D,F)
```

The DRS (76) can be simplified by merging the consequents of the two implications that have logically equivalent antecedents taking into account proper renaming of variables:

(77) Logically equivalent alternative to DRS (76):

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
  [C]
  part_of(C,B)
  structure(C,atomic)
  =>
  [D,E,F]
  object(C,man)
  structure(E,atomic)
  quantity(E,cardinality,D,count_unit)
  value(D,eq,1)
  object(E,table)
  structure(F,event)
  predicate(F,lift,C,E)
```

In the following I will stick to the more explicit representations of type (76) and will not perform the simplification in (77).

In section 4.3 I argued that in many cases for the internal argument of transitive verbs collective and distributive readings are indistinguishable. I assumed a collective reading that is indeterminate with respect to the possible realizing situations. The representation of this indeterminate collective reading involves just one event. The sentence

(78) A man lifted some tables.

thus gets the preferred semantic representation – again ignoring tense information:

(79) A man lifted some tables.

```
[A,B,C,D,E]
structure(B,atomic)                       % a
quantity(B,cardinality,A,count_unit)
```

```
value(A,eq,1)
object(B,man)                          % man
structure(D,group)                     % some
quantity(D,cardinality,C,count_unit)
value(C,geq,2)
  [F]                                  % tables
  structure(F,atomic)
  part_of(F,D)
  =>
  []
  object(F,table)
structure(E,event)                     % lift
predicate(E,lift,B,D)
```

It is often not clear whether sentences with plural noun phrases in internal argument position are conceptualized as single events or as a sum of events. Does example (78) introduce one single event of a man lifting some tables or does it introduce some events of a man lifting a table (or the corresponding sum of events respectively)? I represent the sentence as introducing just one event and leave the possible event-structuring indeterminate. The event can have sub-events that are accessed either by additional axioms that hold for certain verb classes or by additional part-structure modifiers like *one-by-one*, *as a whole*, *at the same time* etc. See section 4.5.5.2 for a detailed discussion.

As discussed above there are sentences that require a distributive reading of the internal argument. One reading of the example (due to Parsons 1990)

(80) Samantha quickly polished some boots.

expresses that each polishing is quick. The distributive reading with respect to the internal argument position can be represented as (81). The modifier *quickly* occurs inside the universal quantification introduced by the distributive reading.

(81) Distributive reading of (80)

```
[A,B,C,D]
named(B,'Samantha')
structure(B,atomic)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
structure(D,group)-1
quantity(D,cardinality,C,count_unit)
value(C,geq,2)
  [E]
  structure(E,atomic)
  part_of(E,D)
  =>
  []
  object(E,boot)
  [F]
  structure(F,atomic)
  part_of(F,D)
  =>
  [G]
  structure(G,event)
```

```
predicate(G,polish,B,F)
modifier(G,duration,none,quickly)
```

The somewhat non-standard representation of the modifier *quickly* will be commented in more detail in section 4.5.5.1.

At this point I do not yet determine which factors determine which of the possible readings is actually chosen. Since the factors that determine the reading are so complex I argued for a “global strategy” meaning that the ambiguity is not attributed to a single lexical element but to the global structure of the sentence. This means that neither determiners, noun phrases, verbs nor verb phrases are analysed as being ambiguous but that ambiguity arises when certain elements are combined, e.g. individual denoting plural NPs with non-distributive predicates, or noun phrases with prepositional complements. How this is “global strategy” is realized in the grammar will be discussed in chapter 5. Here it is important to note that lexical elements themselves will not be analysed as being ambiguous between a collective and a distributive lexical entry, they receive just one unambiguous semantic representation.

4.5.2 Noun Phrases

Noun phrases are classified into individual denoting noun phrases and quantificational noun phrases. Individual denoting noun phrases introduce discourse referents into the DRS whereas quantificational noun phrases do not refer to individuals. I adopt this distinction here and will further sub-classify the two groups of noun phrases in the following sections. I will show that the flat first-order DRS language is expressive enough to represent collective and distributive readings and to explain inferences associated with plural noun phrases. Another important point will be to describe the monotonicity properties of many non-standard quantifiers, in particular the non-monotone increasing quantifiers, in a first-order setting.

4.5.2.1 Indefinite Noun Phrases: ‘a’, ‘some’, ‘several’, ‘a few’

Indefinite noun phrases are a subgroup of individual denoting noun phrases. Indefinite noun phrases can be introduced by the determiners *some*, *several*, *a few*, etc. Furthermore, bare plurals like *men* belong to indefinite noun phrases. Also noun phrases like *someone*, *something* with no explicit restrictor belong to the indefinite noun phrases. In my analysis indefinite noun phrases get just one reading. The quantificational use as discussed in section 3.2.2.3 above is neglected.

Representation

The above explained representation for the indefinite NPs *a man* and *some men* are repeated here:

(82) a man

```
[A,B]
structure(B,atomic)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
```

```
object(B,man)
```

(83) some men

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
   [D]
   structure(D,atomic)
   part_of(D,B)
   =>
   []
   object(D,man)
```

There are several possibilities to treat vague indefinite plural noun phrases starting for example with *several*, *a few* etc. depending on the intended preciseness of the semantic representation. The simplest, however, least precise solution is to treat them as equivalent to the determiner *some*. As a result from *Some men wait.* one can then infer *Several men wait.* and *A few men wait*. Also *several* and *a few* could be used interchangeably. If one does not want to support these inferences the vagueness of the determiners can be maintained in the logical form by treating *several* as a property of cardinalities along the following lines:

(84) several men

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
property(A,several)          % several
  [C]
  structure(C,atomic)
  part_of(C,B)-1
  =>
  []
  object(C,man)
```

This logical form prevents the inference from *some* to *several*, or from *several* to *a few*, but it maintains the intended inference from *several*/*a few* to *some*. If in certain application domains vague indefinite determiners should receive a precise reinterpretation it is possible to add logical axioms for example of the following type:

(85) a. $\forall$A(property(A,several) $\rightarrow$ value(A,geq,3))

b. $\forall$A(property(A,a_few) $\rightarrow$ value(A,geq,2) $\wedge$ value(A,leq,15))

The arbitrariness of the axioms shows that this precise reinterpretation only makes sense if there are explicit conventions for certain application domains. If precise interpretations are intended it is, however, advisable to start with more precise formulations directly in the natural language source.

Inferences

Without additional logical axioms the representations (82) and (83) support the right and left upward monotonicity of indefinite plural determiners be it in the collective or the distributive reading. That means, the following inferences come out as logically valid in first-order logic without addition of logical axioms:

(86) *Some* is left monotone increasing
IF Some old men (each) lift a table.
THEN Some men (each) lift a table.

(87) *Some* is right monotone increasing
IF Some men (each) lift an old table.
THEN Some men (each) lift a table.

If we assume representations of type (84) for *several* and *a few* the inference from *some* to *several/a few* falls out automatically, other inferences need additional axioms.

There are further intended inferences that require additional logical axioms, for example, the following inference:

(88) IF Some men (each) wait.
THEN There is a man.

I represent the then-part of (88) as the DRS (82). To perform the inference in (88) in first-order logic we need additional first-order axioms. First, the lattice-theoretic axiom (Ax. 10) that states that from the existence of groups the existence of atomic parts of that group follows. Furthermore we need the definition of the `proper_part_of`-relation in axiom (Ax. 9). Finally, mathematical knowledge is required for the inference. Here we need the knowledge that if a number is greater or equal than 2 it is also greater or equal than 1 (`value(X,geq,2)` implies `value(X,geq,1)`). How this mathematical knowledge is practically implemented in the inference module will be dealt with in chapter 6. Here and in the following sections we take mathematical knowledge about the ordering of numbers as given.

Discussion, Problems and Limitations

By treating indefinites like *some men* as unambiguously individual denoting I only capture the referential reading of indefinites. The systematic ambiguity of indefinites between a referential and a quantificational reading is not explained. For relevant comments and possible solutions I refer to my discussion in section 3.2.2.3 above.

4.5.2.2 Indefinite Noun Phrases: Bare Plurals

Representation

Bare plurals, i.e. plural noun phrases without overt article, occur in a number of functions. They can have existential, universal or generic reading. I will here only deal with the existential reading of bare plurals as in

(89) a. Some students bought books.
b. John bought apples.

In sentence (89)a the bare plural is dependent on another plural noun phrase *some students*. The dependent bare plural need not be understood as denoting a group consisting of more than one atomic part, i.e. the discourse referent introduced by dependent bare plurals is indeterminate with respect to the group-atomic distinction. This indeterminacy of bare plurals can also be observed with non-dependent bare plurals as in (89)b which can – in certain contexts – also be true if John just bought one apple. To model in the DRS that both atomic and group individuals can be denoted by bare plurals I assign the general type `structure(X,dom)`. Thus the bare plural noun phrase *men* is represented as

(90) bare plural NPs (*men*)

```
[A,B]
structure(B,dom)
quantity(B,cardinality,A,count_unit)
value(A,geq,1)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  []
  object(D,man)
```

The DRS for the bare plural *men* is thus equivalent to the representation for *at least one/one or more men*. For a more detailed discussion of (dependent) bare plurals see Kamp and Reyle (1993, chapter 4).

Inferences

Like with the indefinite determiners *some/several/a few* etc. left and right upward monotonicity of existential bare plurals can be explained without additional axioms.

The inference from indefinite plural noun phrases with explicit determiners like *some/several/a few* etc. to bare plural NPs as in

(91) IF John bought some/several/a few apples.
THEN John bought apples.

needs the additional first-order axiom (Ax. 1) plus mathematical knowledge. The reverse inference is not valid. Furthermore, to infer

(92) IF John bought apples.
THEN There is an apple.

we need (Ax. 9) and the following modification of (Ax. 10):

(Ax. 18) $\forall X \forall C(\text{structure}(X,\text{dom}) \wedge \text{quantity}(X,\text{cardinality},C,\text{count\_unit}) \wedge \text{value}(C,\text{geq},1)$
$\rightarrow \exists Y \exists Q(\text{structure}(Y,\text{atomic}) \wedge \text{part\_of}(Y,X) \wedge$
$\text{quantity}(Y,\text{cardinality},Q,\text{count\_unit}) \wedge \text{value}(Q,\text{eq},1))$

There is, however, a problem with the practical implementation of axiom (Ax. 18) in the inference module. Many proofs that should fail do not terminate if (Ax. 18) is added to the inference module. Currently, I have no practical solution to this problem. Therefore the current implementation doesn't contain axiom (Ax. 18) and thus the inference (92) does not come out as valid.

Due to the indeterminate analysis of the bare plural NP we should also be able to reconstruct the reverse inference:

(93) IF John bought an apple.
THEN John bought apples./There are apples.

that is – to simplify matters – we need to deduce from the DRS

(94) There is an apple.

```
[A,B]
structure(B,atomic)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
object(B,apple)
```

the DRS

(95) There are apples.

```
[A,B]
structure(B,dom)
quantity(B,cardinality,A,count_unit)
value(A,geq,1)
  [C]
  structure(C,atomic)
  part_of(C,B)
  =>
  []
  object(C,apple)
```

Besides mathematical knowledge the inference requires additional axioms. First, (Ax. 2) according to which each atom belongs to the domain of objects, second, we need axiom (Ax. 11) according to which atoms do not have proper parts. Furthermore, we need reasoning with equality. For the current inference we will for example use the equality substitution axiom (Ax. 15)-1.

Discussion, Problems and Limitations

In my fragment bare plurals always get an existential reading. I have not investigated other uses of bare plurals. Furthermore, the practical implementation of axiom (Ax. 18) leads to a combinatorial explosion if a proof should fail.

4.5.2.3 Indefinite Noun Phrases: 'someone', 'something'

Representation

The noun phrases *someone/something* that have an implicit general restriction are represented using a general discourse referent without additional quantity information. I represent *someone* and *something* equivalently making no distinction between *someone* denoting a person and something *denoting* an object.

(96) someone/something

```
[A]
structure(A,dom)
```

Leaving out quantity information allows us to infer from both count and mass noun phrases the existence of something/someone.

Inferences

From any individual denoting noun phrase the existence of someone/something can be deduced – if necessary using one of the domain axioms (Ax. 1) to (Ax. 3).

Discussion, Problems and Limitations

A more precise treatment of someone/something should take into account that someone denotes persons and something denotes things. This is however not a principled problem but can be solved by adding object types for discourse variables, e.g. `object_type(X,person)` for persons and `object_type(X,object)` for things.

4.5.2.4 Increasing Cardinality Noun Phrases: 'two', 'at least two', 'more than two'

Representation

In my approach cardinality noun phrases like *two/at least 7/more than two/one ore more men* etc. are classified as individual denoting cardinality noun phrases. As individual denoting noun phrases they introduce discourse referents into the DRS that get an existential reconstruction in first-order logic. Furthermore, the noun phrases imply left- and right upward monotonicity.

The simple cardinality noun phrase *two men* is represented as:

(97) two men

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,eq,2)
  [C]
  structure(C,atomic)
  part_of(C,B)
  =>
  []
  object(C,man)
```

This representation of numerals does not exclude that there are more than two men. As discussed above in section 3.3.3.2 the exactly effect of numerals has to be explained by scalar implicatures.

The determiners *at least two*, *two or more* introduce the DRS condition `value(A,geq,2)` instead of `value(A,eq,2)`, the determiner *more than two* introduces the condition `value(A,greater,2)`, respectively:

(98) at least 2 men/two ore more men

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
  [C]
  structure(C,atomic)
  part_of(C,B)
  =>
  []
  object(C,man)
```

The representation of the singular cardinality NPs *one man* is equivalent to *a man*, viz.

(99) one man

```
[A,B]
structure(B,atomic)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
object(B,man)
```

If the cardinality is vaguely specified by *at least one* or *one or men* we need indeterminate discourse referents marked by `structure(B,dom)`. Thus we get the same DRS as for bare plurals:

(100) at least one man/one or more men

```
[A,B]
structure(B,dom)
quantity(B,cardinality,A,count_unit)
value(A,geq,1)
  [C]
  structure(C,atomic)
  part_of(C,B)
  =>
  []
  object(C,man)
```

The determiner *more than one* is represented like in (100) with `value(A,greater,1)` instead of `value(A,geq,1)`.

Inferences

The representation of *(at least) n* implies left- and right-upward monotonicity both for the distributive and the collective reading. Therefore, inferences of the following type are valid without additional first-order axioms:

(101)	IF	John bought five red apples.
	THEN	John bought five apples.
(102)	IF	Two young men (each) bought an apple.
	THEN	Two men (each) bought an apple.

Inferences of the following type need additional mathematical knowledge.

(103) IF There are (at least) three apples.
THEN There are at least three/at least two/some apples.

(104) IF There are (at least) four apples.
THEN There are more than two/ more than three apples.

The inference from *at least three* to *at least one/more than one* additionally requires the domain axiom (Ax. 1).

To reconstruct the following inference

(105) IF There are (at least) three apples.
THEN There are three/ two apples.

we need – apart from mathematical knowledge – the additional axiom (Ax. 7) that expresses transitivity of the part-of relation, the definition of the proper-part-of relation (Ax. 9), and we need to deduce from the existence of groups the existence of groups with smaller cardinality. This inference requires an additional axiom that states the existence of proper parts of groups:

(Ax. 19) $\forall X \forall C$(structure(X,group) $\wedge$ quantity(X,cardinality,C,count_unit) $\wedge$
value(C,greater,2)
$\rightarrow \exists Y \exists Q$(structure(Y,group) $\wedge$ proper_part_of(Y,X) $\wedge$
quantity(Y,cardinality,Q,count_unit))

Furthermore we need an axiom that states that proper parts have a smaller cardinality than the whole group. This cannot be directly implemented using only first-order logic axioms since it would require an axiom of the following type using mathematical knowledge:

(106) $\forall X \forall C \forall N$(structure(X,group) $\wedge$ quantity(X,cardinality,C,count_unit) $\wedge$
value(C,geq,N) $\rightarrow$ $\forall Y$(proper_part_of(Y,X) $\wedge$ quantity(Y,cardinality,Q,count_unit)
$\rightarrow$
$\exists M$(M < N $\wedge$ value(Q,eq,M))))

A practical implementation of this axiom would require that new numbers have to be inferred from existing numbers, which is not practicable since there are infinitely many possibilities. In chapter 6 I will therefore discuss how this inference can be more efficiently implemented in the automatic theorem-prover Otter and the model-builder Satchmo using built-in arithmetic predicates.

Note also that the inference from the existence of *at least n* to *n* for $n \geq 2$ requires the following axiom:

(Ax. 20) $\forall X \forall C \forall N$(structure(X,group) $\wedge$ quantity(X,cardinality,C,count_unit) $\wedge$
value(C,geq,N)
$\rightarrow \exists Y \exists Q$(structure(Y,group) $\wedge$ part_of(Y,X) $\wedge$
quantity(Y,cardinality,Q,count_unit) $\wedge$ value(Q,eq,N)))

Similar to axiom (Ax. 18) the addition of axiom (Ax. 20) to the practical implementation works if proofs succeed but leads to non-termination for many proofs that should fail. Therefore axiom (Ax. 20) is currently not part of the practical implementation leading to incompleteness concerning the inference from the existence of *at least n* to the existence of *n*.

The representation and the axioms also allow to reconstruct the following inferences. In the distributive reading the following inferences are valid.

(107)	IF	Five boys each buy an apple.
	THEN	A boy buys an apple.
(108)	IF	Five boys each buy an apple.
	THEN	Three boys each buy an apple.

Apart from mathematical knowledge the inference (107) requires the definition of the proper-part-of relation (Ax. 9), and the axiom (Ax. 10). The inference (108) requires the transitivity of the part-of relation (Ax. 7), (Ax. 9) and (Ax. 19). In contrast, from the collective reading of the sentence

(109) Five boys buy an apple.

neither of the consequences in the examples (107) and (108) can be inferred, neither can their collective counterparts be inferred.

Discussion, Problems and Limitations

The treatment of increasing cardinality determiners as individual denoting determiners implies upward monotonicity, i.e. the exactly effect of numerals has to be explained via scalar implicatures. Thus the following description of a situation will not lead to an inconsistency.

(110) a. Four men lifted three tables.
b. Six men each lifted a table.

Cumulativity as expressed for example in

(111) In total, four men lifted three tables.

will be represented by a doubly collective reading. There is the possibility to add maximality conditions to the logical form to exclude that there are other table lifting men. Logically this could be realized by the following DRS for (111).

(112) In total, four men lifted three tables.

```
[A,B,C,D,E]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,eq,4)
  [F]
  structure(F,atomic)
  part_of(F,B)
  =>
  []
```

```
    object(F,man)
  structure(D,group)
  quantity(D,cardinality,C,count_unit)
  value(C,eq,3)
    [G]
    structure(G,atomic)
    part_of(G,D)
    =>
    []
    object(G,table)
  structure(E,event)
  predicate(E,lift,B,D)
    [K,H,I]                         % maximality condition
    structure(K,dom)
      [O]
      structure(O,atomic)
      part_of(O,K)
      =>
      []
      object(O,man)
    structure(H,dom)
      [J]
      structure(J,atomic)
      part_of(J,H)
      =>
      []
      object(J,table)
    structure(I,event)
    predicate(I,lift,K,H)
    =>
    []
    part_of(K,B)
    part_of(H,D)
```

The DRS states that there is a group of four men and a group of three tables that stand in the lifting relation. The maximality condition added expresses that if there are other individuals or groups that lift tables they are part of the group of four men, and if there are other tables that are lifted by men they are part of the group of three tables. This representation excludes upward monotonicity due to the universal quantification in the maximality condition. For reasons discussed in section 4.3 above this form of cumulativity is currently not implemented. The representation is intended to show that the representation of cumulativity would be logically possible in the framework suggested here.

4.5.2.5 Non-Monotone Increasing Cardinality Noun Phrases: ‘exactly two’, ‘at most two’, ‘less than two’, ‘not more than two’

Representation

The representations for the cardinality determiners *two*, *at least two*, *more than five* entail upward monotonicity. Using the same technique to represent *explicitly* non-monotone increasing cardinality quantifiers like *exactly three*, *at most four*, etc. would therefore lead to inadequate results. In section 3.3 I have discussed the problem that non-monotone cardinality quantifiers do have an anaphoric potential and can be read collectively which requires the introduction of a group-denoting discourse referent that gets an existential reconstruction in first-order logic. However, to avoid upward-monotonicity due to the existential quantifier the representation has to be augmented with additional conditions. By adding a universally quantified condition the upward-monotonicity effect can be avoided.

The problem is, however, how exactly this additional condition should look like. There are several complicating issues. In *Generalized Quantifier Theory* determiners like *exactly n*, *precisely n*, *all except n*, *an odd/even number of* are generally classified as neither left monotone increasing nor left monotone decreasing, or more generally as non-monotonic determiners. In contrast, quantifiers like *at most n*, *less than n*, *not more than n* are classified as monotone decreasing. The problem with this classification is that the monotonicity behaviour is difficult to judge on different readings of the cardinality determiners. This problem has been stated by other authors before, e.g. Lønning (1987, p. 205), van der Does (1992, pp. 46) or Link (1998a, pp. 37). The following example which is adapted from van der Does (1992, p. 46) shows the problem for *exactly n*.

(113) a. Yesterday, exactly five boys bought a boat together in the shop.
b. Yesterday, exactly five boys each bought a boat in the shop.
c. Yesterday, exactly 836 people bought boats in the shop.

In sentences (113)a and (113)b the total number of boys that bought a boat in the shop during the day may well be more than five. This is impossible in (113)c where the number of boat buying people is relevant. The collective reading in (113)a is about an unspecified group of exactly five boys, while the distributive reading (113)b counts only boys that bought a boat individually. Both sentences leave the possibility of other groups of boys that bought a boat. In contrast, (113)c counts all people that bought one or more boats. This difference is also one reason for our unclear intuitions about the monotonicity behaviour of the determiner *exactly*:

(114)	IF	Exactly five British boys together bought a boat.
	THEN (?)	Exactly five boys together bought a boat.
(115)	IF	Exactly five British boys each bought a boat.
	THEN (?)	Exactly five boys each bought a boat.
(116)	IF	Exactly 836 British people bought boats.
	THEN (?)	Exactly 836 people bought boats.

A further reason for the difficulties in judging the correctness of the inferences is the systematic ambiguity of cardinality noun phrases between an individual denoting and a quantificational reading (see section 3.2).

The inference (114) for the collective reading seems to be valid, but there are other examples with a collective reading of *exactly n* that are less clear, e.g.

(117)	IF	Exactly seven British boys gathered in the classroom, and exactly five Danish boys gathered in the classroom.
	THEN (?)	Exactly seven boys gathered in the classroom.

Due to the unclear intuitions I will therefore not analyse *exactly n* in the collective reading as monotone increasing. Neither will the representation for the distributive reading be monotone increasing.

Similar problems occur when judging the monotonicity behaviour of *at most n* or *not more than n*. The following examples indicate that *at most n* is not monotone increasing:

(118)	IF	At most five British boys together bought a boat.
	THEN (?)	At most five boys together bought a boat.
(119)	IF	At most five British boys each bought a boat.
	THEN (?)	At most five boys each bought a boat.
(120)	IF	At most 836 British people bought boats.
	THEN (?)	At most 836 people bought boats.

According to my intuition neither of the inferences (118)-(120) should come out as valid.

We said that to prevent upward monotonicity the logical form has to be augmented with additional logical conditions. How should these conditions look like? Link (1998a) points out a further complicating factor in determining these extra conditions, viz. the fact that non-monotone increasing quantifiers are special in that they take scope over the material that follows. This scope varies from context to context as the following example (adapted from Link 1998a) shows.

(121) At most four men went to the bar together and had a beer each.

In one interpretation sentence (121) means that there were not more than four men in the room, i.e. the scope is *men*. The sentence can also mean that there were lots of men in the room but at most four men went to the bar together. In this case the scope is *men that went to the bar together*. Finally, the scope can extend to all of *men that went to the bar together and had a beer each* in which case the sentence means that there were not more than four men in the room that went to the bar together and had a beer each. In this case there could be more than four men that went to the bar together; but only a group of no more than four men was such that each of its members had a beer. From this observation Link concludes that "sentences involving such determiners cannot be evaluated unless the material is specified that goes under their scope." In the semantic representation the additional condition has to specify the scope of the determiner.

To avoid upward monotonicity within a first-order representation we have to add a condition that states the maximality of the NP denotation under consideration. This additional condition may not enforce uniqueness which would lead to unacceptable results for predicates that have distributive subentailments. The additional condition has to guarantee that we pick the "maximal" object for which a relation holds.

For sentences with collective readings I suggest the following representation for *exactly n*:

(122) Exactly five boys gathered. (collective reading)

```
[A,B,C]                              % Five boys gathered
```

```
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,eq,5)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  []
  object(D,boy)
structure(C,event)
predicate(C,gather,B)
  [E,F]                            % Maximality Condition
  structure(E,group)
    [G]
    structure(G,atomic)
    part_of(G,E)
    =>
    []
    object(G,boy)
  structure(F,event)
  predicate(F,gather,E)
  =>
  []
  part_of(E,B)
```

The logical form expresses that there is a group of five boys that gathers and if there are other groups that gather they are part of the first group. The maximality condition in representation (122) shows that the meaning of *exactly five* cannot be given in isolation but needs to take into account the scope of the determiner – which is the whole sentence in our example. This becomes particularly clear for the representation of two place-predicates:

(123) Exactly five boys together bought a boat.

```
[A,B,C,D,E]
structure(B,group)         % five boys together bought a boat
quantity(B,cardinality,A,count_unit)
value(A,eq,5)
  [F]
  structure(F,atomic)
  part_of(F,B)
  =>
  []
  object(F,boy)
structure(D,atomic)
quantity(D,cardinality,C,count_unit)
value(C,eq,1)
object(D,boat)
structure(E,event)
predicate(E,buy,B,D)
  [G,H,I,J]                % Maximality Condition
  structure(G,group)
    [K]
    structure(K,atomic)
    part_of(K,G)
    =>
    []
```

```
      object(K,boy)
    structure(I,atomic)
    quantity(I,cardinality,H,count_unit)
    value(H,eq,1)
    object(I,boat)
    structure(J,event)
    predicate(J,buy,G,I)
    =>
    []
    part_of(G,B)
```

This representation states that there is a group of five boys that together bought a boat and if there are other groups that together buy a boat they are part of the group of five boys. This representation may be considered too strong since it does not allow for groups of more than five boys that buy a boat. As we have said above for the collective readings intuitions about the monotonicity properties quantifiers like *exactly n* are not clear. I therefore chose a stronger representation that allows for less inferences. If one wishes to guarantee upward-monotonicity for the collective reading one has to omit the maximality condition and give the same representation for the noun phrases *five boys* and *exactly five boys* in the collective reading.

For the distributive reading of *exactly n* the maximality condition has to state that there are no other individual boys who buy a boat or no other groups of boys who each buy a boat.

(124) Exactly five boys each bought a boat. (distributive reading)

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,eq,5)
  [C]
  structure(C,atomic)
  part_of(C,B)
  =>
  []
  object(C,boy)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  [E,F,G]
  structure(F,atomic)
  quantity(F,cardinality,E,count_unit)
  value(E,eq,1)
  object(F,boat)
  structure(G,event)
  predicate(G,buy,D,F)
  [H]                                % Maximality Condition
  structure(H,dom)
    [I]
    structure(I,atomic)
    part_of(I,H)
    =>
    []
    object(I,boy)
    [J]
```

```
      structure(J,atomic)
      part_of(J,H)
      =>
      [K,L,M]
      structure(L,atomic)
      quantity(L,cardinality,K,count_unit)
      value(K,eq,1)
      object(L,boat)
      structure(M,event)
      predicate(M,buy,J,L)
    =>
    []
    part_of(H,B)
```

The maximality condition uses a neutral discourse referent indicated as `structure(H,dom)` and states that if there is an object `H` consisting of boys (be it a group of boys or an individual boy) and each atomic part of that object buys a boat then `H` is a part of the group of five boys `B` who each buy a boat. Assuming that parts do not have a larger cardinality than the whole group representation (124) excludes – as desired – that there are more than five individuals boys who buy a boat.

The suggested representations (123) and (124) do not cover the mixed and cumulative "readings" of *exactly n*. Sentence

(125) A boy bought exactly three boats.

has a collective reading which I represent as (126). This representation just counts groups of boats bought by a boy in one buying event. The distributive reading of (125) is represented as (127). Representation (127) counts all atomic boats for which there is a buying event of a boy.

(126) A boy bought exactly three boats. (collective reading)

```
[A,B,C,D,E]
structure(B,atomic)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
object(B,boy)
structure(D,group)
quantity(D,cardinality,C,count_unit)
value(C,eq,3)
  [F]
  structure(F,atomic)
  part_of(F,D)
  =>
  []
  object(F,boat)
structure(E,event)
predicate(E,buy,B,D)
  [G,H]
  structure(G,group)
    [I]
    structure(I,atomic)
    part_of(I,G)
    =>
    []
```

```
    object(I,boat)
  structure(H,event)
  predicate(H,buy,B,G)
  =>
  []
  part_of(G,D)
```

(127) A boy bought exactly three boats. (distributive reading)

```
[A,B,C,D]
structure(B,atomic)                      % a boy
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
object(B,boy)
structure(D,group)                         % three boats
quantity(D,cardinality,C,count_unit)
value(C,eq,3)
  [E]
  structure(E,atomic)
  part_of(E,D)
  =>
  []
  object(E,boat)
  [F]                                      % each bought
  structure(F,atomic)
  part_of(F,D)
  =>
  [G]
  structure(G,event)
  predicate(G,buy,B,F)
  [H]                                      % maximality condition
  structure(H,dom)
    [I]
    structure(I,atomic)
    part_of(I,H)
    =>
    []
    object(I,boat)
    [J]
    structure(J,atomic)
    part_of(J,H)
    =>
    [K]
    structure(K,event)
    predicate(K,buy,B,J)
  =>
  []
  part_of(H,D)
```

To achieve a weaker intermediate representation that would count both groups of boats and individual boats one would have to replace the condition `structure(G,group)` in representation (126) by `structure(G,dom)`. For reasons discussed in section 4.3 above I have not implemented a weak or neutral reading.

The technique of using additional maximality conditions in the DRS can also be applied to the non-monotone increasing determiners *at most n*, *less than n*. Again, collective and distributive readings seem to behave differently. The distributive reading induces the following representation

(128) (Some but) At most five boys bought a boat. (distributive reading)

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,leq,5)
  [C]
  structure(C,atomic)
  part_of(C,B)
  =>
  []
  object(C,boy)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  [E,F,G]
  structure(F,atomic)
  quantity(F,cardinality,E,count_unit)
  value(E,eq,1)
  object(F,boat)
  structure(G,event)
  predicate(G,buy,D,F)
  [H]
  structure(H,dom)
    [I]
    structure(I,atomic)
    part_of(I,H)
    =>
    []
    object(I,boy)
    [J]
    structure(J,atomic)
    part_of(J,H)
    =>
    [K,L,M]
    structure(L,atomic)
    quantity(L,cardinality,K,count_unit)
    value(K,eq,1)
    object(L,boat)
    structure(M,event)
    predicate(M,buy,J,L)
  =>
  []
  part_of(H,B)
```

The main difference to a standard GQT representation of *at most n* is the existential import. The problem with representation (128) is therefore that it does not cover the existence free reading of *at most n*. I have currently no general solution to this problem. One solution consists of translating the existence free reading of *at most n* as *not more than n* as in (129).

(129) At most five boys each bought a boat. =
Not more than five boys each bought a boat. (distributive reading)

```
[]
NOT
  [A,B]
  structure(B,group)
  quantity(B,cardinality,A,count_unit)
  value(A,greater,5)
    [C]
    structure(C,atomic)
    part_of(C,B)
    =>
    []
    object(C,boy)
    [D]
    structure(D,atomic)
    part_of(D,B)
    =>
    [E,F,G]
    structure(F,atomic)
    quantity(F,cardinality,E,count_unit)
    value(E,eq,1)
    object(F,boat)
    structure(G,event)
    predicate(G,buy,D,F)
```

Accordingly, the collective reading would be represented as the negation of the collective reading of *More than five boys together bought a boat*. The corresponding representations do not have existential import. As a side effect the noun phrases lose their anaphoric potential. Both representations (128) and (129) prevent upward monotonicity. As a side effect, (129) models downward-monotonicity whereas (128) does not.

Inferences

The suggested representations for *exactly n*, *at most n*, *less than n* etc. allow for collective besides distributive readings, and – due to the existential quantifier – admit an easy reconstruction of discourse anaphora. The additional maximality conditions prevent left and right upward monotonicity. Thus the following inferences are not valid.

(130)	IF	Exactly/at most/less than three Swiss boys eat a pizza.
	~~THEN~~	Exactly/at most/less than three boys eat a pizza.
(131)	IF	Exactly/at most/less than three boys eat an extra large pizza.
	~~THEN~~	Exactly/at most/less than three boys eat a pizza.

In the GQT-literature quantifiers like *at most n* or *less than n* are additionally analysed as left- and right monotone decreasing, meaning the reverse inferences in (130) and (131) should come out as valid. This downward monotonicity cannot be calculated from the logical forms given in my approach. Only the logical representation for *not more than n* supports left- *and* right downward monotonicity and still prevents upward monotonicity.

The inference from *exactly n* to the simple cardinality counterpart *n* comes out as valid provided the sentences are assigned the same reading.

(132) IF Exactly three boys eat an apple.
THEN Three boys each eat an apple.

The reverse direction is not valid.

Moreover, as with increasing cardinality noun phrases the existence of exactly 7 men entails the existence of (at least) 6, 4, 5 … men using additional axioms as indicated in section 4.5.2.4 above. Analogously, the inference to indefinites like *some* and to bare plurals is valid.

The additional maximality conditions allow to detect inconsistencies of the following type:

(133)
a. There are at most five students. There are seven students.
b. Less than 5 boys eat a pizza. 5 boys each eat a pizza.
c. Exactly seven men are Swiss. Eight men are Swiss.
d. At most five students passed the exam. At least six students passed the exam.
e. Exactly three students together own a car. Four students together own a car.

To detect the inconsistencies we need mathematical knowledge plus a measurement axiom that states that parts "are smaller" than the whole. For the cardinality domain this means that parts have a smaller cardinality than the whole. This axiom can be implemented as:

(Ax. 21)
1. ∀W∀Q1∀N∀P∀Q2(quantity(W,cardinality,Q1,count_unit) ∧ value(Q1,leq,N) ∧ part_of(P,W) ∧ quantity(P,cardinality,Q2,count_unit) → value(Q2,leq,N))
2. ∀W∀Q1∀N∀P∀Q2(quantity(W,cardinality,Q1,count_unit) ∧ value(Q1,less,N) ∧ part_of(P,W) ∧ quantity(P,cardinality,Q2,count_unit) → value(Q2,less,N))

I will show below that (Ax. 21) will be generalized to deal with measurement phrases in general. The measurement dimension (here `cardinality`) and the measurement unit (here `count_unit`) will be replaced by universally quantified variables. The relevant axiom is introduced as (Ax. 35) below.

Furthermore, we need mathematical knowledge, e.g. we need to associate the mathematical relations `eq`, `geq`, `leq`, `less`, `greater` with each other. Here I will only list three axioms necessary to prove the above inconsistencies:

(Ax. 22)
1. ∀V∀N(value(V,eq,N) → (value(V,leq,N) ∧ value(V,geq,N)))
2. ∀V∀N(value(V,leq,N) → ¬ value(V,greater,N))
3. ∀V∀N(value(V,less,N) → ¬ value(V,geq,N))
4. …

To deduce for example the inconsistency in (133)a. we need axioms (Ax. 21)-1, (Ax. 22)-1 and (Ax. 22)-2. Furthermore, we need to encode the mathematical knowledge that `value(V,geq,7)` and `value(V,eq,5)` are contradictory. This can be done via the above axioms and the knowledge that if `value(V,geq,G)` where `G` is a number then for any number `N` that is smaller than `G` it holds that `value(V,greater,N)`. Again, we cannot directly formalize this knowledge as a first-order logic axiom like

(134) $\forall V \forall G \forall N(\text{value}(V,\text{geq},G) \land N < G \rightarrow \text{value}(V,\text{greater},N))$

since (134) takes recourse to the mathematical comparison `N > G`. In chapter 6 I will show how (134) and corresponding mathematical knowledge can be implemented practically.

To detect the inconsistency in (133)b we need to apply axioms (Ax. 22)-1,(Ax. 22)-3 and (Ax. 21)-2.

DRSs generated from non-monotone increasing quantifiers tend to be very long due to the added maximality condition that "copies" the whole material. A problem is that complicated DRSs like the ones generated by (133)d make inferences very inefficient. In particular, distributive readings cause problems. For example, in the current implementation the consistency check (133)d does not terminate within a reasonable time limit. The collective readings (133)e cause no problems. I will discuss this problem in more detail in chapter 6. Inconsistencies involving *not more than n* are more efficiently deduced:

(135) a. Not more than two men are hungry. (At least) five men are hungry.
b. Not more than five students passed the exam. At least six students passed the exam.

Note that no inconsistencies can be deduced if one of the sentences is read distributively and the other sentence is read collectively. Therefore in the following examples

(136) a. Exactly three students together own a car. Four students each own a car.
b. Not more than two students together eat an extra large pizza. Three students each eat an extra large pizza.

no inconsistencies can be calculated.

Discussion, Problems and Limitations

In the above representations the maximality conditions did not contain any special conditions of events, in particular I did not state in the maximality condition that events have to be part of the larger event. In Krifka (1989b) the role of events in the representation of negation and quantification is discussed, since negation and quantification have been considered to pose particular problems for event semantics. The problem with negation is that depending on the size of the situation a sentence can be true or false, e.g. the sentence *No one slept.* can be true in a certain situation (say, on the left side of a classroom) but false in a larger situation (say, in the whole classroom). To handle negation correctly Krifka therefore suggests to take into account situations which are "large enough" (Krifka 1989b, p. 100). He therefore introduces the concept of a 'maximal event', that is the fusion of all events at a certain time. The sentence *John didn't sleep.* then refers to a maximal event that doesn't contain an event of John's sleeping. In Krifka (1989b, pp. 101) a formal definition of maximal events is given. Krifka also uses maximal events to explain the semantics of non-monotone increasing quantifiers like *Less than three girls sang*. That means the sentence cannot be evaluated locally but we have to take into account maximal events. The sentence *Less than three girls sang.* is considered to be true of "maximal events that contain a singing of one or two girls at most (or contain no singing of girls at all, a limiting case that should be filtered out by a pragmatic rule)" (Krifka 1989b, p. 105).

In my treatment I have not introduced maximal events explicitly. I consider it not to be necessary since in my setting the maximal situation is implicitly given by the description of the situation as a natural language text. The situation is nothing else but what is described by the set of all sentences. Evaluation of a sentence takes place with respect to *all* sentences in the text or at least with respect to all sentences of a certain paragraph of the text (and possibly with respect to a given set of axioms). That is, sentences are always evaluated with respect to a maximal situation and the maximality conditions guarantee that all involved objects are checked.

Many authors feel that monotone-decreasing quantifiers are intrinsically distributive and therefore ignore collective readings with these determiners altogether (e.g. Kamp and Reyle 1993). And among the authors who discuss the possibility of collective readings there is no agreement as to how this reading is exactly represented. For example, for *at most n* van der Does (1993, p. 530) proposes genuine plural quantification over groups (see Link 1987, reprinted as chapter 4 in 1998b, for a discussion). According to van der Does the collective reading of *at most n* in (137)a is paraphrased as (137)b and would have to be represented as (137)c in the DRT format proposed here:

(137) a. At most four heroines came together.

b. All collections of heroines which came together contained at most four heroines.

c. DRS

```
[]
  [E,F,G]
  structure(F,group)
  quantity(F,cardinality,E,count_unit)
  value(E,geq,2)
    [H]
    structure(H,atomic)
    part_of(H,F)
    =>
    []
    object(H,heroine)
  structure(G,event)
  predicate(G,gather,F)
  =>
  []
  value(E,leq,4)
```

The problem with this representation is that it does not allow for easy reconstruction of discourse anaphora since the representation has no existential import.

Another suggestion (e.g. Krifka 1989b) is to modify the maximality condition so as pick up the largest group with the desired property and require that all other groups with the same property have a smaller cardinality. In the DRS (138) the maximality condition is augmented by quantity information and in the consequent we have the condition `relation(G,leq,A)`.

(138) At most five boys together bought a boat.

```
[A,B,C,D,E]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,leq,5)
  [F]
  structure(F,atomic)
  part_of(F,B)
  =>
  []
  object(F,boy)
structure(D,atomic)
quantity(D,cardinality,C,count_unit)
value(C,eq,1)
object(D,boat)
structure(E,event)
predicate(E,buy,B,D)
  [G,H,I,J,K]                          % Maximality Condition
  structure(H,group)
  quantity(H,cardinality,G,count_unit)
  value(G,geq,2)
    [L]
    structure(L,atomic)
    part_of(L,H)
    =>
    []
    object(L,boy)
  structure(J,atomic)
  quantity(J,cardinality,I,count_unit)
  value(I,eq,1)
  object(J,boat)
  structure(K,event)
  predicate(K,buy,H,J)
  =>
  []
  relation(G,leq,A)
```

The maximality condition of DRS (138) is weaker than the maximality condition proposed in (123) since (138) only requires the cardinality of other groups to be smaller whereas (123) requires other groups to be part of the largest group. That means (138) allows for altogether more than five boys, e.g. there can be many different groups of five members that together buy a boat. The maximality condition suggested in (123) excludes such a situation. There is just one maximal group that together buys a boat. For the distributive reading we definitely need the stronger maximality condition, for the collective reading intuitions are not clear. If we adopt the weaker maximality condition in (138) we have to replace the axioms (Ax. 21) by the following axioms (Ax. 23) to derive inconsistencies of type (133).

(Ax. 23)
1. $\forall N \forall M \forall N1 \forall M1(\text{value}(N,\text{leq},M) \wedge \text{value}(N1,\text{leq},M1) \wedge \text{relation}(N,\text{leq},N1) \rightarrow \text{value}(N,\text{leq},M1))$
2. $\forall N \forall M \forall N1 \forall M1(\text{value}(N,\text{leq},M) \wedge \text{value}(N1,\text{less},M1) \wedge \text{relation}(N,\text{leq},N1) \rightarrow \text{value}(N,\text{less},M1))$
3. $\forall N \forall M \forall N1 \forall M1(\text{value}(N,\text{geq},M) \wedge \text{value}(N1,\text{leq},M1) \wedge \text{relation}(N,\text{leq},N1)$

$\rightarrow$ value(N,leq,M1))
4. $\forall$N$\forall$M$\forall$N1$\forall$M1(value(N,geq,M) $\wedge$ value(N1,less,M1) $\wedge$ relation(N,leq,N1)
$\rightarrow$ value(N,less,M1))

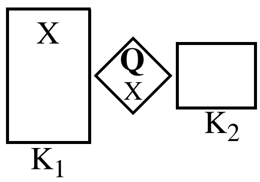
I have addressed above that it is not at all clear what goes under the scope of non-monotone increasing cardinality quantifiers, that means, what should be part of the maximality condition. In the current implementation the whole sentence belongs to the scope of the quantifier, that is the whole sentence is "copied" into the maximality condition – which results in rather large logical forms that are not particularly efficient to process. However, this problem cannot be satisfactorily avoided:

> The interpretation of quantifiers is a complex area of semantics, and one's simple, elegant notions of how the information in sentences can be represented run up against difficulties as soon as quantifiers are considered. Everyone who examines quantifiers is obliged to introduce substantial complexities into their logical notation to accommodate them. (Hobbs 1996, p. 75)

I have also addressed the problem that the representation of *at most n*, *less than n* carries an existential import that is not adequate in all examples. As said above I suggest to use *not more than n* if one wants to guarantee existence free formulae. Furthermore, downward monotonicity that may be desired at least for the distributive reading of *at most n*/*less than n* cannot be modelled.

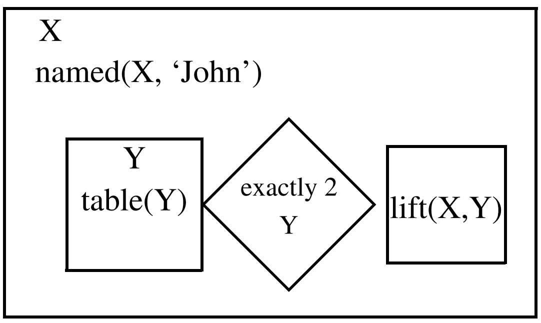
For the sake of completeness I want to briefly show how Kamp and Reyle (1993, chapter 4.4.4) deal with complex cardinality quantifiers. Generalized quantifiers are basically represented via duplex conditions along the following lines using their notation for DRSs:

(139) Duplex Conditions in Kamp and Reyle (1993)



(140) a. John lifts exactly two tables.

b.



To explain the possibility of discourse anaphora and the co-occurrence with existential *there*-constructions Kamp and Reyle use set abstraction to make available a non-atomic discourse referent η that anaphora can be resolved to:

(141) a. John lifts exactly two tables.

b.

X η
named(X, 'John')

η = ΣY: [Y | table(Y), lift(X,Y)]

|η| = 2

However, since abstraction is always over atomic discourse referents complex cardinality quantifiers can never get a collective reading in this approach – unless they are assumed to be ambiguous between a collective and a distributive reading as in Scha's approach. A further problem with Kamp and Reyle's approach is that it is not at all clear what triggers and what constrains abstraction. In my approach these noun phrases are classified as individual denoting from the beginning, which allows for collective and distributive readings. A further advantage of my approach is that I use only standard first-order notation making the application of existing theorem provers easier.

4.5.2.6 Context-Dependent Cardinality Quantifiers: 'many', 'few'

Representation

In section 3.2.1.3 I discussed that *many* and *few* are ambiguous determiners with a cardinal and a proportional reading. Furthermore, the interpretation of *many* and *few* was shown to be vague and context-dependent and several contextual factors determining the meaning of *many* and *few* were addressed. Remember that *many* and *few* are vague with respect to exactly which number is considered to be many and few, and the determiners are context-dependent in the sense that the standard of what counts as many/few varies from context to context.

In my fragment I will give the construction *few/many men* a cardinal reading. The partitive constructions *few/many of the men* will receive a proportional reading. The proportional reading will be discussed below in section 4.5.2.10. In the cardinality reading what counts is the cardinality of the intersection of the noun meaning and the verb phrase meaning. The treatment of *few/many men* as a cardinality quantifier explains the possibility of collective readings and of discourse anaphora, furthermore the occurrence in *there*-insertion sentences is easily explained. I will analyse *many* in its cardinal reading as a monotone-increasing quantifier, and *few* as a non-monotone quantifier. The context dependency and vagueness will not be resolved but left underspecified via the additional conditions `property(A,many)` and `prop-`

`erty(A,few)` that express that the value of `A` is relatively high/low with respect to the context. If desired, this context-dependency can be resolved by additional first order-axioms. For example, in a certain domain a cardinality may be considered as many if it is at least 80% of the size of the domain. Correspondingly, a cardinality may be considered as few if it is less or equal than 20% of the size of the domain. Assuming the size of the domain is given to be 100 *many* can be reinterpreted as denoting a number larger or equal than 80, and *few* as denoting a number less or equal than 20. This can be express in the following first-order axioms:

(142) a. $\forall$A(property(A,many) $\leftrightarrow$ value(A,geq,80))
b. $\forall$A(property(A,few) $\leftrightarrow$ value(A,leq,20))

That means in its cardinal reading *many* corresponds to *at least m* and *few* corresponds to *at most n* where *m* and *n* are contextually determined numbers.

Since *many* is considered to be an increasing quantifier the representation of the distributive reading will be (143) where the condition `property(A,many)` is added.

(143) Many men lifted three tables. (distributive reading)

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
property(A,many)                        % vagueness of many
  [C]
  structure(C,atomic)
  part_of(C,B)
  =>
  []
  object(C,man)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  [E,F,G]
  structure(F,group)
  quantity(F,cardinality,E,count_unit)
  value(E,eq,3)
    [H]
    structure(H,atomic)
    part_of(H,F)
    =>
    []
    object(H,table)
  structure(G,event)
  predicate(G,lift,D,F)
```

The representation of the collective reading is straightforward.

The representation of the distributive reading of the non-monotone determiner *few* adds the condition `property(A,few)` and – to explain non-monotonicity – employs a maximality condition analogous to *at most n*.

(144) Few men lifted a table. (distributive reading)

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
property(A,few)
  [C]
  structure(C,atomic)
  part_of(C,B)
  =>
  []
  object(C,man)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  [E,F,G]
  structure(F,atomic)
  quantity(F,cardinality,E,count_unit)
  value(E,eq,1)
  object(F,table)
  structure(G,event)
  predicate(G,lift,D,F)
  [H]                                % Maximality condition
  structure(H,dom)
    [I]
    structure(I,atomic)
    part_of(I,H)
    =>
    []
    object(I,man)
    [J]
    structure(J,atomic)
    part_of(J,H)
    =>
    [K,L,M]
    structure(L,atomic)
    quantity(L,cardinality,K,count_unit)
    value(K,eq,1)
    object(L,table)
    structure(M,event)
    predicate(M,lift,J,L)
  =>
  []
  part_of(H,B)
```

The representation of the collective reading of *few* is debatable. As Roberts (1987, pp. 180) points out the truth conditions of

(145) Few persons agree (on this issue).

cannot be paraphrased as ‘there are few groups of people who agree’ since one would run into the *proportion problem* for plural quantification (Kratzer, p.c. to Roberts). Assume a situation with four people, a, b, c, and d, such that three of them, a, b, and c, agree on the issue and the fourth, d, doesn’t agree with any of the others. The number of non-atomic objects generated by the sublattice of a, b, c, and d is eleven. If we partition these eleven group objects into those who agree and those who don't we get the following result: the class of groups who agree will contain the objects a⊕b, a⊕c, b⊕c, and a⊕b⊕c, the class of group objects who don't agree includes seven objects, denoted by a⊕d, b⊕d, c⊕d, a⊕b⊕d, a⊕c⊕d, b⊕c⊕d, and a⊕b⊕c⊕d. Thus there are more objects who don't agree than objects who agree, despite the fact that three out of four atomic individuals agree. This doesn't reflect our intuitive understanding of the proportion of agreement among the group of four atomic individuals. Generalizing this observation we will find that "no matter what the number of atomic individuals of the relevant sort who agree with each other in some way, the number of i-sums whose members are not all in agreement will always be equal or greater than the number of i-sums whose members are all in agreement" (Roberts 1987, p. 182). This is what Kratzer calls the proportion problem for plural quantification. The problem shows that to obtain the correct meaning of (145) not all groups are relevant. Rather, we check the cardinality of the *maximal* collection of people who agree. Knowing that number we decide by some contextually given measure whether this number is 'few'. In my notation I can represent this meaning buy adding a maximality condition analogous to (138) above. At the same time this maximality condition prevents upward monotonicity.

(146) Few persons agree.

```
[A,B,C]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
property(A,few)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  []
  object(D,person)
structure(C,event)
predicate(C,agree,B)
  [E,F,G]                         % Maximality condition
  structure(F,group)
  quantity(F,cardinality,E,count_unit)
  value(E,geq,2)
    [H]
    structure(H,atomic)
    part_of(H,F)
    =>
    []
    object(H,person)
  structure(G,event)
  predicate(G,agree,F)
  =>
  []
  relation(E,leq,A)
```

The DRS (146) expresses that the cardinality of the largest group of persons who agree is few. In representation (146) we do not quantify over all groups that are in the lattice but only over a restricted type of groups, viz. those which are a group of persons who agree on the same issue. Again the cardinality that counts as few can be differently determined. As Roberts points out both the cardinality and the proportional reading of *few* can be represented as (146). On the cardinality reading we have a small number in mind and we check if the maximal number of persons who agree is less or equal than this number. On the proportional reading of *few* we have some idea of what constitutes a small proportion of agreers relative to some other – contextually given – set. In either reading, the cardinality of the maximal collection is assigned the property *few*. For a detailed discussion of the representation of *few* and *many* in different frameworks I refer the interested reader to Roberts (1987, section 3.2.4)

Inferences

Due to the analysis of *many* as monotone increasing the following inference is valid

(147) a. IF Many Dutch climbers reached the summit.
THEN Many climbers reached the summit.
b. IF Many climbers reached the summit before 12 o'clock.
THEN Many climbers reached the summit.

Since *many* is both ambiguous and context-dependent the validity of the inferences is not as cogent as with other cardinality quantifiers since it is not clear whether *many* has a cardinal or a proportional reading. And on a proportional reading of *many* the inference (147)a is intuitively not valid since the antecedent of (147)a can have a whole lot of meanings. It may mean e.g. that the number of Dutch climbers who reached the summit is high relative to the number of all Dutch climbers who attempted the summit that day, or it may mean that the number of Dutch climbers who were successful is high with respect to all climbers that day, or it may mean that the number of successful Dutch climbers is high relative to the daily average of Dutch climbers who reached the summit in the respective climbing season, etc. In neither of the readings we want to infer that many climbers reached the summit. Even on a cardinality reading left-monotonicity is not always valid. Right monotonicity of *many* is less problematic as the inference in (147)b indicates. Since it is very difficult – if not impossible – to derive the meaning of *many* automatically I decided to give *many* a rather weak meaning. One advantage is for example that given the following sentences

(148) Many Dutch climbers reached the summit. Many Swiss climbers reached the summit.

we can derive a positive answer to the question

(149) Did many climbers reach the summit?

by retrieving both sentences of (148) leaving the exact interpretation of *many* to the user.

Further inferences licensed by the representation of *many* are that from the antecedent of (147)a it can be deduced that at least two climbers reached the summit. Adding an axiom of type (142)a we would also allow us to deduce that at least 80 climbers reached the summit (using further axioms as discussed above).

Due to the added maximality condition *few* is non-monotone increasing. In some frameworks *few* is analysed as monotone-decreasing. For the same reasons as discussed for non-monotone cardinality quantifiers downward-monotonicity is not modelled by the given representation for *few*.

The following inference shows that the suggested representation of *few* has existential import:

(150) IF Few climbers reached the summit.
THEN At least one climber/at least two climbers reached the summit.

It can be criticised that the inference from *few* to *at least two* is too strong. If only the deduction *at least one* from *few* should be allowed one would have to alter in representation (144) the conditions `structure(B,group)` and `value(A,geq,2)` to `structure(B,dom)` and `value(A,geq,1)`, resp.

Assuming the axioms (142) we can derive inconsistencies for example of the following type.

(151) Inconsistent
a. Many men wait. Less than 75 men wait.
b. Few men wait. At least 31 men wait.

(151)a can be proven to be inconsistent in the collective reading using axioms (142)-a, (Ax. 22)-3, (Ax. 21)-2. The distributive reading additionally requires axiom (Ax. 1). The sentences (151)b can be proven to be inconsistent in the distributive reading using axioms (142)-b, (Ax. 1), (Ax. 22)-2 and (Ax. 21)-1. In the collective reading (151)b is inconsistent using axioms (142)-b (Ax. 23)-3, (Ax. 22)-2.

Furthermore we can derive the inconsistency

(152) Inconsistent
Few men wait. Many men wait.

If the axioms (142)-b and (142)-a are present the inconsistency in the collective reading can be proven by additionally applying (Ax. 23)-3 and (Ax. 22)-2, and inconsistency in the distributive reading requires additionally (Ax. 22)-2 and (Ax. 21)-1. If axioms (142)-b and (142)-a are not present we need an additional axiom explicitly relating the quantities expressed by *few* and *many*. In the collective reading we need axiom (Ax. 24), in the distributive reading we need – besides axiom (Ax. 1) – the axiom (Ax. 25).

(Ax. 24) $\forall N \forall N1(\neg(\text{property}(N1,\text{few}) \wedge \text{relation}(N,\text{leq},N1)) \wedge \text{property}(N,\text{many}))$

(Ax. 25) $\forall W \forall Q1 \forall P \forall Q2(\text{quantity}(W,\text{cardinality},Q1,\text{count\_unit}) \wedge \text{property}(Q1,\text{few}) \wedge \text{part\_of}(P,W) \wedge \text{quantity}(P,\text{cardinality},Q2,\text{count\_unit}) \rightarrow \neg\text{property}(Q2,\text{many}))$

Note also, that – due to the upward-monotonicity of *many* – the following sentences (153) come out as inconsistent

(153) Inconsistent

a. Many young men wait. Few men wait.

b. Many young boys each buy an apple. Few boys buy an apple.

whereas the sentences in (154) are consistent

(154) Consistent

a. Few young men wait. Many men wait.

b. Few young boys each buy an apple. Many boys buy an apple.

The representation of *few* in the suggested setting has existential import and does not imply downward monotonicity. A possibility to avoid existential import and guarantee downward monotonicity is to use *not many* instead of *few*:

(155) Not many men lifted a table. (distributive reading)

```
[]
NOT
  [A,B]
  structure(B,group)
  quantity(B,cardinality,A,count_unit)
  value(A,geq,2)
  property(A,many)
    [C]
    structure(C,atomic)
    part_of(C,B)
    =>
    []
    object(C,man)
    [D]
    structure(D,atomic)
    part_of(D,B)
    =>
    [E,F,G]
    structure(F,atomic)
    quantity(F,cardinality,E,count_unit)
    value(E,eq,1)
    object(F,table)
    structure(G,event)
    predicate(G,lift,D,F)
```

The representation of the collective reading is straightforward. The representations of *not many* guarantee downward monotonicity, i.e. the following inferences are valid without additional axioms.

(156) a. IF Not many men lifted a table.
THEN Not many men lifted a heavy table.

b. IF Not many young men lifted a table.
THEN Not many men lifted a table.

Furthermore, the following examples show the relationship between *few* and *not many*.

(157) a. IF Few men wait.

THEN Not many men wait.

b. IF Not many men wait.
~~THEN~~ Few men wait.

Due to the existential import of *few* the inference (157)b is not valid, however, the inference (157)a is valid using the additional axiom (Ax. 24) in the collective reading, and the axioms (Ax. 1) and (Ax. 25) in the distributive reading.

Thus we can see that even if the context dependency of *few* and *many* is not resolved certain inferences can still be reconstructed using a representation of *few*, *many*, *not many* as presented above.

Discussion, Problems, Limitations

Concerning the problems of interpreting *few* as a cardinality quantifier basically the same comments hold as for treating *at most n/less than n* as cardinality quantifiers. Concerning the practical implementation performance problems related to the distributive reading of the non-monotone quantifier *few* apply accordingly

The above sections only introduced a cardinality reading of *few* and *many*. I will show in section 4.5.2.10 (where I analyse partitive constructions) how a proportional reading of *few* and *many* can be represented given sentences like *few/many of the men wait*.

4.5.2.7 Proportional Quantifier: 'most'

Representation

In contrast to *many* and *few* the determiner *most* hardly gets an absolute cardinality reading but is almost always interpreted as a proportional quantifier (see section 3.2.1.3). Remember that proportional quantifiers presuppose the existence of a (contextually salient) base set with respect to which the intersection of noun and the verb phrase denotation is compared.

(158) Most climbers summited.

To determine the meaning of (158) we need to calculate the proportion between the cardinality of the climbers who summited and the cardinality of some contextually salient set of climbers. The practical difficulty lies of course in determining what exactly this base set is. It is not very likely that the base set consists of the set of all existing climbers. There are several possibilities how the base set is given. In all cases the base set is dependent on a particular situation which – in my approach – is established via a number of sentences. The base set always has to be determined with respect to the given set of sentences. Beyond that there are several possibilities how the base set can be established. First of all it can be determined anaphorically, meaning the relevant base set is mentioned before. This includes direct mentioning (exactly the same head noun is used) or indirect mentioning (the antecedent uses e.g. synonyms of the head noun). An example for a direct mentioning of the base set is

(159) 61 climbers attempted Everest. Most (of the) climbers summited.

Here the base set consists of the set of 61 climbers.

Second, the base set can be given deictically, that is it is salient in the current utterance situation or is otherwise commonly known. For example, uttered in a situation where a group of climbers approaches the Base Camp the base set of *most men* in (158) is just this set of approaching climbers.

Furthermore, the base set can be explicitly introduced by additional textual information, e.g. additional attributes or restrictive relative clauses. This is likely in partitive constructions like

(160) Most of the climbers who attempted Everest on the 16th of May summited.

In (160) the base set is explicitly introduced via the definite noun phrase modified by a restrictive relative clause.

There are further possibilities how the base set can be given. An automatic determination of the base set is very difficult and goes beyond the scope of this work. Here I will only deal with directly anaphorically given base sets and newly introduced base sets. No matter how the base set is determined an adequate representation of proportional quantifiers requires the existence of discourse referents representing the base sets. There are several possibilities how base sets can be practically represented. If the base set is anaphorically given as in (159) the logical representation has to link the previously introduced base set with the group that is relevant for the current predication. As an abbreviated representation of (159) I suggest the DRS (161):

(161) 61 climbers … Most (of the) climbers summited. (distributive reading)

```
[A,B,C,D, …]
structure(B,group)                      % 61 climbers
quantity(B,cardinality,A,count_unit)
value(A,eq,61)
  [E]
  structure(E,atomic)
  part_of(E,B)
  =>
  []
  object(E,climber)
...
structure(D,group)
quantity(D,cardinality,C,count_unit)
value(C,geq,2)
part_of(D,B)                       % part_of relation to base set B
relation(C,most,A)                 % proportion bw. cardinalities
  [F]
  structure(F,atomic)
  part_of(F,D)
  =>
  [G]
  structure(G,event)
  predicate(G,summit,F)
```

The part of the DRS (161) that represents *Most climbers summited.* expresses that there is a group object `D` with the cardinality greater or equal than two, `D` is part of the anaphorically given group `B` of 61 men (`part_of(D,B)`) and `D`’s cardinality `C` is most of the cardinality of `A` (`relation(C,most,A)`). Furthermore, for each atomic part `F` of `D` there is a summiting event `G`. Thus the proportional reading is expressed by relating the two groups `B` and `D` via the `part_of` relation and by relating their respective cardinalities by the vague relation `relation(C,most,A)`. Which proportion counts as 'most' is left vague, in many frameworks it gets a precise reinterpretation as e.g. 'larger than 0.5'. Additional axioms can be added that relate the various vague determiners with each other. I will list these axioms as (Ax. 26)-(Ax. 34), pp. 168.

If the base set is not anaphorically given it has to be introduced locally. However, simply adding an existentially quantified noun phrases consisting of a set of climbers is not sufficient since it would not prevent upward monotonicity, i.e. from *Most Dutch climbers summited.* we would be able to conclude *Most climbers summited.* which is an inadequate inference. Thus an additional condition has to be added that prevents upward monotonicity. There are several possibilities to add this condition. We can add a maximality condition that states that the base set is maximal with respect to all climbers in the current domain of quantification resulting in the following DRS for sentence (158).

(162) Most climbers summited. (version 1)

```
[A,B,C,D]
structure(B,group)            % Base Set B
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
  [E]
  structure(E,atomic)
  part_of(E,B)
  =>
  []
  object(E,climber)
structure(D,group)
quantity(D,cardinality,C,count_unit)
value(C,geq,2)
part_of(D,B)                 % part_of relation to base set B
relation(C,most,A)           % proportion bw. cardinalities
  [F]                        % distributive reading
  structure(F,atomic)
  part_of(F,D)
  =>
  [G]
  structure(G,state)
  predicate(G,summit,F)
  [H]                        % the base set of climbers B is maximal
  structure(H,dom)
    [I]
    structure(I,atomic)
    part_of(I,H)
    =>
    []
    object(I,climber)
  =>
  []
  part_of(H,B)
```

The DRS (162) explicitly introduces a discourse referent for the base set and states its maximality with respect to the noun denotation, that means all objects that consist of climbers (in a particular situation) are part of the base set. The maximality condition prevents left upward monotonicity of *most* while preserving right-monotonicity. The problem is how to restrict the domain of quantification of the maximality condition. As stated above we do not want to count all existing climbers but only a set of climbers that is relevant or salient in a particular situation. Any new occurrence of *most climbers* in the same situation and the same context would again introduce a new base set with a new maximality condition. Logical inference will prove the two maximal sets to be equivalent but the inference itself is very costly.

Instead of asserting an explicit maximality condition one can add the base set (plus its semantic specification) to some form of contextually salient context set. Used in the same context, any occurrence of proportional determiners with the same restrictor refers to the same base set. This base set has to be introduced only once when it occurs first. All further occurrences are anaphorically related to the first occurrence. The current implementation of this idea is preliminary since it uses logical forms that do not conform to the standards of the intended semantic language as specified in section 4.2.4 above. More concretely, arguments are complex terms. I will nevertheless indicate one possible implementation. For the first occurrence of a proportional determiner I added the following condition (163) instead of the maximality condition in (164):

(163)
```
maximal(B,context,drs([H],[structure(H,dom),
                           drs([J],[structure(J,atomic),part_of(J,H)])
                           =>
                           drs([],[object(J,climber)])])
                       =>
                       drs([],[part_of(H,B)]))
```

That means I simply copy the maximality condition and treat it as an argument of the predicate `maximal/3` which is relative to some context. Any further occurrence of a proportional quantifier with the same restrictor in the same context is treated by resolving its base set with the base set `B`. No further base sets have to be introduced. In the practical implementation this approach leads to more efficient inferences since quantification over the whole domain of entities for any occurrence of proportional determiners is avoided. In this approach left upward monotonicity is prevented since a different restrictor like in *most Dutch climbers* would introduce a different maximality condition that doesn't match with the maximality condition introduced by *most climbers*.

As indicated above this solution is preliminary since it uses non-standard first-order logic formulae and its theoretical foundations and implications are not well-investigated. In particular, the approach is not suitable for a model-theoretic approach to semantics. The latter would require something like (162). For the current application based on proof-theoretic semantics, however, inferences concerned with plural proportional quantifiers are more efficiently implemented via the preliminary solution using the condition `maximal/3` than by the solution (162).

The problem in general is that there is no simple way to automatically determine the base set in a certain context. The advantage now is that using the same noun phrase in the same context means that the same base set is referred to without, however, explicitly resolving this base set. We will encounter analogous problems with the representation of (plural) definite descriptions below.

The proportional determiner *most* strongly prefers a distributive reading but can – under certain circumstances – combine with collective predicates.

(164) Most men gathered.

Kamp and Reyle (1993, section 4.4.6.) try to explain these occurrences of *most* with so-called collective predicates like *gather* by reducing the supposedly collective predicate to predicates that can be true of individual elements, e.g. *gather* is reduced to *come to a certain place*. Semantically this intuition may be correct but since such a lexical decomposition is practically difficult to implement I simply adopt a “normal” collective reading. I abbreviate the maximality condition is abbreviated as `maximal(B,context,'men')`):

(165) Most men gathered.

```
[A,B,C,D,E]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
  [F]
  structure(F,atomic)
  part_of(F,B)
  =>
  []
  object(F,man)
structure(D,group)
quantity(D,cardinality,C,count_unit)
value(C,geq,2)
part_of(D,B)
relation(C,most,A)
structure(E,event)
predicate(E,gather,D)
maximal(B,context,'men')       % abbreviated condition maximal/3
```

Both the distributive and the collective reading of *most* introduce (apart from the base set) an additional discourse referent that can be used in collective predication or that can be picked up by discourse anaphora. The status of the exact reference of possible anaphora to noun phrases like *most men* is controversial. For an overview see Geurts (1997, pp. 87-94).

Inferences

The representations (162)/(163) and (165) explain that *most* is right-upward monotone as in (166) but not left-upward monotone as in (167).

(166)	IF	Most boys buy an red apple.
	THEN	Most boys buy an apple.
(167)	IF	Most hungry boys eat a pizza.

~~THEN~~ Most boys eat a pizza.

From the representation of *most men* we can deduce *some men* and *at least two men* by simply assuming transitivity of the `part_of` relation expressed in axiom (Ax. 7). The inference from *most* to the cardinality reading of *many* should – according to my intuition – not be valid.

(168) IF Most men wait.
~~THEN~~ $\text{Many}_{\text{card}}$ men wait.

However, an inference to the proportional reading of *many* is valid.

(169) IF Most men wait.
THEN $\text{Many}_{\text{prop}}$ (of the men) wait.

The proportional reading of *many* will be discussed in section 4.5.2.10 below. Here I will only list an additional axiom (170) that will be necessary for the deduction in (169). This axiom will be repeated below as (Ax. 30).

(170) $\forall N \forall M(\text{relation}(N,\text{most},M) \rightarrow \text{relation}(N,\text{many\_of},N))$

Further inferential relations between the quantifiers *all*, *most*, *more than half of the*, $\textit{many}_{\text{card}}$, $\textit{many}_{\text{prop}}$ etc. follow in the respective sections below.

Discussion, Problems, Limitations

In my approach noun phrases with *most* are treated referentially, i.e. they introduce discourse referents into the domain of discourse that can be referred to by anaphora or used in collective predication. This referential approach facilitates a first-order approximation of the meaning of *most* and its associated inferences. There are problems with determining and representing the base set and two possible solutions have been addressed above. Evaluating (162) over domains containing infinitely many climbers would require to calculate the cardinality of all (i.e. infinitely many climbers), which is not possible. The non-standard solution (163) is less problematic in this respect. However, since descriptions of application domains most often describe a finite domain the solutions given are less problematic. The solution (162) still leads to practical problems since e.g. monotonicity inferences based on representation (162) lead to efficiency problems as soon as lattice theoretic axioms are added.

The meaning of the “collective” use of *most* is – as addressed above – debatable. Also the introduction of discourse referents that can be referred to by anaphora not always complies with our intuitions. However, giving a precise semantics of *most* in different contexts is beyond the scope of this investigation and the given representation is a good compromise between linguistic precision and computational tractability. Furthermore, I assume that the given representation for collective uses of *most* is superior to abstraction techniques as suggested by Kamp and Reyle for reasons discussed in 3.3.2.2 and 4.5.2.5.

4.5.2.8 Definite Descriptions: 'the'

Representation

Definite descriptions (noun phrases with the definite article *the*) occur frequently and in a number of different uses. Accordingly, describing the semantics of definite descriptions is a research topic on its own and the literature about this topic abounds. Poesio and Vieira (1998) and Vieira and Poesio (2000) give a good recent summary of different approaches with a focus on computational applications.

Vieira and Poesio (2000) summarize the possible uses of definite descriptions in three categories. Definite descriptions can be used as *direct anaphora*, as *bridging descriptions* or as introducing *discourse new* entities. When definite descriptions are used as *direct anaphora* they refer back to an antecedent in the discourse such that description and antecedent have the same head noun.

(171) There were 53 canisters at Camp IV. The [53] canisters had been set aside for the May 10 climb.

When definite descriptions are used as *bridging descriptions* they have a discourse antecedent denoting the same discourse entity without having the same head noun. The relation can be established via synonyms as in (172) with the pair *the climbers … the mountaineers*, via generalizations (hypernyms) as in *an oak … the tree* or sometimes through specialization (hyponyms) as in *a tree … the oak*.

(172) 13 climbers reached Everest. The mountaineers returned before dark.

Bridging descriptions also occur when definite descriptions relate to an entity already introduced in the discourse by a relation other than identity, e.g. the description is part of the antecedent as in *a car … the wheel*.

However, not all definite descriptions depend on the previous discourse for their interpretation but refer independently. These uses – that are similar to the use of proper nouns – are called *discourse new*. The reference of discourse new definite description can be given deictically, i.e. the descriptions refer to a salient entity in the physical environment (e.g. *the butter* in a breakfast situation), some are used for objects that are assumed to be generally known (e.g. *the pope*), and some are used with explanatory modifiers that explicitly establish the referent. Explanatory modifiers are e.g. restrictive relative clauses as in

(173) a. The climbers who summited were Dutch.
b. Remove the two screws securing the exhaust manifold to the cylinder exhaust port.

Further explanatory modifiers are NP complements (*the fact that John climbed Everest*), of-constructions (*the summit of Mount Everest*, *the chambered edges of the cylinder exhaust port*), nominal modifiers (*the colour white*, *the number 13*) and others.

Vieira and Poesio (2000) discuss possible heuristics to resolve direct anaphora, do anchor bridging definite description and to detect discourse-new descriptions. In particular the automatic and efficient resolution of bridging descriptions is very difficult. In my fragment I will only deal with the representation and resolution of direct anaphora and with discourse-new definite descriptions. In particular the representation of discourse-new plural definite descriptions is intricate.

Definite descriptions that function as *direct discourse anaphora* are processed by equating the discourse referent introduced by the antecedent with the discourse referent for the definite description. Thus the noun phrases in the discourse (171) are (after simplification of the DRS) represented as:

(174) There were 53 canisters … The [53] canisters …

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,eq,53)
  [C]
  structure(C,atomic)
  part_of(C,B)
  =>
  []
  object(C,canister)
```

As for *discourse new* singular definite description a representation as existentially quantified objects (analogue to indefinite NPs) is not sufficient since wrong inferences, e.g. left upward monotonicity would be generated. We would e.g. be able to make the following invalid deduction:

(175) Invalid Inference

IF The British climber summited and the Dutch climber did not summit.

~~THEN~~ The climber summited.

There are a number of proposals to represent discourse new occurrences of definite descriptions. For singular definite descriptions the Russellian account proposes to add uniqueness conditions into the logical representation that are intended to guarantee that the definite description uniquely specifies a single individual (and that prevent left upward monotonicity). A definite description like *the man* then means there is a man and all other men are identical to that man. This could be expressed in the DRS as

(176) The man waits. (Russellian account)

```
[A,B,C]
structure(B,atomic)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
object(B,man)
structure(C,state)
predicate(C,wait,B)
  [D]                      % uniqueness condition
  structure(D,atomic)
  object(D,man)
```

```
=>
[]
is_equal(D,B)
```

If the uniqueness condition is not satisfied (i.e. if there are more individuals with the specified property) sentences will simply come out as false. It has been objected that uniqueness should be *presupposed* instead of *asserted*, and uniqueness failure should therefore not lead to plain falsity but to a presupposition failure. Furthermore, "global" uniqueness for definite descriptions is too strong and uniqueness should rather be seen as a local concept. That means a certain object is unique with respect to a local domain of objects that is salient when the sentence is used and not with respect to all existing objects. Thus a more realistic account of (singular) definite descriptions should take into account that definite descriptions single out a unique object relative to a given context set. It is however difficult to determine salient context sets. In my setting the sentences of a coherent paragraph would construct the context set. To represent that definite descriptions are dependent on the context and to avoid universal quantification over the whole domain of men I use – in the practical application – the same technique as with the proportional quantifier *most* in (163) above. I copy the uniqueness condition and treat it as an argument of the predicate `maximal/3` which is relative to a context.

(177) The man waits.

```
[A,B,C]
structure(B,atomic)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
object(B,man)
structure(C,state)
predicate(C,wait,B)
maximal(B,context,drs([D],[structure(D,atomic), object(D,man),
             object_type(D,person)])=>drs([],[is_equal(D,B)]))
```

As with *most* this approach prevents upwards monotonicity and leads to more efficient automatic inferencing. Again the proviso applies that DRS (177) does not conform to the ideal requirements for semantic representations as stated above and that it is not easily applicable to model-theoretic database applications.

For mass nouns the representation of the maximality condition would be

(178) the water

```
[A]
structure(A,mass)
object(A,water)
maximal(A,context,drs([C],[structure(C,mass),
                object(C,water)])=>drs([],[part_of(C,A)]))
```

Discourse new *plural* definite descriptions tend to denote the maximal object satisfying the property in the restrictor. Under this perspective the plural noun phrase *the Dutch climbers* refers to the maximal group such that every Dutch climber (in a certain context) is part of that group. For collective readings predicates are predicated of the whole group, distributive readings quantify over the atomic parts of that maximal group. An explicit maximality condition expressing this idea is displayed in DRS (179). The maximality condition is analogous to (162) above:

(179) the Dutch climbers (version 1)

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  []
  object(D,climber)
  property(D,'Dutch')
  [E]                          % maximality condition
  structure(E,dom)
    [F]
    structure(F,atomic)
    part_of(F,E)
    =>
    []
    object(F,climber)
    property(F,'Dutch')
  =>
  []
  part_of(E,B)
```

Concerning the maximality conditions we encounter the same problems as with representation (162) above. There I discussed the problem of how to restrict the domain of quantification appropriately and, furthermore, I addressed problems that occur in the practical implementation with respect to the efficiency of logical inferences. There is a further problem in that maximality may be too strong because the reference of definite descriptions does not necessarily correspond to the *maximal* set but possibly only to *relevant parts* of it. I therefore suggest the same solution as in (163) above and replace the maximality condition in (179) by the following DRS condition using the condition `maximal/3`:

(180) maximality condition for *the Dutch climbers*

```
maximal(B,context,drs([E],[structure(E,dom),
              drs([G],[structure(G,atomic),part_of(G,E)])=>
              drs([],[object(G,climber),property(G,'Dutch')])])
          => drs([],[part_of(E,B)]))
```

Both solutions, (179) and (180), prevent left upward monotonicity. Again the logical status of the added non-standard condition `maximal/3` needs further investigation but does its “job” in the current implementation.

Both representations allow to represent definite descriptions with restrictive relative clauses that e.g. have a distributive reading whereas the main clause has a collective reading, for example:

(181) The climbers who (each) summited together got an award.

Thus the current proposal does not concretely determine the extension of the reference of the definite description but tries to model its inferential behaviour via the added complex maximality condition.

Inferences

For (direct) anaphoric references of definite descriptions the inferences depend on the type and the representation of the antecedent. Thus for example the following inconsistency can be derived:

(182) Inconsistent
Five climbers summited. The climbers did not summit.

For discourse new definite descriptions the suggested representation prevents left upward monotonicity as in (183) while maintaining right upward monotonicity as in (184).

(183)	IF	The Dutch climbers summited.
	~~THEN~~	The climbers summited.
(184)	IF	The Dutch climbers saw some British climbers.
	THEN	The Dutch climbers saw some climbers.

Furthermore, inferences from definite descriptions to indefinite noun phrases are valid (not vice versa). The cardinality of the definite description can be specialized by adding additional numericals as in *the five climbers*. From these definite descriptions the cardinality noun phrase *five climbers* can be deduced.

Due to the “indirect” maximality condition `maximal/3` – that is not related to any additional logical axioms that calculate maximality – sentences of the following type are calculated as consistent although they may under certain conditions be felt as inconsistent.

(185) Consistent
The Dutch climbers summited. A Dutch climber did not summit.

Participation by all objects can be explicitly expressed by the following constructions that then lead to inconsistencies.

(186) Inconsistent
a. Every Dutch climber summited. A Dutch climber did not summit.
b. All Dutch climbers summited. A Dutch climber did not summit.

Problems occur when the antecedent of the inference contains both antecedent and anaphora whereas the consequent of the inference only contains the definite description as in:

(187)	IF	Five men summited. The men were Dutch.
	THEN (?)	The men were Dutch.

In my setting the occurrence of *the men* in the consequent will be treated as discourse new and no connection between the anaphoric use of *the men* in the antecedent can be established. Also inferences like

(188) IF Five men summited. The men were Dutch.
THEN (?) The five men who summited were Dutch.

cannot be dealt with.

Discussion, Problems, Limitations

Definite descriptions not only raise problems concerning their semantic representation but predominantly problems concerning the determination of their referent. Since I have not investigated the latter problem I refer to the investigations mentioned above (e.g. Vieira and Poesio 2000) and the references therein.

The current proposal represents direct anaphoric references of definite descriptions and discourse new occurrences. Concerning the representation of direct anaphoric references it is not clear how much material has to be repeated within the definite description. For example in the discourse

(189) Three Dutch climbers who reached Mount Everest got bad frostbite.
Seven climbers who reached Mount Everest had no injuries.
The (three) (Dutch) climbers (who reached Mount Everest) received medical treatment.

Adding more material to the anaphoric definite descriptions leads to an unequivocal determination of the antecedent. Leaving out the additional material in brackets *the climbers* will be resolved to *seven climbers who reached Mount Everest* simply due to recency. Note also, that I have not included phenomena of antecedent construction via summation (Kamp and Reyle 1993, section 4.1.1). In example (189) this would mean that *the climbers* can refer to the group consisting of the three Dutch climbers with frostbite and the seven climbers without injuries.

I have made two suggestions for the representation of discourse new definite descriptions, one using a universally quantified maximality condition, a second using a somewhat non-standard maximality condition `maximal/3` that resembles – in a way – Link's usage of the ι-operator. Only that in my setting the maximality condition is not related to any further logical axioms. The maximality condition simply prevents – in a practically efficient way – left upward monotonicity of definite descriptions and guarantees that discourse new occurrences of definite descriptions with the same head noun and the modifications refers to the same object. This is for example relevant when the definite descriptions occur as discourse new in both antecedent and consequent of a proof. I am aware that this treatment of definite description will not be the "final word" but I evaluated it to be sufficient for the current purpose, in particular with respect to efficiency within automatic theorem proving tasks.

4.5.2.9 Individual Denoting Noun Phrase: 'all'

Representation

In my setting *all* introduces an individual denoting noun phrase with a preference for a distributive reading but which, under certain circumstances, also allows for a collective reading. The semantics of *all* is similar to discourse new *the* only that participation of all elements is required. Thus, adding the maximality condition `maximal/3` without additional axioms would not be sufficient but the explicit universally quantified maximality condition has to be used (with negative consequences for the efficiency in the practical application). Furthermore, I chose to add a non-standard element `property(A,all)` for the representation of *all* to be able to reconstruct inferences between *all* and the vague quantifiers *most*, *many*, *few*. Thus I suggest the following representation:

(190) all Dutch men

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
property(A,all)
  [C]
  structure(C,atomic)
  part_of(C,B)
  =>
  []
  object(C,man)
  property(C,'Dutch')
  [F]
  structure(F,dom)
    [H]
    structure(H,atomic)
    part_of(H,F)
    =>
    []
    object(H,man)
    property(H,'Dutch')
  =>
  []
  part_of(F,B)
```

If we want to treat *all* and *all of* with discourse new definite descriptions (e.g. *all of the men*) equivalently we additionally have to add a maximality condition `maximal/3` along the lines of (180) above. Due to the existential import of *all* we are also able to treat collective readings.

Inferences

Representations for *all* are right monotone increasing but not left monotone increasing. Again inferencing for the distributive readings is very inefficient. Due to the explicit maximality condition the following inconsistencies (for the distributive reading of *all*) can be derived.

(191) Inconsistent

a. All men summited. There is a man who did not summit.

b. All men summited. Not every man summited.

The inconsistencies in (191) require the domain axiom (Ax. 2), axiom (Ax. 11) according to which atoms do not have proper parts, and the identity axiom (Ax. 15)-1. It is evident that with the same axioms we can deduce the negation of the second sentences of (191).

(192) IF All men summited.
THEN Every man summited.

Note, that the reverse inference is not valid due to the existential import of *all*.

There is a complex interrelation between the various absolute and relative quantifiers. Here I will only show an exemplary logical relation between *all* and *most*. To deduce

(193) IF All men wait.
THEN Most men wait.

we need – apart from axiom (Ax. 1) an additional axiom that states the logical relation between *all* and *most*. This axiom will be repeated below as (Ax. 28).

(194) $\forall$N(property(N,all) $\rightarrow$ relation(N,most,N))

Note also, that this inference requires – apart from the explicitly universally quantified maximality condition – the addition of the condition `maximal/3` along the lines of (180) above.

We will see in section 4.5.2.10 below a number of other relations between quantifiers.

Discussion, Problems, Limitations

In my setting noun phrases with *all* have an existential import and thus can have collective and distributive readings. Maximality conditions have to be added to prevent left upward monotonicity and to explain that the object denoted buy *all N* is maximal with respect to the property expressed by the restriction *N*. Additional logical axioms can neatly be added to simulate logical relations between a number of quantifiers like *most*, *many*, *more than half of*, *few*, *few of*, etc. The explicit maximality conditions lead to efficiency problems in the practical application, in particular when distributive readings are involved.

4.5.2.10 Partitives

Representation

Partitive constructions consist of a determiner followed by a prepositional *of*-phrase whose noun phrase is usually definite.

(195) Five of the climbers who summited got frostbite.

The embedded definite noun phrase provides a base set which restricts the domain of quantification for the determiner. As discussed above in 4.5.2.8 this base set can be given anaphorically (directly or indirectly) or it can be discourse new. The determiner itself can be universally quantifying like *each*, it can be an absolute cardinality quantifier like *two*, *at most two*, or an indefinite quantifier like *some*, *several*, or it can be a proportional quantifier like *few*, *many*, *most*, *more than half of*. I chose to represent the definite noun phrase analogously to the representations in section 4.5.2.8 above. Depending on the quantifier the partitive relation between a subgroup and the base set is expressed via an additional condition `part_of/2`. There are different representations depending on the classification of the determiner.

Partitives with Quantifying Determiners: 'each of', 'both of'. The quantifier *each* simply triggers universal quantification over the given or the new base set. If the base set is discourse new we get the following representation.

(196) Each of the 28 climbers summited.

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,eq,28)
  [C]
  structure(C,atomic)
  part_of(C,B)
  =>
  []
  object(C,climber)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  [E]
  structure(E,event)
  predicate(E,summit,D)
maximal(B,context,drs([F],[structure(F,dom),drs([H],
[structure(H,atomic), part_of(H,F)])=>drs([],[object(H,climber),
object_type(H,person)])])) => drs([],[part_of(F,B)]))
```

Note that I will represent the determiner *both* and *both of the* equivalent to *each of the two* yielding the following representation (with abbreviated maximality condition).

(197) Both men wait. = Both of the men wait. = Each of the two men wait.

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,eq,2)
  [C]
  structure(C,atomic)
  part_of(C,B)
  =>
  []
  object(C,man)
maximal(B,context,'the men')
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
```

```
  [E]
  structure(E,state)
  predicate(E,wait,D)
```

Both always triggers a distributive reading.

Partitives with Right-Increasing Absolute Determiner: 'all of'. The representation of

(198) All of the climbers summited.

is analogous to the representation of *all* as described in section 4.5.2.9 above. Only that a discourse-new definite description introduces a maximality condition `maximal/3`.

Partitives with Absolute Increasing Determiners: 'at least five of the', 'some of the'. To establish a relation between the base set and its part an explicit `part_of/2` relation is introduced into the DRS. Again, I present the representation for discourse new definite descriptions (abbreviating the maximality condition).

(199) at least five of the climbers

```
[A,B,C,D]
structure(B,group)                      % the climbers
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
  [F]
  structure(F,atomic)
  part_of(F,B)
  =>
  []
  object(F,climber)
maximal(B,context,'the climbers')       % maximality cond. (abbr.)
structure(D,group)                      % partitive
quantity(D,cardinality,C,count_unit)
value(C,geq,5)
part_of(D,B)
```

Apart from the base set `B` representation (199) introduces another group `D` of cardinality greater or equal than 5 that is part of the base set referred to by `B`. As usual distributive readings are represented by universal quantification over the discourse referent `D`, in collective readings the predicate is directly applied to `D`.

Partitives with Absolute Non-Increasing Determiners: 'at most two of', 'exactly 27 of'. As with non-partitive counterparts non-increasing quantifiers require additional maximality conditions. In partitive constructions this maximality relates only to the extension of the base set leading to the following representation for collective readings.

(200) At most two of the men gathered. (collective reading)

```
[A,B,C,D,E]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
  [F]
  structure(F,atomic)
  part_of(F,B)
```

```
    =>
    []
    object(F,man)
  maximal(B,context,'the men')
  structure(D,group)
  quantity(D,cardinality,C,count_unit)
  value(C,leq,2)
  part_of(D,B)
  structure(E,event)
  predicate(E,gather,D)
    [O,P]                         % explicit maximality condition
    structure(O,group)
    part_of(O,B)
    structure(P,event)
    predicate(P,gather,O)
    =>
    []
    part_of(O,D)
```

The explicit maximality condition expresses that any group that is part of the base set `B` and that gathers is part of `D`. For distributive readings the maximality condition is shown in (201).

(201) At most two of the men summited. (distributive reading)

```
  [A,B,C,D]
  structure(B,group)
  quantity(B,cardinality,A,count_unit)
  value(A,geq,2)
    [E]
    structure(E,atomic)
    part_of(E,B)
    =>
    []
    object(E,man)
  maximal(B,context,'the men')
  structure(D,group)
  quantity(D,cardinality,C,count_unit)
  value(C,leq,2)
  part_of(D,B)
    [N]
    structure(N,atomic)
    part_of(N,D)
    =>
    [O]
    structure(O,event)
    predicate(O,summit,N)
    [P]
    structure(P,dom)
    part_of(P,B)
      [Q]
      structure(Q,atomic)
      part_of(Q,P)
      =>
      [R]
      structure(R,event)
      predicate(R,summit,Q)
    =>
```

```
  []
  part_of(P,D)
```

The maximality condition expresses that any object `P` that is part of the base set `B` such that each atomic part of `P` summits is also part of the "partitive object" `D`.

Partitives with Context-Dependent Proportional Right-Increasing Determiners: 'many of', 'most of', 'more than half of'. Above I discussed that vague determiners like *many* and *few* are ambiguous between a cardinal and a proportional reading. In my fragment these determiners get a proportional reading if used in partitive constructions where the base set is made explicit via the definite description. With '*many*' I denote the cardinal reading of *many*, whereas '*many of*' expresses the proportional interpretation of *many*. To express the proportional reading in the DRS the condition `relation(N,many_of,M)` is introduced. Thus the proportional reading of the right increasing quantifier *many* of is similar to the representation of *most*/*most of*. Following is an abbreviated representation for *many of the men*:

(202) many of the men

```
[A,B,C,D]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
  [F]
  structure(F,atomic)
  part_of(F,B)
  =>
  []
  object(F,man)
maximal(B,context,'the men')
structure(D,group)
quantity(D,cardinality,C,count_unit)
value(C,geq,2)
part_of(D,B)
relation(C,many_of,A)
```

Due to the maximality condition `maximal/3` left upward monotonicity is prevented. The proportional readings most of and more than half of are represented using the DRS conditions `relation(C,most_of,A)` and `relation(C,more_than_half_of,A)`, resp.

Partitives with Context-Dependent Proportional Non-Increasing Determiners: 'few of'. The treatment of *few of* is analogous to *many of* only that maximality conditions to prevent upward monotonicity have to be added. The collective reading will be represented as:

(203) Few of the men gathered.

```
[A,B,C,D,E]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
  [F]
  structure(F,atomic)
  part_of(F,B)
  =>
  []
```

```
  object(F,man)
maximal(B,context,'the men')
structure(D,group)
quantity(D,cardinality,C,count_unit)
value(C,geq,2)
part_of(D,B)
relation(C,few_of,A)
structure(E,event)
predicate(E,gather,D)
  [O,P,Q]                          % maximality condition
  structure(P,group)
  quantity(P,cardinality,O,count_unit)
  value(O,geq,2)
  part_of(P,B)
  structure(Q,event)
  predicate(Q,gather,P)
  =>
  []
  relation(O,leq,C)
```

The maximality condition of the distributive reading is displayed in (204).

(204) maximality condition for distributive reading of (203)

```
[P]
structure(P,dom)
part_of(P,B)
  [Q]
  structure(Q,atomic)
  part_of(Q,P)
  =>
  [R]
  structure(R,event)
  predicate(R,summit,Q)
=>
[]
part_of(P,D)
```

As with the non-partitive equivalent in section 4.5.2.6 this representation leads to efficiency problems in the practical application.

Inferences

Non-context dependent determiners occurring in partitive constructions trigger analogous inferences as their non-partitive counterparts. I will therefore not go into details here. Note, that the partitive use implies the non-partitive use but not vice versa.

(205) IF At least seven of the men wait.
THEN At least seven men wait.

The inference only requires the transitivity of the part-of relation expressed in axiom (Ax. 7).

Logical relations between context-dependent determiners require a number of additional axioms. There are three classes of additional axioms. First, auxiliary axioms that express the relation between *most* and *most of*. Second, axioms treating logical relations between proportional readings of determiners and finally, axioms to explain absolute cardinality readings of *few* and *many*.

1. Auxiliary axioms

(Ax. 26) $\forall N \forall M(\text{relation}(N,\text{most},M) \rightarrow \text{relation}(N,\text{most\_of},M))$
(Ax. 27) $\forall N \forall M(\text{relation}(N,\text{most\_of},M) \rightarrow \text{relation}(N,\text{most},M))$

2. Proportional Determiners

(Ax. 28) $\forall N \forall M(\text{property}(N,\text{all}) \rightarrow \text{relation}(N,\text{most},N))$
(Ax. 29) $\forall N \forall M(\text{relation}(N,\text{most},M) \rightarrow \text{relation}(N,\text{more\_than\_half\_of},N))$
(Ax. 30) $\forall N \forall M(\text{relation}(N,\text{most},M) \rightarrow \text{relation}(N,\text{many\_of},N))$
(Ax. 31) $\forall N \forall M \forall Z(\neg(\text{relation}(N,\text{few\_of},Z) \wedge \text{relation}(M,\text{leq},N) \wedge \text{relation}(M,\text{many\_of},Z)))$
(Ax. 32) $\forall W \forall Q_1 \forall P \forall Q_2 \forall Z(\neg(\text{quantity}(W,\text{cardinality},Q_1,\text{count\_unit}) \wedge$
$\text{quantity}(P,\text{cardinality},Q_2,\text{count\_unit}) \wedge \text{part\_of}(P,W) \wedge$
$\text{property}(Q_1,\text{few\_of},Z) \wedge \text{property}(Q_2,\text{many\_of},Z)))$

3. Absolute Determiners

(Ax. 33) $\forall N \forall M(\neg(\text{property}(N,\text{few}) \wedge \text{relation}(M,\text{leq},N) \wedge \text{property}(M,\text{many})))$
(Ax. 34) $\forall W \forall Q_1 \forall P \forall Q_2(\neg(\text{quantity}(W,\text{cardinality},Q_1,\text{count\_unit}) \wedge$
$\text{quantity}(P,\text{cardinality},Q_2,\text{count\_unit}) \wedge \text{part\_of}(P,W) \wedge$
$\text{property}(Q_1,\text{few}) \wedge \text{property}(Q_2,\text{many})))$

To give a few examples the axioms allow to make deductions listed in (206) and to derive inconsistencies listed in (207).

(206) a. IF Many men summited.
THEN Some men summited.
b. IF All men wait.
THEN Many of the men wait. / Some men wait.
c. IF Few men wait.
THEN Some men wait.
d. IF Few men wait.
THEN Not many men wait.
e. IF Few of the men wait.
THEN Some men wait. / Some of the men wait.
f. IF Few of the men wait.
THEN Not many of the men wait.

Furthermore, the following inconsistencies can be derived.

(207) Inconsistent
a. Few men summited. Many men summited.
b. Few of the men summited. Many of the men summited.

c. Few of the men summited. Most of the men summited.
d. Few of the men summited. All (of the) men summited.

Though intuitively valid, I am currently not able to reconstruct the following inconsistencies:

(208) Inconsistent (but not yet derivable)
a. Many of the men wait. Many of the men do not wait.
b. Few of the men wait. Few of the men do not wait.

Furthermore, it may be counterintuitive that the distributive reading of *all of the men* implies *each of the men* but not vice versa.

Discussion, Problems, Limitations

In the previous sections I have given a representation for partitive constructions with different types of determiners. Without giving an "exact" semantics of *most*, *many*, *few* in terms of assigning percentages or absolute numbers I have shown how logical relations between the determiners can be reconstructed using a number of first-order axioms. One of the problems is again how to determine the reference of the definite description correctly. Furthermore, I have given determiners like *many* and *few* that occur in partitive constructions a proportional interpretation whereas *many* and *few* occurring in non-partitive constructions receive an absolute cardinality reading. This treatment is a simplification in that non-partitive constructions with *many* and *few* can also have proportional readings. Moreover, the maximality conditions for the distributive readings of non-monotone increasing quantifiers cause efficiency problems in the practical implementation.

4.5.2.11 Quantificational Noun Phrases: 'every', 'each', 'everything', 'no', 'nothing', 'no one', 'not every', …

Quantificational noun phrases do not introduce discourse referents into the domain of discourse but are always used quantificationally. The prototypical quantificational determiners are every and each which correspond to the classical universal quantifiers. The corresponding DRS is:

(209) Every man waits.

```
[]
  [A,B]
  structure(B,atomic)
  quantity(B,cardinality,A,count_unit)
  value(A,eq,1)
  object(B,man)
  =>
  [C]
  structure(C,state)
  predicate(C,wait,B)
```

The quantifier *everyone* that does not have an explicit restrictor is represented as

(210) Everyone sleeps.

```
[]
  [A]
  structure(A,dom)
  =>
  [B]
  structure(B,state)
  predicate(B,sleep,A)
```

Negated versions of universal quantifiers like *not every* are represented as:

(211) Not every man sleeps.

```
[]
NOT
  []
    [A,B]
    structure(B,atomic)
    quantity(B,cardinality,A,count_unit)
    value(A,eq,1)
    object(B,man)
    =>
    [C]
    structure(C,state)
    predicate(C,sleep,B)
```

I will also treat singular and plural *no* as quantificational determiner. The representation of the singular *no man* is straightforward:

(212) No man waits.

```
[]
  [A,B]
  structure(B,atomic)
  quantity(B,cardinality,A,count_unit)
  value(A,eq,1)
  object(B,man)
  =>
  []
  NOT
    [C]
    structure(C,state)
    predicate(C,wait,B)
```

Representation (212) is equivalent to the representation for

(213) It is not the case that there is a man who waits.

The interpretation of the plural noun phrase *no men* is less clear due to possible collective/distributive ambiguities. Does the sentence

(214) There are no boys who ate a pizza.

mean that there is no group of boys who ate a pizza together, or that there is no single boy who ate a pizza, or that both there is no single boy *and* no group of boys that ate a pizza (alone or together). I chose to give plural *no* a strong interpretation in the latter sense, i.e. it denies both distributive and collective readings. This is realized by using a neutral discourse referent.

(215) No men wait.

```
[]
  [A,B]
  structure(B,dom)
  quantity(B,cardinality,A,count_unit)
  value(A,geq,1)
    [C]
    structure(C,atomic)
    part_of(C,B)
    =>
    []
    object(C,man)
  =>
  []
  NOT
    [D]
    structure(D,state)
    predicate(D,wait,B)
```

The noun phrase *no one* also uses a neutral discourse referent:

(216) No one sleeps.

```
[]
NOT
  [A,B]
  structure(A,dom)
  structure(B,state)
  predicate(B,sleep,A)
```

Scope ambiguities can occur when quantifying noun phrases co-occur with other noun phrases. These scope ambiguities will be dealt with in chapter 5.

Inferences

The universal quantifiers *every/each* are left-decreasing and right-increasing without additional axioms. The quantifiers *everyone/everything* are right-increasing. The quantifier *not every* is left-increasing and right-decreasing.

The quantifier *no* is left- and right-decreasing. *Nothing/no one* are right-decreasing.

With the given representations the standard logic relations between universal and existential quantifiers and negation can be reconstructed, for example

(217) IF No boy buys an apple.
THEN Every boy buys no apple. / Every boy does not buy an apple.

Also classical inference relations like *modus pones* or *modus tollens* can be reconstructed.

(218) IF Every boy sleeps. John is a boy.
THEN John sleeps.

(219) IF Every boy sleeps. No one sleeps.
THEN There is no boy.

The modus tollens in (219) additionally requires the domain axiom (Ax. 2).

The logical relation between singular and plural *no* and other determiners require additional axioms. For example we can deduce (220) but the reverse direction is not valid.

(220) IF No men disappear.
THEN No man disappears.

The deduction in (220) requires the additional domain axiom (Ax. 2), the axiom that atoms have no proper part (Ax. 11), furthermore the equality substitution axiom (Ax. 15)-1 and the number axiom (Ax. 22)-1. With the same axioms the following sentences can be proven to be inconsistent.

(221) Inconsistent

a. John sees a man. John sees no men.
b. John is a man. No men wait. John waits.

Furthermore, the following inconsistencies can be proven using additional axioms.

(222) Inconsistent

a. Some men lift a table. No men lift a table.
b. Some men each lift a table. No men lift a table.
c. Some men each lift a table. No man lifts a table.

(222)a requires the domain axiom (Ax. 1) and (222)b requires an interaction of the domain axiom (Ax. 2), axioms (Ax. 9) and (Ax. 11) concerned with the proper part of relation, identity axiom (Ax. 15)-1, the number axiom (Ax. 22)-1 and (Ax. 10) that states that groups consist of atomic parts. The inconsistency (222)c requires axioms (Ax. 9) and (Ax. 10). Note however, that the following sentences come out as consistent.

(223) Consistent
Some men (together) lift a table. No man lifts a table.

since the sentence *no man lifts a table* does not exclude that there are groups of men that lift a table.

Discussion, Problems, Limitations

The main problem when different types of quantifiers co-occur is to determine the relevant scope relations. Here we are only concerned with the logic representation. Furthermore, the behaviour of event variables within negation is a complex field of research that cannot be dealt with here. Here event variables always get narrow scope with respect to the negation. A further observation is that I treat the distributive reading of *all* different from *every*. This might be confusing in certain situations. However, since there are examples where distributive *all* does have readings that are different from *every* I suggest to use the singular *every* whenever simple universal quantification is expressed and no existential import is intended.

4.5.2.12 Measurement Phrases

Representation

Measurement phrases are of the form "numeral + measure word + of + mass noun or plural count noun". Examples are *two ounces of gold*, *three pounds of apples*, *a cup of coffee* etc. I have said in section 4.5.1.1 that noun phrases are associated with quantity information:

(224) `quantity(X,Dimension,N,Count_Unit)`

Apart from the cardinality there are other possibilities to measure the quantity of an object, e.g. via explicit measurement expressions. The representation of measurement phrases with plural count nouns replaces cardinalities as measurements with these explicit measurements. The measurement unit and the dimension of measurement are lexically specified by the measure word (e.g. *ounce*, *pound*, *litre*). We then get the following representation for measurement phrases with plural count nouns:

(225) two pounds of apples

```
[A,B]
structure(B,group)
quantity(B,weight,A,pound)
value(A,eq,2)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  []
  object(D,apple)
```

Measurement phrases with plural count nouns always get a collective reading since something about the quantity of an object is predicated.

To describe the semantics of measurement phrases with mass nouns I first show the representation for mass noun phrases without explicit measurement information. These noun phrases are not associated with explicit quantity information in the DRS. The noun phrase *water* in *John drinks water* is simply represented as:

(226) (some) water

```
[A]
structure(A,mass)
object(A,water)
```

Quantity information is added via explicit measurement phrases as in

(227) two liters of water

```
[A,B]
structure(B,mass)
quantity(B,volume,A,liter)
value(A,eq,2)
```

The measure word itself is countable. Possible measurement dimensions are for example *size*, *distance*, *area*, *volume*, *weight*, *speed*, *temperature*, etc.

Measurements of objects can also be expressed by other constructions, e.g. using measure verbs (the semantics of which resembles copular verbs):

(228) a. John weighs 80 kg.
 b. The books cost 10 dollars.

or constructions like

(229) The weight of John is 80 kg.
(230) All data queue entries are 80 bytes long.
(231) Each local record is 48 characters long.
(232) The connector box is 0.56 meter (22 inches) long.

Measure verbs like *weigh*, *cost* have to be explicitly defined in the lexicon and associated with a dimension. (228) can then be represented as

(233) John weighs 80 kg

```
[A,B,C,D]
named(B,'John')
structure(B,atomic)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
quantity(B,weight,C,kg)
value(C,eq,80)
structure(D,state)
predicate(D,be,B)
```

The example shows that dimensions of measurement for an object can be mixed. Under one perspective the cardinality is relevant under a different perspective the weight is considered. The representation of (229) will be equivalent to (233).

Inferences

For explicit measurement constructions analogous inferences hold as for simple cardinality noun phrases. In particular the inferences concerned with numerals like *at least n*, *exactly n*, *at most n* apply accordingly.

Note however, that the quantities of two objects can only be compared with respect to a certain measurement dimension and unit of measurement. Therefore we need a generalization of axiom (Ax. 21) introduced in section 4.5.2.5 that expresses that parts have a smaller quantity than the whole. In the following axiom the dimension and the count unit are also universally quantified:

(Ax. 35) 1. ∀W∀Q1∀N∀P∀Q2∀C∀U(quantity(W,C,Q1,U) ∧
 value(Q1,leq,N) ∧ part_of(P,W) ∧ quantity(P,C,Q2,U)
 → value(Q2,leq,N))
 2. ∀W∀Q1∀N∀P∀Q2∀C∀U(quantity(W,C,Q1,U) ∧
 value(Q1,less,N) ∧ part_of(P,W) ∧ quantity(P,C,Q2,U)
 → value(Q2,less,N))

To derive the following inference we need axioms that describe the mass domain.

(234) IF There are three ounces of gold.
THEN There are two ounces of gold.

The inference (234) requires axioms (Ax. 16), (Ax. 17) that were introduced in section 4.4.4, pp. 108.

The given axioms allow to derive both of the following inconsistencies, (235)a from the count domain, (235)b from the mass domain.

(235) Inconsistent
a. John drinks at most 2 liters of water. John drinks 3 liters of water.
b. John buys exactly 3 pounds of apples. John buys 4 pounds of apples.

Both inconsistencies in (235) require the number axioms (Ax. 22)-1 and (Ax. 22)-2 plus the generalized axiom (Ax. 35).

Furthermore, the inferences of the following type are valid:

(236) IF John drinks two liters of water.
THEN John drinks water.
(237) IF John drinks two liters of water.
THEN John drinks something.

Inference (237) requires the additional domain axiom (Ax. 3).

Discussion, Problems, Limitations

There are many more possibilities to express measurements in natural language. The suggested examples were intended to show the basic ideas of the semantic representation. An interesting discussion about representing quantities can be found in Dale (1992). The problem in practical applications is of course to automatically recognize measurement constructions and distinguish them from noun phrases with "ordinary" *of*-PP modifiers. This is particular difficult for less precise units of measurement like *cup*, *bottle* etc. I have addressed this problem in section 2.3.6, pp. 22 where I showed that e.g. the noun phrase *two bottles of wine* can occur in both constructions. I will not touch problems concerned with the aspectual change that can be observed when verbs combine with objects that express a specified quantity in contrast to objects that express an unspecified quantity:

(238) a. John drinks two liters of water.
b. John drinks water.

These problems are for example discussed in articles by Verkuyl and Krifka.

4.5.2.13 Proper Nouns

Representation

There are proper nouns like *John*, *Mary*, *the Smiths* that can – in different contexts – refer to different individuals. In a given context, however, they usually uniquely pick out a single individual or a unique group of individuals. Other proper nouns like *Mount Everest*, *the Alps* etc. usually refer to the same object independent of the context. Since in a particular context proper nouns have unique reference different occurrences of the same proper noun will relate to the same discourse referent in the DRS representation. Furthermore, proper nouns always get topmost scope. For the current application proper nouns have to be predefined in the lexicon.

Singular proper nouns like *John* are represented as

(239) DRS representation of *John*

```
[A,B]
named(B,'John')
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
```

Plural proper nouns like *the Smiths* or *the Alps* introduce group objects the internal structure of which is not further specified.

(240) DRS representation of *the Alps*

```
[A,B]
named(B,'the Alps')
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
```

It is not quite clear whether plural proper nouns should receive a distributive reading. In

(241) The Smiths are Dutch.

the property of being Dutch is predicated of each element of the Smiths, however, the distinction between plural proper nouns and plural definite descriptions is not quite clear in this example. Currently, I do allow distributive readings for plural proper nouns.

Inferences

Since any occurrence of the same proper noun relates to the same discourse referent the following set of sentences can be proven to be inconsistent without further axioms.

(242) Inconsistent
Mount Everest is a mountain. John climbs Mount Everest. John does not climb a mountain.

Furthermore, since proper nouns get topmost scope, the following inference is valid.

(243) IF Every climber sees Mount Everest. Mount Everest is a mountain.

THEN Every climber sees a mountain.

Moreover, the inference from proper nouns to *someone/something* is valid using the domain axioms (Ax. 1) and (Ax. 2).

Discussion, Problems, Limitations

In the current application proper nouns have to be predefined in the lexicon. For large applications the interesting issue is, of course, to recognize proper nouns automatically. Furthermore, as noted above, the distinction between plural proper nouns and plural definite descriptions is not always clear.

4.5.2.14 Noun Phrase Conjunction

Representation

In the literature on plural semantics NP coordination examples like the following (Landman 1989)

(244) The boys and the girls had to sleep in different dorms, met in the morning at breakfast, and were then wearing their blue uniforms.

have lead to a dispute whether it is necessary to introduce multi-level plural objects into the domain of discourse. The reason is that in (244) *the boys and the girls* has at the same time to stand for an unstructured sum to fit the predicate *meet*, to distributive one level down to fit the predicate *sleep in different dorms* and to distribute two levels down to fit the predicate *wearing*. The representation of these examples is a notorious problem since the summing operation deletes structure. Once the sum is formed there is no simple way to reconstruct intermediate parts that were used to build the structure. A number of solutions have been proposed to solve the problem that I will not present in detail here. A good summary can be found in Link (1998a, pp. 27). The proposals range from assuming just one intermediate level (Link 1984), to Landman's (1989) proposal to erect "the full cumulative hierarchy of order omega over any given domain of discourse" (Link 1998a, p. 29). Other proponents of the latter direction are Hoeksema (1983). More recently, Landman's proliferation of entities has been seen as a clear problem of over-representation and arguments have been provided to explain the data without introduction of new types of objects (Schwarzschild 1990, 1991, Krifka 1991b). These arguments suggest to integrate phenomena like intonation, syntactic structure, pragmatic knowledge etc. to explain the data. Details can be found in the respective articles or are summarized in Link (1998a, pp. 27).

In my approach a conjunction of two individual denoting noun phrases will not introduce new types of objects. However, the DRS for the conjunction will reflect in which way the complex noun phrase denotes a group consisting of parts. For example the representation for *a boy and a girl* (245) introduces a discourse referent `D` for *a boy* and a discourse referent `F` for *a girl*, furthermore a group discourse referent `B` representing the sum of `D` and `F` (`sum_of(B,[D,F])`) is introduced and the conditions that `D` and `F` are proper parts of `B` are added.

(245) a boy and a girl

```
[A,B,C,D,E,F]
structure(B,group)                     % group object
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
sum_of(B,[D,F])                        % coordination
structure(D,atomic)                    % a boy
quantity(D,cardinality,C,count_unit)
value(C,eq,1)
object(D,boy)
proper_part_of(D,B)
structure(F,atomic)                    % a girl
quantity(F,cardinality,E,count_unit)
value(E,eq,1)
object(F,girl)
proper_part_of(F,B)
```

In section 4.4 I stated that `sum_of(B,[D,F])` is a practically suitable form to represent that `B` is the supremum of the objects `D` and `F`. To model that `B` is the least upper bound I will take recourse to list-manipulations operations like flattening or permutation. Also commutativity, associativity and idempotence of conjunction are not directly enforced via first-order axioms but more efficiently simulated by list processing operations. The condition `sum_of/2` can only be introduced by NP conjunction.

Coordination of two individual denoting plural noun phrases is treated analogously.

(246) two men and three women

```
[A,B,C,D,E,F]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
sum_of(B,[D,F])                        % coordination
structure(D,group)                     % two men
quantity(D,cardinality,C,count_unit)
value(C,eq,2)
  [H]
  structure(H,atomic)
  part_of(H,D)
  =>
  []
  object(H,man)
proper_part_of(D,B)
structure(F,group)                     % three women
quantity(F,cardinality,E,count_unit)
value(E,eq,3)
  [I]
  structure(I,atomic)
  part_of(I,F)
  =>
  []
  object(I,woman)
proper_part_of(F,B)
```

A collective reading will predicate the property of the whole group `B`, a fully distributive reading will distribute over the atomic parts of `B` and an "intermediate" reading will predicate the property of the explicitly mentioned group parts of `B`, viz. `D` and `F`. In my fragment I have no operator to trigger this intermediate reading, only fully collective and fully distributive readings can be explicitly triggered. Intermediate readings have to be rephrased.

Inferences

To prevent inferences of the following type

(247) IF John and Bill and Mary lifted a table.
~~THEN~~ John and Bill lifted a table.

the additional condition `sum_of/2` is essential. Without this condition the inference would wrongly come out as true. To model commutativity of conjunction in the practical implementation it turned out to be most efficient to take recourse to list manipulation operations of the respective programming language. It is also possible to model these operations via first-order axioms but I found no efficient solution to do so. How this is practically implemented will be dealt with in chapter 6. We will see there that the following inferences are valid as a result of the list manipulation operations.

(248) IF Two men and three women together lifted a table.
THEN Three women and two men together lifted a table.

Also the following deductions are valid

(249) IF Two men and three women each lift a table.
THEN A woman lifts a table.

Inference (249) requires the axioms (Ax. 7), (Ax. 9), (Ax. 22)-1, (Ax. 10).

Though intuitively valid the following inference are currently practically not deducible.

(250) Valid inferences (but currently not deducible)

a. IF John and Bill and Mary each lift a table.
THEN John and Bill each lift a table.

b. IF Two men and three women each lift a table.
THEN Three women each lift a table.

b. IF Two men and three women each lift a table.
THEN A man and a woman each lift a table.

Discussion, Problems, Limitations

The above examples only scratched the surface of the problems raised by NP conjunction. I have only looked at the coordination of right-increasing individual denoting noun phrases. There are interesting investigation on which types of noun phrases can be felicitously coordinated at all (e.g. *a man and two women* vs. ?*two men and every woman*). Link (1984, reprinted in 1998b) addresses an interesting problem that he calls "hydras". Hydras occur in examples like

(251) the boys and the girls who met yesterday

where you cannot distributive the relative clause over the conjuncts which means the relative clause has multiple heads (hence the name "hydras"). The problem is not so much how to represent hydras but how to determine whether the relative pronoun relates to the immediately preceding head noun or to some constructed group object.

Furthermore, I have not looked at non-phrasal coordination like

(252) the boys and girls

Logically, the treatment of coordination via the additional condition `sum_of/2` and recourse to list processing operations seems disappointing. This does, however, not mean that these list processing operations cannot, in principle, be expressed in first-order logic. I will again address this option in chapter 6, in particular in section 6.3.2

4.5.3 Negation

Negation occurs in several forms. One form is via implicitly negative quantifiers like *no* or *no one*, or via negated quantifiers like *not all*, *not many*, *not every*. Another form of negation occurs via sentential negation as in (253)a and (253)b or via verb phrase negation as in (253)c.

(253) a. It is not the case that the five climbers carried an oxygen bottle.
b. It is not the case that the five climbers found an empty tent.
c. 13 climbers did not carry an oxygen bottle.

Negation is a classical scope-bearing element and therefore can lead to scope ambiguities with quantifiers. Additionally, collective/distributive ambiguities can occur as for example in (253)b. Since I'm currently not investigating scope ambiguities of negation I will interpret negation "in situ", i.e. assume no scope alternations. Sentential negation as in (253)a will simply negate the whole DRS of the sentences without negation. Verb phrase negation gets a lower scope. For example, the collective reading of (253)c will be represented as

(254) collective reading of (253)c

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,eq,13)
  [C]
  structure(C,atomic)
  part_of(C,B)
  =>
  []
  object(C,climber)
NOT
  [D,E,F]
  structure(E,atomic)
  quantity(E,cardinality,D,count_unit)
```

```
value(D,eq,1)
object(E,oxygen_bottle)
structure(F,event)
predicate(F,carry,B,E)
```

The sentence means that there are 13 climbers which (together) do not have a certain property.

Concerning inferences we can rely on standard first-order inference rules plus a number of domain axioms (if plurals are involved). Simple examples are:

(255) a. IF A climber did not summit.
THEN Not every climber summited.
b. IF John carried two oxygen bottles.
THEN John did not carry no oxygen bottles.

The influence of negation to the determination preferred readings will be investigated in chapter 5. The determination of the correct scope of the negation is again a harder problem than its representation. This problem has not been investigated in this thesis.

4.5.4 Noun Phrase Modification

4.5.4.1 Adjectives

Adjectives can be used to modify nouns, that is to give more information about objects denoted by noun phrases. Most adjectives are intrinsically distributive, that is they can only be true of individuals, e.g. *young*, *happy*, *red*. There are, however, a number of "measure" adjectives that can trigger collective/distributive ambiguities. Examples are adjectives like *heavy*, *light*, *long* etc. as in

(256) John lifted some heavy tables.

Sentence (256) prefers a distributive reading of *heavy* in which case each of the tables is heavy. But the sentence can also mean that the group of tables as a whole is heavy. The two readings are represented as

(257) some heavy tables (distributive)

```
[B,C]
structure(C,group)
quantity(C,cardinality,B,count_unit)
value(B,geq,2)
  [E]
  structure(E,atomic)
  part_of(E,C)
  =>
  []
  object(E,table)
  property(E,heavy)           % distributive reading of 'heavy'
```

(258) some heavy tables (collective)

```
[B,C,D]
structure(C,group)
```

```
quantity(C,cardinality,B,count_unit)
value(B,geq,2)
property(C,heavy)              % collective reading of 'heavy'
  [E]
  structure(E,atomic)
  part_of(E,C)
  =>
  []
  object(E,table)
```

If the ambiguous adjectives occur in predicative constructions with the copula *be* collective/distributive ambiguities can also occur. See section 4.5.6 below.

4.5.4.2 Relative Clauses

Restrictive relative clauses can modify noun phrases that are interpreted collectively with respect to the main clause but distributively with respect to the relative clause (or vice versa).

(259) Five men who (each) summited (together) presented a slide-show.

Due to my global strategy according to which noun phrases are not ambiguous between a collective and a distributive reading these sentences can be straightforwardly represented.

(260) Representation of (259)

```
[A,B,C,D,E]
structure(B,group)                     % five men
quantity(B,cardinality,A,count_unit)
value(A,eq,5)
  [F]
  structure(F,atomic)
  part_of(F,B)
  =>
  []
  object(F,man)
  [G]                                  % who each summited
  structure(G,atomic)
  part_of(G,B)
  =>
  [H]
  structure(H,event)
  predicate(H,summit,G)
structure(D,atomic)                    % a slide-show
quantity(D,cardinality,C,count_unit)
value(C,eq,1)
object(D,slide_show)
structure(E,event)                     % together presented
predicate(E,present,B,D)
```

I have addressed the problem of "hydras" (relative clauses with multiple heads) in section 4.5.2.14 above.

4.5.4.3 Prepositional Phrases

Plural noun phrases that occur within prepositional phrases that modify noun phrases can also show collective/distributive ambiguities. Furthermore, these constructions can introduce scope ambiguities which I will ignore presently. An example for a collective/distributive ambiguity can be found in

(261) Some participants of a German expedition have a Swiss sponsor.

can mean that each of the participants belongs to a different German expedition or that they all belong to the same German expedition. The whole noun phrase *some participants of a German expedition* can then itself be read collectively or distributively with respect to the predicate *have a Swiss sponsor*.

The wide scope collective interpretation of some participants with respect to a German expedition will be represented as follows:

(262) some participants of a German expedition (collective interpretation)

```
[A,B,C,D]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
  [H]
  structure(H,atomic)
  part_of(H,B)
  =>
  []
  object(H,participant)
structure(D,atomic)
quantity(D,cardinality,C,count_unit)
value(C,eq,1)
property(D,'German')
object(D,expedition)
relation(B,participant,of,D)
```

The DRS introduces the condition `relation(B,participant,of,D)` that states that `B` and `D` stand in the participant relation. The argument `participant` is added for the following reason. If we introduced only the relation `relation(B,of,D)` we could not prevent inferences of the following type:

(263) Inference to be prevented

IF	John is a father of Mary. John is a son of Sue.
THEN	John is a son of Mary.

Therefore, the additional argument stating the type of the relation is added. I offer no account of relational nouns which would give an alternative solution to this problem.

The distributive reading of *some participants* with respect to *a German expedition* will be represented as

(264) some participants of a German expedition (distributive interpretation)

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
```

```
[F]
structure(F,atomic)
part_of(F,B)
=>
[]
object(F,participant)
[G]
structure(G,atomic)
part_of(G,B)
=>
[H,I]
structure(I,atomic)
quantity(I,cardinality,H,count_unit)
value(H,eq,1)
property(I,'German')
object(I,expedition)
relation(G,participant,of,I)
```

Other collective/distributive representations of prepositional phrases modifying noun phrases follow accordingly.

4.5.4.4 Floated Quantifiers and Adnominal Part-Structure Modifiers

Floated Quantifiers

In section 2.3.6, pp. 19 I have discussed that floated quantifiers like *each*, *all*, *both* trigger a distributive reading of the preceding noun phrase. The semantic effect of the floated quantifiers *each* and *all* in my setting is simply to trigger a distributive reading of the plural noun phrase. The quantifier *both* additionally adds the condition that the cardinality of the preceding noun phrase referent equals two. In the current fragment floated quantifiers can only occur directly after the noun phrase that is to be interpreted distributively.

Adnominal Part-Structure Modifiers

Besides floated quantifiers there are other elements of the language that have a disambiguating effect. In particular, there are noun phrase modifiers that do not act as "normal" modifiers but that elaborate on the part-structure of the noun phrase referent with respect to the eventuality of which the referent is a participant. Examples are *as a whole*, *as a group*, *together* that occur in adnominal position after the noun phrase.

(265) a. The oxygen bottles as a whole/together cost 2000 dollars.
b. John and Mary together have six oxygen bottles.
c. John carried three bottles as a whole.
d. The oxygen bottles as a whole are heavy.

A further example is the adjective *individual* that occurs before the modified noun.

(266) The individual oxygen bottles cost 50 dollars.
The individual oxygen bottles are light.

Furthermore, *same* and *different* in their so-called internal reading (Carlson 1987, Moltmann 1997, p. 136) have a disambiguating effect.

(267) John and Mary used the same tent/the same tents.
John and Mary used different oxygen bottles.

In the following I will propose a semantic representation for different types of part-structure modifiers.

As a Whole, As a Group, Together. The semantic effect of *as a whole*, *as a group* or *together* is that the modified object is to be evaluated with respect to the properties it has as whole, and not the properties of its individual parts. The addition of these modifiers blocks a distributive reading. Thus, in a way these modifiers can be seen as the "collective" counterpart of the floating-quantifier *each*. One way to give a semantics to these modifiers is to simply let them trigger a collective reading without adding additional conditions into the DRS. However, to achieve an analogous treatment to adverbial part-structure modifiers with the same or similar semantic functionalities I chose to add additional conditions into the DRS for these "collective" modifiers. Furthermore, the additional conditions are necessary to prevent certain inferences. More concretely, the DRS condition `elaboration(E,group_structure,coll,X)` is added. Where `E` stands for the eventuality, `group_structure` describes the dimension that the part-structure modifier elaborates on, `coll` indicates that it triggers a collective reading, and `X` is the discourse referent of the modified noun phrase.

(268) The oxygen bottles as a whole are heavy.

```
[A,B,C]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  []
  object(D,oxygen_bottle)
property(B,heavy)
elaboration(C,group_structure,coll,B)          % as a whole
structure(C,state)
predicate(C,be,B)
maximal(B,context,'the oxygen bottles')
```

In the representation of *as a whole*, *together*, *as a group* information about the original lexical realization of the part-structure modifier gets lost. If one wants to keep this information one would have to add an additional logical axiom for each modifier, e.g.

(269) ∀E∀X(elaboration(E,group_structure,as_a_whole,X) →
elaboration(E,group_structure,coll,X))

I chose the more direct but less informative representation for efficiency reasons. If necessary one can add additional logical axioms that specify the meaning of `elaboration/4` more closely. Since for the current investigation the fine-grained analysis is not necessary I will not further analyse these modifiers. The interested reader is referred to the investigations of Moltmann (1997).

Individual. The representation of the part-structure modifier *individual* triggers a distributive reading of the respective noun phrase. I have not investigated methods to determine with respect to which other element of the sentence the noun phrase is to be interpreted distributively.

Same/Different. The modifiers *same/different* occur in a number of constructions, see e.g. Moltmann (1997, pp. 135). They have an indexical reading (*John saw the same tree.*), appear in comparatives (*John found the same solution as Mary.*), get a bound interpretation with a quantified antecedent (*Everyone saw a different tree.*), and finally *same/different* can be anaphorically related to a plural NP. This reading is called – following Carlson (1987) – the internal reading. I will only briefly consider the internal reading here.

(270) Five climbers used the same tent.

means in a coarse approximation that the tent that each of the five climbers used was identical. I will not spell out this meaning directly in the DRS but I will also add a condition `elaboration/4` that can be amended by additional logical axioms.

(271) Representation of (270)

```
[A,B,C,D,E]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,eq,5)
  [F]
  structure(F,atomic)
  part_of(F,B)
  =>
  []
  object(F,climber)
structure(D,atomic)
quantity(D,cardinality,C,count_unit)
value(C,eq,1)
object(D,tent)
elaboration(E,D,same,B)            % same
structure(E,event)
predicate(E,use,B,D)
maximal(D,context,'the tent')
```

The representation expresses that there is an event `E` of using a tent `D` by five climbers `B`. The additional condition `elaboration(E,D,same,B)` is a coarse-grained abbreviation of the fact that for any two distinct proper parts `E1` and `E2` of `E`, the part of the tents used in `E1` (by a proper part of the five climbers) is the same as the part of the tents used in `E2` (by a proper part of the climbers). If desired this can be formulated within a first-order logical axiom. I have however not added this axiom to the practical implementation. The representation of *different* will accordingly add the condition `elaboration(E,D,different,B)`.

Since the internal use of *same/different* has an anaphoric element it can be a problem to determine the right antecedent. In the current simplified setting only the closest preceding plural noun phrase is chosen as an antecedent.

Adverbial part structures will be discussed in section 4.5.5.2 below.

4.5.5 Event Modification

With the introduction of event variables into the DRS the representation of event modification becomes possible. I will distinguish two types of event modification. First, I will consider "standard" modification (e.g. by adverbs and prepositional phrases) the meaning of which is independent of whether the modifiers are combined with singular or plural noun phrases. Second, I will consider so-called "part-structure modifiers" that relate group structure and event structure and thus require a different semantic treatment.

4.5.5.1 "Standard" Adverbs and Prepositional Adjuncts

Event modification is typically expressed by adverbs (*quickly*, *upwards*, *carefully*) or by prepositional phrases (*in the morning*, *onto the mountain*). In my representation event modifiers add the predefined condition `modifier/4` into the discourse representation structure. For adverbs like *slowly* in (272)a the condition is realized as (272)b.

(272) a. John walked slowly.
b. `modifier(E,manner,none,quickly)`

The variable `E` represents the event that is modified, `manner` corresponds to the modification type that is predefined via the lexical entry of *quickly*, the fourth argument represents the adverb itself, and the third argument `none` is a "dummy argument" that has the effect that `modifier/n` has the arity 4 for both adverbs and prepositional phrases.

If the event is modified by a prepositional phrase we add the condition (273)b.

(273) a. John reached the summit in the morning.
b. `modifier(E,time,in,A)`

The modification type `time` in (273)b is calculated from the type of the preposition and the type of the head noun of *the morning* according to an algorithm described in Schwitter (1998). The third argument represents the preposition, the fourth argument `A` corresponds to the discourse referent introduced by the noun phrase *the morning*.

Event modifiers can correlate to existing ambiguities or can even add more ambiguities. In example (80) – due to Parsons (1990) – here repeated as

(274) Samantha quickly polished some boots.

we have discussed that the whole polishing event can be quick or that each polishing of one of the boots can be quick. Similarly, sentences with prepositional phrase modifiers like (275) can correlate to distributive/collective ambiguities. The distributive reading will be represented as:

(275) Two clients climbed with a Sherpa. (distributive reading)

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,eq,2)
  [C]
  structure(C,atomic)
  part_of(C,B)
  =>
  []
  object(C,client)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  [E,F,G]
  structure(F,atomic)
  quantity(F,cardinality,E,count_unit)
  value(E,eq,1)
  object(F,sherpa)
  structure(G,event)
  predicate(G,climb,D)
  modifier(G,comitative,with,F)
```

In fact, the full range of collective/distributive ambiguities and scope ambiguities can occur if a sentence contains a prepositional phrase as event modifier.

(276) a. The climbers had 30 oxygen bottles at three camps.
b. The climbers had 10 oxygen bottles at every camp.

Again, the fundamental problem is to choose the correct interpretation with respect to collective/distributive ambiguities and scope ambiguities. I will come back to this in chapter 5.

4.5.5.2 Adverbial Part-Structure Modifiers

Introduction

In the previous section I have considered event modifiers the meaning of which is independent of whether they are combined with singular or plural noun phrases. In the semantic representation of these "standard" modifiers the event was taken as a primitive concept the structure of which has not been analysed. There are, however, other modifiers that relate the group structure and the event structure and thus require a different semantic treatment. Remember that to represent the indeterminacy of the collective reading I assumed possibly complex events that can but need not have a certain substructuring. The intuition behind assuming just one and only one event argument for the (indeterminate) description of the collective reading is described by Moltmann (1992) as:

> Even though this event argument may in principle be a group event whose members are distant in time and are otherwise independent from each other, it is preferably taken to be an event that has a certain degree of integrity, for instance with

> respect to time or with respect to the interaction of participants. This can be considered an instance of a very general condition on the individuation of entities. Entities that are semantic reference objects are 'better' the more they are integrated wholes, where the integrity of events can be constituted on the basis of connectedness in time or in space or on the basis of causal relations or the participation of other entities in the event. (Moltmann 1992, p. 428)

The indeterminacy of the collective reading can be reduced by adding event modifiers like *as a whole*, *as a group*, *individually*, *at the same time*, *one at a time* etc. that detail the internal structure of the event with respect to group participants. In Moltmann's words adverbially occurring event modifiers specify whether an entity or its parts are "integrated wholes" or belong to an integrated whole in a particular event (Moltmann 1992, p. 165). Examples for elaborations are:

(277) The Sherpa carried three oxygen bottles simultaneously/one by one/one after the other/one at a time/as a whole.

Although syntactically these adverbial modifiers behave like ordinary modifiers semantically they require a different treatment. There are several semantic theories that analyse modifiers of this type. Moltmann (1997) offers a detailed account discussing many natural language examples. She calls these modifiers *part-structure modifiers* – a concept that I will adopt for the current discussion. The semantics of the modifier *together* is controversially discussed by Schwarzschild (1991b) and Lasersohn (1995). Lasersohn (1995) concentrates on giving a unified semantics for the modifier *together*. Other relevant discussions can be found for example in Krifka (1989b), Eberle (1998) or Verkuyl (1999). I will not go into the details of the theories here. Most of the mentioned approaches offer a fine-grained analysis of part-structure modifiers and of event structures in general. For the practical applications of this thesis this precision will – as a starting point – not be necessary, nevertheless the foundation for further refinements should be established. Therefore, I will address certain basic insights here.

To describe the semantics of part-structure modifiers many of the approaches assume that the event domain is structured by a part-whole relation, i.e. it is assumed that complex events consisting of *subevents* exist. For example the sentence

(278) John lifted two tables simultaneously.

is considered to express one complex event E. This event E has two subevents E_1 and E_2 where one subevent E_1 corresponds to the lifting of one table and the other subevent E_2 corresponds to the lifting of the other table and the times of lifting the tables are identical. Note however, that the concept of a subevent is not unproblematic since the criteria for the individuation of subevents are much less clear than within the domain of countable objects. While group objects have as natural parts atomic individuals, subevents can be obtained in different ways. In particular events have to be understood as objects that are more than purely temporal units like time intervals or time points. An event may have both a spatial and a temporal part structure, and, moreover, it may have different part structures correlating with different sorts of event participants (Moltmann 1997). In the example (278) the part-structure of the event is determined by the plural object *two tables*. A reduction of event parts to temporal parts would rule out the possibility to call E_1 and E_2 *subevents* of E. Note that depending on the situation an event can have different part-structures. In contrast to the parts of groups the parts of events are not determined in advance.

The sentences in (279)–(284) below show that there are several dimensions with respect to which the part-structure of an event and its participants can be elaborated on. Possible dimensions can be temporal or spatial part structures where an event and its participants are structured with respect to temporal or spatial closeness, furthermore participant related part structures where event parts relate to participant parts. As further dimensions coordinated action, social accompaniment, assembly and others were proposed (Lasersohn 1995, Moltmann 1997). The examples show furthermore that some modifiers allow a unique assignment of the dimension (*simultaneously*, *at the same time*) whereas others can be used to express different dimensions (*together*). Furthermore, we see that the dimensions are not always independent of each other; some examples could be classified within different dimensions depending on the perspective. For example if a group of objects does something separately very often both a spatial and a participant-structure dimension are involved.

(279) temporal part structure

a. John and Mary stood up together. (Lasersohn 1995, pp. 231)

b. The system emits two photons at the same time.

c. Five FTP client sessions are active at the same time/concurrently.

d. The method displays rows one at a time, instead of all at once.

e. Photons are fired off one at a time.

f. John sold the apples three at a time. (Moltmann 1997, p. 213)

(280) spatial part structure

a. John and Mary sat together. (Lasersohn 1995, pp. 231)

b. Two climbers bivouacked at the same place/ together.

c. Two climbers bivouacked separately.

(281) group structure of participants

a. You can use these methods *individually* or *in combination*.

b. The system manages the internal entries *separately*.

c. The servers can also either be started *all together* or *individually*.

d. The immunglobulin domains unfold *one by one*.

e. John sold the apples *in groups of three*. (Moltmann 1997, p. 213)

f. Printed publications are available *individually*, *by order number*, or *in groups of books on related topics*, *by feature number* or *by BOF number*.

g. John and Mary have lifted the piano as *a group/together*. (Moltmann 1997, p. 166)

h. The students studied the problem *individually*. (Moltmann 1997, p. 167)

Besides spatial, temporal and participant-structure dimensions there are a number of other dimensions with respect to which the part-structure can be elaborated on. Following are suggestions by Lasersohn (1995, pp. 231).

(282) coordinated action
John and Mary worked together.

(283) social accompaniment
John and Mary went to the movies together.

(284) assembly
John put the bicycle together.

Although part-structure modifiers operate on different dimensions many of them seem to have a common meaning component, for example *together* has the effect of treating objects as a unit, be it spatially, temporally or with respect to a collective action.

Representation

There are several problems for the semantic treatment of part-structure modifiers in practical applications. First, part-structure modifiers that behave syntactically like other modifiers have to be distinguished from "normal" modifiers which requires additional non-syntactic knowledge. Second, part-structure modifiers relate events with group participants. This implies that there is an anaphoric component that has to be automatically resolved. This resolution can be problematic since ambiguities are possible. Third, it is difficult to automatically determine the dimension of the part-structure modifier. Fourth, using the concept of subevent to describe the semantics of part-structure modifiers leads to ontological difficulties. We have already addressed above that the criteria for the individuation of subevents are much less clear than within the domain of countable objects. As we have seen above an event may have different types of part structures and the relevant parts of an event are often contextually determined. Furthermore, the identity criteria for events are less clearly defined that the identity criteria for objects. Sentences can refer to different, still physically equivalent events, as is exemplified in the following examples due to Bach (1986):

(285) a. Jones poured poison into the water main. E_1
b. Jones poisoned the populace. E_2

According to Bach the events E_1 and E_2 refer to the same physical entity, but they function as different events.

Provided that a part-structure modifier is recognized, its antecedent is resolved and the dimension is determined there is still some leeway of how to describe the semantics of the modifier. Depending on the granularity of the representation different sets of entailments induced by part-structure modifiers can be reconstructed. A main question therefore concerns the intended granularity of the system: Do we need very fine-grained inferences that specify a situation very precisely, or are coarse-grained inferences sufficient for the intended applications?

I start with a proposal for a very coarse grained representation that can – if necessary – be amended by additional first-order axioms. For the practical application of this thesis I will assume that part-structure modifiers are lexically predefined and that the antecedent is the closest preceding individual denoting plural noun phrase. To simplify representation and reasoning I will only assume the three dimensions `time`, `space` and `group_structure`. Dimensions that cannot be clearly identified as temporal or spatial will be subsumed under the group-structure dimension. I have distinguished temporal and spatial dimensions since I consider them potentially important for temporal and spatial reasoning in practical applications. Sentences with part-structure modifiers are assigned a collective reading and a condition `elaboration(E,Dim,K,X)` is added to the DRS. The condition relates the event discourse referent `E` and the group discourse referent `X`, it expresses the dimension `Dim` of the modification and adds an (abstract) value `K` for the structuring. Depending on the dimension and the realization of the part-structure modifiers different conditions result as is shown in Table 6.

Dimension	DRS Condition	Examples
temporal	`elaboration(E,time,K,X)`	
	`K = same`	simultaneously, at the same time, concurrently, all at once
	`K = different`	at different times, one at a time, one after the other
	`K = three_at_time | any | …`	in any temporal order, three at a time,
spatial	`elaboration(E,space,K,X)`	
	`K = same`	at the same place
	`K = different`	at different places
	`K = side_by_side | …`	side by side, one behind the other
group-structure	`elaboration(E,group_structure,K,X)`	
	`K = coll`	collectively, jointly, together, as a whole, as a group
	`K = distr`	one by one, separately, individually
	`K = in_groups_of_3 | …`	in groups of three, by order number, …

Table 6 DRS Conditions for Part-Structure Modifiers

Thus the representation for the following sentence (286) is based on a collective reading plus an additional condition for the elaboration marker.

(286) John lifted two tables simultaneously.

```
[A,B,C,D,E]
named(B,'John')
structure(B,atomic)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
structure(D,group)
quantity(D,cardinality,C,count_unit)
value(C,eq,2)
  [F]
```

```
      structure(F,atomic)
      part_of(F,D)
      =>
      []
      object(F,table)
    structure(E,event)
    predicate(E,lift,B,D)
    elaboration(E,time,same,D)                 % simultaneously
```

Non-temporal elaboration markers like *as a group* or *as a whole* add the condition `elaboration(E,group_structure,coll,D)`. As with adnominal part-structure modifiers the additional condition `elaboration/4` already abstracts the concrete realization of the part-structure modifier and adds predefined values if the part-structure modifier has either a collectivizing or a distributive use. There are also part-structure modifiers that are not at one or the other end of the spectrum, e.g. *in groups of three*, *three at a time*. I have currently no proposal for a compositional treatment of these modifiers. Currently, they have to be lexically predefined and are directly represented in the DRS as e.g. `elaboration(E,group_structure, in_groups_of_3,X)`. Again I recommend Moltmann (1997) for a detailed discussion of these modifiers.

Inferences

These very coarse grained representations can be made more precise by adding additional first-order axioms specifying the meaning of the elaboration markers more precisely. These additional axioms will take recourse to event parts and object parts. The event part relation is indicated as `e_part_of(E1,E2)` and the proper event part relation is abbreviated as `proper_e_part_of`. For example the elaboration marker *simultaneously* could trigger the following axiom:

(287) $$\begin{aligned}&\forall E\forall P\forall X\forall Y(\text{predicate}(E,P,X,Y) \wedge \text{elaboration}(E,\text{time},\text{same},Y) \rightarrow\\ &\quad\forall Y_1\forall Y_2\forall E_1\forall E_2(\text{structure}(Y_1,\text{atomic}) \wedge \text{part\_of}(Y_1,Y) \wedge \text{structure}(Y_2,\text{atomic})\\ &\qquad\wedge \text{part\_of}(Y_2,Y) \wedge \neg(\text{is\_equal}(Y_1,Y_2)) \rightarrow\\ &\quad\exists E_1\exists E_2(\text{proper\_e\_part\_of}(E_1,E) \wedge \text{proper\_e\_part\_of}(E_2,E)\wedge\\ &\qquad\neg(\text{is\_equal}(E_1,E_2)) \wedge \text{predicate}(E_1,P,X,Y_1) \wedge\\ &\qquad\text{predicate}(E_2,P,X,Y_2) \wedge \text{temporal\_order}(E_1,\text{same},E_2))))\end{aligned}$$

Here `temporal_order/3` is yet another unanalyzed condition that expresses the temporal relation between events. These relations are needed for temporal reasoning – a topic that I will not go into detail here.

I have considered further possible axioms detailing the meaning of part-structure modifiers, however, I have not integrated these axioms into the practical application. This means that currently – as with normal modifiers like *quickly*, *into the building*, etc. – the further interpretation of the part-structure modifiers is left to the human interpreter of the text. The main reason is that for the current applications the fine grained inferences that the additional axioms make possible are not necessary and would thus lead to an unnecessary complication of the inference engine. However, the syntax of the representation allows a straightforward integration of additional axioms. For example, as soon as temporal reasoning is added it is possible to further resolve the temporal elaboration markers.

It is also possible to add axioms that directly relate the elaboration markers of different dimension to each other. For example, if we wanted to reconstruct the inference

(288) IF John lifts five tables one at a time.
THEN (?) John lifts five tables individually.

we would have to add the following axiom.

(289) ∀E∀D(elaboration(E,time,different,D) →
elaboration(E,group_structure,distr,D))

I have not added these types of axioms to the practical application, since I am not sure whether we really want these inferences to be valid.

Discussion, Problems, Limitations

The previous sections explained some of the problems related to the semantics of part-structure modifiers. Part-structure modifiers relate object parts to event parts along a certain dimension. I gave a coarse grained representation of this relation that can – if necessary – be amended by additional axioms. The approach relies on several (simplifying) assumptions. First, part-structure modifiers have to be lexically predefined. Second, no compositional derivation of part-structure modifiers that are not lying at one or the other end of the spectrum is given. Furthermore, the anaphoric relation between the event and the group discourse referent is determined by a simple algorithm. Finally, the dimension has to be lexically given although in real texts it is often contextually determined. These simplifications are justified if we do not expect fine grained inferences concerned with part-structure modifiers. Modifications are necessary if we add for example temporal reasoning.

Moreover, I have not investigated what the scopal effects of part-structure modifiers are. Does a sentence like

(290) John gives two books separately to a child.

have a reading where there are two books each of which John gives to a different child? Currently, due to the initial assignment of a collective reading no distributive reading with a possible scopal effect is generated.

4.5.5.3 Temporal Information

It is possible to add tense information and temporal order of events to the DRS representations discussed so far. Since I have not investigated temporal reasoning, I will only provide a simple representation of temporal information that is open for further extension.

Tense Information

Tense information states whether the predicate expresses present, past or future tense. In the current system I only integrated simple present and simple past tense. Tense is represented by an additional condition `tense/2` that relates the event to the tense. For example, if tense information is relevant, we can represent the past tense of *sleep* as

(291) slept

```
[C]
structure(C,state)
predicate(C,sleep,B)
tense(C,past)
```

The present tense is represented accordingly by adding the condition `tense(E,present)`. The addition of tense information to the DRS prevents inferences of the following type.

(292) IF John slept.
~~THEN~~ John sleeps.

No further inferences related to tense in natural language are currently considered.

Temporal Structure of Events

Sometimes events are explicitly ordered in a text by adding *and then*, or by adding temporal conjunctions like *before* or *after*. Also the modifiers *simultaneously*, *at the same time*, *in any temporal order* etc. can be used not only as part-structure modifiers that relate groups and events but also as modifiers that relate two or more explicitly mentioned events. Here are some – grammatically simplified – examples:

(293) a. John looks for water *and then* pitches the tent.
b. *Before* John lights the stove, he builds a wind protection.
c. John stirs the soup and *at the same* time eats chocolate.
d. John eats the soup and *at a different time* drinks the tea.
e. John checks the map *while* the soup is simmering.
f. *While* the soup is simmering John checks the map.
g. *After* John finishes the meal he cleans the pot.

In the DRS the temporal order can be represented by adding an additional condition `temporal_order/3` relating two events and stating their relative temporal order. Possible values are shown in Table 7.

DRS Condition	Paraphrase	Examples	
`temporal_order(E1,K,E2)`			
`K = before`	E1 is before E2	before	(293)a,g
`K = after`	E1 is after E2	after, and then	(293)b
`K = same`	E1 is at the same time as E2	and simultaneously, and at the same time,	(293)c
`K = different`	the time of E1 is different from the time of E2	at a different time	(293)d
`K = part_of`	E1 is included in E2	while	(293)e
`K = include`	E1 includes E2	while	(293)f
`K = any`	the temporal relation of E1 and E2 is arbitrary	in any temporal order	
`K = overlap`	the time of E1 overlaps with the time of E2		

Table 7 DRS Conditions for Temporal Order of Events

Not all temporal relations have natural correspondences in the language. I have sill included the relations since they may be needed for temporal reasoning.

4.5.6 Predicative Constructions with the Copula 'Be'

4.5.6.1 Overview

In the logical tradition of Frege and Russell several uses of the copula *be* are distinguished. The copula can express:

(294) a. predication
Tenzing is strong.
b. class inclusion
Some Sherpas are climbers.
c. identity
The strong Sherpa is Tenzing.
d. existence
Sokrates is.

The *be* of predication attributes a property to an individual. It can be expressed by adding an adjective complement or a prepositional phrase (section 4.5.6.4 and section 4.5.6.5). The *be* of class inclusion is typically constructed by adding a singular or plural indefinite noun phrase as a complement (section 4.5.6.2). The *be* of identity requires two definite noun phrases. They can be realized as proper nouns, definite descriptions, or coordinations thereof (section 4.5.6.3). In my fragment, the existence use of the copula is expressed by *there is* constructions (section 4.5.2.1 above). Since the copula can combine with plural noun phrases collective/distributive ambiguities are possible. In the following I will order the representations of the copula according to the different types of complements.

4.5.6.2 Indefinite Noun Phrase Complements

Singular and Plural Indefinites

Following are the DRS representations for singular and plural indefinites (ignoring collective nouns for the moment). The constructions express a form of class inclusion. Apart from introducing a discourse referent denoting a state (here `C`) the semantic function of the copula is empty (as long as tense is ignored).

(295) Tenzing is a Sherpa.

```
[A,B,C]
named(B,'Tenzing')
structure(B,atomic)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
object(B,sherpa)
structure(C,state)
predicate(C,be,B)
```

(296) Some Sherpas are strong climbers.

```
[A,B,C]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  []
  object(D,sherpa)
  [E]
  structure(E,atomic)
  part_of(E,B)
  =>
  []
  object(E,climber)
  property(E,strong)
structure(C,state)
predicate(C,be,B)
```

Indefinites with Collective Nouns

The copula can also combine with collective nouns, i.e. nouns that can only be true of groups (*team*, *committee*, *group*). The corresponding representation is:

(297) The Sherpas are a small group.

```
[A,B,C]
structure(B,group)
quantity(B,cardinality,A,count_unit)
```

```
value(A,geq,2)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  []
  object(D,sherpa)
property(B,small)
object(B,group)
structure(C,state)
predicate(C,be,B)
```

4.5.6.3 Definite Noun Phrase Complements

I will represent constructions with the copula *be* and two definite noun phrases (definite descriptions, proper nouns or conjunctions of definite noun phrases) as expressing identity between two discourse referents. The construction introduces the condition `property(B,equal,D)`. Again the *be* itself only contributes the state discourse referent `E`.

(298) Tenzing is the leading sherpa.

```
[A,B,C,D,E]
named(B,'Tenzing')
structure(B,atomic)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
structure(D,atomic)
quantity(D,cardinality,C,count_unit)
value(C,eq,1)
property(D,leading)
object(D,sherpa)
property(B,equal,D)
structure(E,state)
predicate(E,be,B)
maximal(D,context,'the leading Sherpa')
```

There are two possibilities to "resolve" the equality expressed in the condition `property(B,equal,D)`. First, one could equate the discourse referents directly within the DRS, i.e. any occurrence of `D` would be replaced by `B` and the condition `property(B,equal,D)` and the discourse referent `D` would have to be deleted from the DRS. Another possibility is to resolve the identity during inferencing. This would require the following additional first-order axiom:

(299) $\forall X \forall Y(\text{property}(X,\text{equal},Y) \rightarrow \text{is\_equal}(X,Y))$

plus the identity axioms for `is_equal/2`.

If two definite *plural* noun phrases are arguments of the copula *be* the following representation is generated.

(300) The mountain guides are the Sherpas.

```
[A,B,C,D,E]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
```

```
    [F]
    structure(F,atomic)
    part_of(F,B)
    =>
    []
    object(F,mountain_guide)
  structure(D,group)
  quantity(D,cardinality,C,count_unit)
  value(C,geq,2)
    [G]
    structure(G,atomic)
    part_of(G,D)
    =>
    []
    object(G,sherpa)
  property(B,equal,D)
  structure(E,state)
  predicate(E,be,B)
  maximal(D,context,'the mountain guides')
  maximal(B,context,'the Sherpas')
```

An analogous representation is generated if the definite noun phrase is formed by a conjunction as in

(301) a. Tenzing and Pemba are the mountain guides.
b. The climbing clients are John, Bill and Mary.

Note, that the plural constructions trigger a collective reading: a group of guides is set equal to a group of Sherpas. Nothing is said about which guide is equal to which Sherpa.

4.5.6.4 Adjective Complements

Intransitive Adjectives

Many adjectives are inherently distributive, that is they can only be true of individuals, e.g. *red*, *tired*, *injured*. There are also adjectives like *heavy*, *strong* etc. that can be true of both individuals and groups which results in possible ambiguities. Furthermore, adjectives like *identical*, *similar*, *equal*, *different* can only be true of groups.

(302) Tenzing is strong.

```
[A,B,C]
named(B,'Tenzing')
structure(B,atomic)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
property(B,strong)
structure(C,state)
predicate(C,be,B)
```

To represent the occurrence with plural subjects there are two possibilities. One possibility is to put the variable representing the state outside the universal quantification as in (303).

(303) Some Sherpas are strong. (distributive reading of adjective, version 1)

```
[A,B,C]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  []
  object(D,sherpa)
  [E]
  structure(E,atomic)
  part_of(E,B)
  =>
  []
  property(E,strong)
structure(C,state)
predicate(C,be,B)
```

A second possibility is to put the variable inside the universal quantification as in (304).

(304) Some Sherpas are strong. (distributive reading of adjective, version 2)

```
[A,B]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
  [C]
  structure(C,atomic)
  part_of(C,B)
  =>
  []
  object(C,sherpa)
  [D]
  structure(D,atomic)
  part_of(D,B)
  =>
  [E]
  property(D,strong)
  structure(E,state)
  predicate(E,be,D)
```

The advantage of the second representation is that the inference

(305) if	Some Sherpas are strong.	
then	A Sherpa is strong.	

comes out as valid without introducing auxiliary meaning postulates for the copula that distribute the state down to the atomic parts of the subject.

The collective reading of the adjective is represented as (306).

(306) The Sherpas are strong. (collective reading of adjective)

```
[A,B,C]
structure(B,group)
quantity(B,cardinality,A,count_unit)
value(A,geq,2)
```

```
    [D]
    structure(D,atomic)
    part_of(D,B)
    =>
    []
    object(D,sherpa)
  property(B,strong)
  structure(C,state)
  predicate(C,be,B)
  maximal(B,context,'the Sherpas')
```

Adjectives like *identical*, *equal* as in

(307) The two values are equal.

trigger a collective reading and add the condition `property(B,equal)`. If necessary for a certain domain, adjectives like *identical*, *equal* can be equipped with further axioms that detail their meaning.

Transitive Adjectives

There are also transitive adjectives like *fond of*, *equal to* etc. They are predefined in the lexicon and add the conditions `property/3`, e.g. `property(B,fond_of,D)` or `property(B,equal,D)`. Collective and distributive ambiguities are possible.

Comparison

I have currently no theory of comparatives. The comparative is simply represented as a transitive relation, e.g. `property(B,smaller_than,D)`, or as a ditransitive relation, e.g. `property(B,fonder_of_than,D)`. The superlative is represented as an intransitive adjective, `property(B,smallest)`. A proper theory of comparatives has to relate the uses of positive, comparative and superlative systematically.

4.5.6.5 Prepositional Phrase Complements

Copulae with prepositional phrase complements express e.g. the location of an object with respect to a state as in

(308) Tenzing is in a tent.

```
[A,B,C,D,E]
named(B,'Tenzing')
structure(B,atomic)
quantity(B,cardinality,A,count_unit)
value(A,eq,1)
structure(D,atomic)
quantity(D,cardinality,C,count_unit)
value(C,eq,1)
object(D,tent)
structure(E,state)
predicate(E,be,B)
modifier(E,location,in,D)
```

If plural noun phrases occur as arguments collective/distributive ambiguities are possible. Furthermore, due to the occurrence of two noun phrases, scope ambiguities can occur.

(309) Three climbers are in an expedition tent.
(310) An oxygen bottle is in every tent.

The representations for collective/distributive readings follow straightforwardly from the above considerations.

4.5.7 Representation of Questions

Introduction

There are two main types of questions: *yes/no*-questions and *wh*-questions. *Yes/no*-questions such as

(311) a. Is the oxygen bottle full?
b. Did Tenzing reach Everest?

can be answered by *yes* or *no*. A more informative answer additionally gives the evidence that lead to the positive or negative answer. In contrast, so-called *wh*-questions such as

(312) a. Who reached Everest?
b. What did Tenzing carry?
c. When did Tenzing reach Everest?
d. Which oxygen bottles did Tenzing carry?
e. How many bottles did Tenzing carry?

expect an answer that specifies a particular person, thing, place, time, property, amount etc.

In proof-theoretic approaches questions are not directly added to the knowledge base created by a natural language discourse but questions are used to examine the contents of the knowledge base. In my approach query answering is performed by logical deduction. This requires that the logical representation of the question is provable from the knowledge base, and – what is more – that there is a mechanism that keeps track of the sentences necessary for the proof. In this section I am only concerned with the logical representation of the question, tracking will be dealt with in chapter 6.

Representation

'Yes'/ 'No'-questions. In the current setting, *yes/no*-questions can only be uttered in a "query context". They are translated like the corresponding positive statement, but additionally trigger that a proof of the positive statement from the knowledge base is performed. If the proof succeeds we do not only get the answer *yes* but also the sentences necessary to deduce the positive answer. If the proof fails, we get the answer *no*.

'Wh'-questions. All *wh*-query-words introduce the condition `query(I,QueryWord)`. Additional DRS conditions depend on the query-word used. The conditions are designed to guarantee that a proof from the knowledge base succeeds if there is an answer to the question. Before the proof is performed the condition `query/2` is filtered out.

The query words *who* and *what* ask for persons or objects. Mostly this information is expressed by full noun phrases. Since it is not clear whether we ask for an atomic individual, a group or a mass object the query word additionally introduces the condition `structure(I,dom)`:

(313) who (accordingly what)

```
[A]
query(A,who)
structure(A,dom)
```

The query word *which* asks for additional properties of objects. Mostly this additional information is expressed via noun modifiers.

(314) which

```
[A]
query(A,which)
```

In (314) the discourse referent `A` corresponds to the referent of the modified noun.

The query words *where*, *when*, *how* etc. mostly ask for information expressed by VP modifiers.

(315) where (accordingly when, how)

```
[D,E]
query(C,where)
modifier(C,where,D,E)
```

In (315) the eventuality discourse referent `C` is introduced by the verb. During the proof the query words *where*, *how* etc. are associated with appropriate modification types (see section 4.5.5 above) via additional first-order axioms (see axioms in (Ax. 36) below).

Questions formed by prepositions and query words such as *from where*, *for whom*, *until when*, *with what*, *how long*, *how often* etc. are syncategorematically predefined and introduce conditions such as

(316) until when

```
[D,E]
query(C,until-when)
modifier(C,until-when,D,E)
```

Again the condition `modifier/4` is associated with the corresponding modification types via additional axioms as described below.

The query word *how many* asks for amounts of objects. This information is typically expressed by numbers. A question like

(317) How many climbers reached Everest?

will retrieve any sentence that contains a statement about existing climbers that reached Everest. This behaviour is achieved by the following representation.

(318) how many

```
[A,B,C]
query(B,how_many)
structure(B,dom)
quantity(B,cardinality,A,count_unit)
value(A,geq,1)
```

I have discussed in section 4.3.2, pp. 94 that a cumulative treatment of *how many* questions is not supported by the current system. Thus, I have no solution for counting objects in a database.

Inferences

Assume we have the following discourse:

(319) a. Two German climbers together use a British tent.
b. Two Swiss climbers each use a Swedish tent.
c. Two French climbers together use a Swedish tent.

The query

(320) Who uses a tent?

can be answered on the basis of each of the three sentences in (319). To derive the query representation from sentence (319)a and (319)c only the additional domain axiom (Ax. 1) is necessary. To derive an answer from (319)b we need the axioms (Ax. 10)-2, (Ax. 2) and the definition of the proper part relation (Ax. 9). Note that the current representation of the question (320) does not allow to retrieve an answer from the sentence

(321) Every climber used a Swedish tent.

unless the existence of a climber is stated elsewhere in the discourse.

The query

(322) Which climbers use a Swedish tent?

can be answered on the basis of (319)b using the axioms (Ax. 2), (Ax. 9), (Ax. 11), (Ax. 15)-1, the number axiom (Ax. 22)-1 and the axiom (Ax. 10). The question can also be answered on the basis of (319)c using only the axioms (Ax. 1) and a number axiom (Ax. 22)-1. Note that the representation of (322) can be derived from both a collective and a distributive reading. If only a distributive reading is desired the question has to be reformulated as

(323) Which climbers each use a Swedish tent?

Assume the following discourse:

(324) a. Two Sherpas waited at Camp IV.
b. Four Sherpas waited at Camp II.
c. Tenzing climbed Everest from Camp IV.

To answer queries that ask for adverbial modifiers such as

(325) Where did Sherpas wait?
(326) From where did Tenzing climb Everest?

additional axioms are necessary. These axioms correlate modification types with query words. For example we need axioms of the following type:

(Ax. 36)
1. $\forall E \forall P \forall L$(modifier(E,location,P,L) $\rightarrow$ modifier(E,where,P,L))
2. $\forall E \forall P \forall L$(modifier(E,time,P,L) $\rightarrow$ modifier(E,when,P,L))
3. $\forall E \forall P \forall L$(modifier(E,origin,P,L) $\rightarrow$ modifier(E,from-where,P,L))
4. $\forall E \forall P \forall L$(modifier(E,duration,P,L) $\rightarrow$ modifier(E,how-long,P,L))
5. …

The association of query words with modification types could also take place already in the lexical entry of the query word. For reasons of flexibility I have chosen the axiomatic approach. The same technique could also be applied to associate query words like *who* or *what* with object types like `person` and `object`. Currently, this is not yet considered.

I analyse the query word *how* as insensitive to a particular modification type. This is achieved by the following axiom:

(Ax. 37) $\forall E \forall H \forall P \forall L$(modifier(E,H,P,L) $\rightarrow$ modifier(E,how,P,L))

As a consequence, by using the query word *how* we can get information about any verb modifier.

Discussion, Problems, Limitations

In general, the syntax and semantics of questions constitutes a research topic on its own. I have said nothing about syntactic problems here. Concerning semantic issues my main concern was to show how questions can be treated within a proof-theoretic approach. The problem of tracking the answers will be dealt with in chapter 6. As to the representation of questions related to plurals my main point is the assumption of a rather indeterminate logical form. This logically weak representation allows – in many cases – to retrieve both collectively and distributively read sentences.

There are several problems and shortcomings with the suggested method for query answering. First, as addressed above, I have no mechanism to treat cumulative queries, that means e.g. numbers are not added up when a *how many* question is asked. Furthermore, the proof-theoretic approach requires that the wording and the syntax of the question is very close to the original text. This has the effect that recall may not be very high, though precision is. Furthermore, in the current setting, query answering consists of reporting whole sentences to users. This may be inadequate, since the user may only be interested in the queried element itself, e.g. noun phrases for *who*-questions. A further problem that was exemplified by (321) is that *wh*-questions cannot be answered if the queried objects do not exist. There are several approaches to implement a more fine-grained analysis of questions that overcomes this last problem. For a recent article on this issue see for example Bos and Gabsdil (2000).

Research on question answering within computational semantics abounds and it is beyond the scope of this thesis to address the issue in more detail here. A number of relevant references can be found in Bos and Gabsdil (2000).

4.6 Evaluation

The previous sections defined flat first-order discourse representations for many difficult plural constructions. Many of the described constructions are traditionally taken as an argument for introducing higher-order semantic representations, e.g. non-monotone increasing quantifiers like *at most n*, *exactly n* or proportional quantifiers like *most*. My approach shows how these quantifiers can be reconstructed in a computationally more suitable first-order language so that most of the intended inferences attributed to the generalized quantifiers can be reconstructed. An important design principle of the proposed representation was to avoid over-precision and to represent just the information needed for reasoning processes related to text understanding. This relates to my decision to just assume distributive and indeterminate collective readings for plurals. To practically implement reasoning processes it was important to use *flat* representations that allow to quantify over "predicates" in a first-order language. Technically, this flat approach allows to integrate first-order auxiliary axioms for plurals, identity, mathematical relations and others. Furthermore, the first-order approach allows to use off-the-shelf first-order theorem provers that by now have reached a high level of maturity. This shows the generality, flexibility and re-usability of my approach. However, choosing a flat first-order notation also has the consequence that the discourse representation structures contain many conditions which may reduce the efficiency of the associated theorem provers. It therefore needs to be investigated whether the notation can be formulated more compactly, e.g. by summarizing the conditions for `value/3` and `quantity/4` into one condition. Also the status of the maximality conditions with respect to efficiency is still a matter of consideration.

My investigation was necessarily restricted to a limited fragment of important plural phenomena. Further research will have to extend the coverage of the language. In particular, I consider it to be interesting to investigate which effects a more precise lexical semantics of verbs could have. Which information can be reasonably extracted from machine-readable lexica? How is the information represented and which meaning postulates have to be assumed? Does a precise lexical semantics scale up to automated reasoning with the representations?

My approach defines disambiguated logical representations for sentences that can be completely parsed. I have not investigated solutions for incomplete parsing, partial disambiguation or partial semantic representation. These techniques are necessary to make a system robust and flexible with respect to input not covered by the rules of the system. However, as soon as partially specified representations are allowed one also has to define reasoning methods that operate on these logical forms. In general, research on partially specified representations constitutes an important field for further research. It would be particularly interesting how my proof-theoretic perspective to (plural) semantics relates to the problem of representing partial information and to reason with partially specified logical forms.

5 An Algorithm To Reduce Plural Ambiguities

5.1 Overview

The aim of this chapter is to develop a plural disambiguation algorithm that is based on the interaction of a set of computationally manageable rules. The algorithm should make empirically adequate predictions. Moreover, for computational applications other criteria are also important. Therefore the algorithm will be formulated in the sense of Hobbs and Shieber (1987):

> The algorithm […] can provide a solid foundation for computational solutions where completeness is sacrificed for efficiency and heuristic efficacy." (Hobbs and Shieber 1987, p. 47)

The plural disambiguation algorithm will use information available in the text plus lexical information that can be automatically extracted. I will show a prototypical implementation of the algorithm. The algorithm will offer the best reading or, alternatively, a selection of several preferred readings ordered according to preference. The advantages of the algorithm compared to other approaches are that the influence of different disambiguation factors can be integrated, that the algorithm offers an ordered set of plausible readings (with the option to choose but the best reading), that the formulation of the rules does not rely on world-knowledge or context and, finally, that the integration of new disambiguation sources (e.g. more fine-grained semantics of verbs) can be neatly added.

Section 5.2 gives an overview over existing approaches to plural and scope disambiguation. I will adopt some insights worked out in the empirical studies 5.2.1 and I will adopt some techniques introduced in 5.2.2. In section 5.3 I will summarize the accessible information sources that I will use for the automatic disambiguation of collective/distributive ambiguities plus some resulting scope ambiguities. Note that I will only briefly address scope-ambiguities but will not include a complete algorithm. For scope ambiguities there is much more literature than for plural ambiguities. I will, however, show an approach how the two related problems can be solved. Section 5.4 introduces the basic algorithm for the automatic disambiguation of plural ambiguities based on the disambiguation information elaborated in 5.3. This algorithm is currently prototypically implemented within the my system DRoPs (Disambiguating and Reasoning with Plurals). I will show the data structures used for the implementation and explain how parts of the algorithm are implemented on the basis of the data structures. In chapter 6 I will then show how the DRoPs disambiguation algorithm is complemented by theorem proving techniques which allow the system to derive further information from disambiguated structures.

5.2 Existing Approaches to Plural Disambiguation

5.2.1 Empirical Studies

5.2.1.1 Difficulties

Linguistic theories are often criticized because they base their data only on individual judgments and not on empirically based studies. In the realm of quantification and quantifier scope, and also in the related realm of determining collective/distributive ambiguities empirical studies are very rare. Hardly have cross-linguistic, dialectal and idiolectal variation, or the behaviour of an individual speaker over time been systematically taken into account.

> It should thus be quite clear, that the introspective judgements of a single linguist (however honest and well-trained he or she may be) cannot serve as the sole data base for the development of an observationally adequate logic for quantification and quantifier scope in natural language. (Gil 1982, p. 425)

To perform thorough and extensive empirical studies based on a sufficiently large set of data and a sufficiently large number of informants is, however, a project in its own which I have not put into the focus of this thesis.

For the empirical foundation of my disambiguation algorithm I examined the results of empirical studies that – though rather concerned with quantifier scope – provide a number of important results also for plural disambiguation. I will summarize these results in section 5.2.1.2 below. During my own "introspective" empirical studies on plural disambiguation I verified some of these results on my own data set. The data set consisted of a number of machine readable English technical texts (see appendix C.1). I run these texts through a POS tagger and I could then investigate certain constructions using the IMS Corpus Workbench (see appendix C.2). This allowed me to check the importance of certain plural constructions and to determine – admittedly via introspection – possible and preferred readings. Since this dissertation is only concerned with English technical texts no cross-linguistic studies were necessary. I have undertaken no systematic quantitative studies on the types of constructions, their frequency and their possible or preferred readings. Empirical research in this direction would definitely constitute a challenging and important project, in particular, since systematic studies on possible and plausible readings of plural constructions will face a number of hard problems. Some of these problems also occurred in the empirical studies on scope ambiguities.

A first problem that occurs with collective/distributive ambiguities and also with other semantic ambiguities is that there is often a lack of clear intuitions on the exact interpretation of the respective sentences. One reason may be that semantic concepts like “collective” and “distributive”, or the concept of “scope” are not fully adequate to describe the intuitions, in particular for a non-linguistically trained informant. For instance in section 4.3.2 I have discussed examples were the distinction of a collective and a distributive reading for the internal argument position of certain verbs makes no reasonable difference.

I have also addressed that a number of different factors determines the preference for a certain reading. The difficulty in general and in empirical studies in particular lies in distinguishing syntactic or structural factors from other types of information, e.g. discourse context, pragmatic knowledge or lexical knowledge, which are less accessible to an adequate formalization. For example, Gil (1982) based an empirical study on the example

(1) Three boys saw two girls.

The example uses the predicate *see* which usually cannot have a real collective reading in its subject position because seeing is typically an activity that single individuals perform. Gil’s results state that the most highly preferred readings for (1) is the strong symmetric reading, i.e. a reading where there are three boys and there are two girls, and each of the boys sees each of the girls. However, I doubt the predictions of his empirical study since using a different verb, e.g. an agentive verb like *lift* would definitely change the preferences. The predicate *lift* can lead to a real collective reading: it makes a difference whether a lifting event is collective or distributive. Similar remarks hold for the choice of arguments. Due to our world-knowledge a sentence like

(2) Three boys ate two apples.

will most probably be interpreted distributively, whereas the sentence

(3) Three boys ate two cakes.

will rather get a collective reading. In many examples it is – in particular for an untrained user – very difficult to distinguish the different sources that lead to a preferred interpretation and that decide whether an interpretation is possible, plausible, or impossible. This point has also been stressed by VanLehn (1978, pp. 7). VanLehn empirically examined factors that determine scope disambiguation. He states:

> The relative strengths of the lexical and syntactic influences is significantly different for quantifier scope than for other linguistic phenomena. Lexical content is much more important in quantifier scope judgements than in, say, the acceptability of np movements or definite np anaphora. (VanLehn 1978, p. 7)

For example the antecedent of a reflexive pronoun usually requires that the antecedent occurs in the same clause as the anaphora. In general, scope is also clausebound. For example in the sentences

(4) a. John blurted out that each senator was offered a TV set.

b. A TV set blurted out that each senator was offended.

the existential NP *a TV set* can only be in the scope of the universal NP *each senator* if the NP occurs in the same clause. However, as VanLehn states, it is not difficult to use lexical content to override the clauseboundedness of quantifier scope. An example is

(5) A quick test confirmed that each drug was psychoactive.

The sentence has a likely interpretation that for each drug there is a different test – a possible reading that is in contrast to the example (4)b above. VanLehn argues that the examples show that lexical content is strong enough to violate almost any syntactic constraint one could develop. Therefore a theory of quantifier scope is inherently different from many other linguistic theories that have been developed. VanLehn then hypothesizes that – unlike real syntactic processes like coreference resolution – quantifier scope is *not* a real process but an epiphenomenon which VanLehn considers to be a main reason for the difficulties in empirical data collection. He observed that informants sometimes simply couldn't make a decision about the interpretation of a certain sentence, or that informants had to think very hard, e.g. when syntactic structure suggests a different reading than lexical content. These difficulties seem to indicate that quantifier scope disambiguation is done *after* informants have understood the sentence in some or the other way.

As to my opinion the same observations hold for plural disambiguation. Certain syntactic structures prefer certain readings, or correlate incidentally, but the rules determining these preferences are not strong enough to "resist" readings that are more plausible due the lexical content of words, our world knowledge or other factors I have discussed above. In this thesis I tried to extract as many syntactic disambiguation factors as possible, but the reader has to be aware that the predicted preferences can – in many cases – be overridden by other factors.

5.2.1.2 Summary of Existing Empirical Studies

General Remarks

Most remarks on plural disambiguation are spread over the plural literature and are based on personal observations rather than on empirical studies.

There are a number of empirical studies (Ioup 1975, VanLehn 1978, Kurtzman and MacDonald 1993) that investigate possible and plausible scopings of quantifiers. Although these studies are only marginally investigating collective/distributive ambiguities they offer important insights for plural disambiguation. I am aware of only one study that explicitly investigates collective/distributive ambiguities (Gil 1982) and – although this article offers interesting methodological remarks – Gil's examples are too specific to be of overall use for the problem of plural disambiguation.

Scopal issues and collective/distributive readings interact. In particular, if a noun phrase gets a distributive (i.e. universally quantified) reading its relative scope to other noun phrases becomes important. Empirical studies have not investigated how the disambiguation of scope and collective/distributive readings interact, e.g. whether one precedes the other, whether one constrains the other etc. Most of the empirical studies do not clearly distinguish the two problems. Often it is assumed that distributivity is parallel to scope, which implies that if a noun phrase has wide scope it is interpreted distributively (and if it has narrow scope it is interpreted collectively, if possible). However, we have seen examples that show that scope and distributivity are (in principle) independent. That means there are also narrow scope distributive readings of noun phrases. For example

(6) John told a story to several children.

can mean that the same story is told to each of the children in separate events. That means although *several children* has narrow scope it can be interpreted distributively. The misleading assumption that scope is parallel to distributivity often results from the choice of the examples. For example, in

(7) John gave several girls a cake.

the wide scope reading of *a cake* and a narrow scope distributive reading of *several girls* would mean that the same cake is given to each of a number of girls. Since this situation doesn't make much sense the narrow scope distributive reading is difficult to perceive. However, this influence is pragmatic, and thus the example cannot be used to argue that narrow scope readings are *always* collective, even though the narrow scope collective reading may occur more frequently.

Therefore, in principle scope is independent of distributivity. One cannot infer one from the other. Only if both noun phrases are interpreted collectively, their relative scope makes no difference for the interpretation.

Still, both distributivity and scope have to be taken into account since investigating the influence of scope presupposes a distributive reading of the respective noun phrase. Therefore it would be more systematic to first test whether certain noun phrases prefer collective and distributive readings and then classify the noun phrases as to their tendency to have wide or narrow relative scope. For example, in Ioup's empirical investigations this has not been considered. VanLehn (1978) however suggests a relation between the distributivity of a noun phrase and its tendency to have wide scope.

Studies on Quantifier Scope Disambiguation

Concerning the relative scope of quantifiers I want to summarize the main results of the studies of Ioup (1975), VanLehn (1978) and Kurtzman and MacDonald (1993). A more extensive summary can be found in Poesio (1994, pp. 18).

Many early proposals on scope disambiguation proposed something like a **linear order principle** according to which the left to right surface order of quantifiers determines their relative scope (Lakoff 1971). This principle was introduced to explain for example the wide scope preference of *every kid* in

(8) Every kid climbed a tree.

Ioup (1975). Experiments by Ioup (1975), however, showed that "… in natural language, order has little to do with the determination of quantifier scope" (Ioup 1975, p. 37). For example, in the preferred reading of

(9) I saw a picture of each child.

the rightmost quantifier *each child* has scope over the textually preceding quantifier *a picture* which contradicts the linear order principle. Ioup's findings suggest instead that quantifier scope is determined by the interaction of two factors: a **quantifier hierarchy** and a **hierarchy of grammatical functions**.

Furthermore, for her experiments Ioup establishes a five-valued scale of ambiguity judgements. The relative scopes of a quantifier pair Q1 and Q2 is then judged by one of the following categories.

(10) Scale of Ambiguity Judgments (Ioup 1975, p. 45)

unambiguous	wide scope Q1
ambiguous	wide scope Q1 preferred
ambiguous	no preference
ambiguous	wide scope Q2 preferred
unambiguous	wide scope Q2

The quantifier hierarchy determines the inherent tendency of quantifiers to take scope over other quantifiers. According to Ioup quantifiers such as *each* or *the* have the inherent lexical property of taking wide scope over indefinites, which are lexically marked to have scope over quantifiers like *all*. Ioup motivates this hypothesis with examples such as

(11) a. I saw a picture of each child.
b. I saw a picture of all the children.

(12) a. Ethel has a dress for every occasion.
b. Ethel has a dress for all occasions.

Her conclusion is that there seems to be a hierarchy of quantifiers that tend to have highest scope regardless of the environment. She suggests the following hierarchy whereby the quantifiers on the list are only the unstressed variants (see section 3.2.2.3 for the difference between stressed and unstressed variants of indefinites).

(13) **Quantifier Scope Hierarchy** (Ioup 1975, p. 42)
Greatest tendency towards highest scope
each
every
[*a, some* (+ NP_{sg})]
all
most
many
several

some (+ NP_{pl})
a few
Least tendency towards highest scope

Ioup has no clear evidence about the classification of *a* and singular *some*. She suggests that they are preceded only by *each* and *every*. The hierarchy is established by a pairwise comparison of the quantifiers in certain sentences.

For example, she supports her hierarchy with the following sentences:

(14) a. Joan gave a few handouts to several pedestrians.
b. Joan gave a few handouts to many pedestrians.
c. Joan gave a few handouts to every pedestrians.

The higher a determiner on the quantifier hierarchy the more likely it gets a wide scope distributive interpretation. Ioup says that this interpretation is evident for (14)c, plausible for (14)b and possible for (14)a. Again I want to criticize that the choice of the predicate *give to* makes a wide scope distributive reading of *a few handouts* difficult since the same things are usually not given away several times to different persons in a certain situation. The choice of the predicate *show to* instead of *give to* would perhaps be less biased in this respect.

Ioup does not consequently argue with the concepts of collective and distributive readings. She rather states that whenever an indefinite plural NP has wide scope it is interpreted distributively and if it has narrow scope it is interpreted collectively (Ioup 1975, p. 45). However, if in the example (14)a *a few* has wide scope I would prefer a cumulative reading, i.e. there are a few handouts and there are several pedestrians and each of the pedestrians receives one of the handouts. This is, however, not equivalent to a wide scope distributive reading of *a few*. In my setting this "cumulative" interpretation would be semantically represented by a doubly collective reading.

Again I find that my intuitions concerning the examples in (14) are rather unclear. As mentioned above it is difficult to separate the imagination of a possible or plausible *situation* from possible or plausible *scopings* of the quantifiers. Many of Ioup's other empirical studies are based on "easier" examples combining quantifiers like *a* and *every*, or *a* and *many* etc.

Ioup continues to argue that the inherent properties of quantifiers are not the sole determinants of relative scope. The grammatical function of the quantified NP has much to do with it. Again a hierarchy can be established according to which NPs in certain grammatical functions prefer wide scope over NPs in other grammatical functions. For example, NPs in subject position tend to have scope over NPs in indirect object position which tend to outscope NPs in direct object position etc. More concretely Ioup establishes the following hierarchy.

(15) **Grammatical Function Hierarchy** (Ioup 1975, p. 43)
Greatest tendency towards highest scope
deep and surface subject
deep subject/ surface subject (but not both)

indirect object
preposition object
direct object
Least tendency towards highest scope

A subject at deep *and* surface level is given in active sentences, e.g. in

(16) Every girl took a chemistry course.

the noun phrase *every kid* is both deep and surface subject. In contrast, in the passive transformation

(17) A chemistry course was taken by every girl.

the deep subject is *every girl* and the surface subject is *a chemistry course*.

The precedence of the subject over the direct object in the hierarchy of grammatical functions accounts for the preferred reading of (16), i.e. the wide scope reading of *every girl*. Since surface and deep subject are on the same level in (17) the preferred wide scope reading of *every girl* has to be explained by the quantifier scope hierarchy. This also explains the preferred wide scope reading of *every chemistry course* in

(18) Every chemistry course was taken by a girl.

Further examples compare other grammatical functions, e.g. the indirect object and the direct object. The sentence

(19) I told every child a story.

prefers a wide scope reading of the indirect object *every child*. In the following sentence the indirect object is filled by an indefinite noun phrase that nevertheless prefers a wide scope reading over the universally quantified direct object.

(20) I told every story to a child.

Further examples comparing preposition object and direct object are

(21) a. I had many conversations with a friend.
b. I had a conversation with many friends.

Ioup observes that in (21)a the prepositional object *a friend* prefers wide scope over the direct object *many conversations*, whereas in (21)b – where the plural noun phrase *many friends* is in the position of a preposition object – the sentence is judged ambiguous.

Ioup finally argues that an NP in topic position tends to take wide scope. This is especially clear in languages where – unlike in English – topic is explicitly marked. A weak topicality effect may nevertheless be found in English where the NP in subject position often plays the role of the sentence topic. Furthermore, special constructs such as left dislocation and fronting (e.g. *In this room, I feel really depressed.*) have the purpose of indicating the topic of a sentence. Ioup proposes to introduce a new grammatical function for topic and gives topic precedence over the subject in the grammatical function hierarchy.

VanLehn (1978). VanLehn (1978) looked at the correlation of quantifier scope judgments and syntactic structure. Several influences on quantifier scope were considered: the influence of the articles themselves, the positions of NPs in syntactic (surface) structure, the embedding of an NP within various structures, and the influence of left to right ordering with respect to NPs on the same level of embedding. In an empirical study he selected 121 sentences out of technical papers. Informants were asked to paraphrase the sentences and – if that did not indicate a clear preference – questions were asked to find out the preferred scoping. VanLehn did find evidence for a quantifier hierarchy although it is different from the hierarchy proposed by Ioup. VanLehn investigated sentences with an indefinite singular NP, and a universal quantifier or a plural NP. All informants preferred the reading where the indefinite took narrow scope when it was paired with the quantifier *each*; the preference became lower and lower with other quantifiers, and with *all*-NPs the preference was reversed.

(22) a. The club president splashed each member with a glass of champagne.
b. The club president splashed a glass of champagne over each member.
c. The club president splashed all the members with a glass of champagne.
d. The club president splashed a glass of champagne over all the members.
e. The club president splashed many of the members with several glasses of champagne.
f. The club president splashed several glasses of champagne over many of the members.

For sentence (22)a 80% of the informants reported a wide scope preference for *each*, in (22)b 90% preferred wide scope of *each*. In (22)c 70% preferred a wide scope reading of *a*, in (22)d 100% preferred a wide scope of *a*. In (22)e there was no clear preference: the sentence was – in my terminology – interpreted with a doubly collective reading such that there are many members and several glasses of champagne and the members are splashed with the champagne, but the sentence could also be interpreted with a wide scope distributive reading of *many*, meaning that each of the many members is splashed with several glasses of champaign. In contrast (22)f only got a wide scope collective reading of *several*. That means within the plural articles there is a further hierarchy. The higher an article is on the hierarchy the greater is the likelihood that the articles get a wide scope distributive reading with respect to articles lower on the hierarchy. VanLehn noted that, considering only NPs with universal, i.e. distributive force, the chance that a universally quantified NP would take wide scope was inversely correlated with the acceptability of that NP as the subject of collective predicates such as *meet*:

(23) a. *Each man met.
b. *?Every man met.
c. ??All of the men met.
d. All the men met.
e. The men met.

VanLehn (1978, p. 23) proposes to replace Ioup's quantifier hierarchy with a **distributivity hierarchy**. The higher a noun phrase in the hierarchy the more difficult it is to read the quantifier collectively and the likelier it is that determiner gets wide scope. VanLehn's hierarchy is as follows:

(24) Distributivity Hierarchy (VanLehn 1978, p. 23)
each > *every* > *all of the* > *all the* > other plural articles

Unfortunately, VanLehn proposes no clear hierarchy within the category "other plural articles".

According to VanLehn, quantifier scope is also dependent on whether a noun phrase is **specific or non-specific**. Specific NPs (definites are considered specific) tend to take wider scope than non-specific NPs (indefinites are often non-specific) as the following examples indicate:

(25) a. The club president splashed each member with a glass of champagne.
b. The club president splashed each member with the glass of champagne.

VanLehn also studied the availability of wide scope readings of NPs that are embedded in different structures. He noticed that there is a correlation between the type of the embedded structure and the scope of the NP.

(26) At the conference yesterday, I managed to talk to a guy
a. who is representing each raw rubber producer in Brazil
b. representing each raw rubber producer in Brazil.
c. from each raw rubber producer in Brazil.

VanLehn suggests the following **embedding hierarchy** for embedded distributives.

(27) Embedding Hierarchy (VanLehn 1978, p. 41)
determiner > PP > gerund > infinitive > finite clause

The higher the embedding structure lies on the hierarchy, the greater the tendency for the embedded quantifier to outscope the non-embedded quantifier and vice versa. However, there are certain asymmetries of specific and non-specific NPs (see VanLehn 1978, pp. 34) which make the hierarchy only partially valid. To simplify matters it turns out that the embedding hierarchy makes correct predictions for specific NPs (e.g. *each* is considered to be a specific NP) but is imprecise for non-specific NPs. That means that in (26)b and (26)c *each* is likely to outscope *a guy*.

Finally, VanLehn tested how the surface order of an NP influences its scope. He especially compared Ioup's hierarchy of grammatical functions with the linear order principle as he noted that the predictions differ only in a number of cases. He finds a comparable amount of counterexamples to both hierarchies and therefore suggests that surface order may be preferable because it is theory independent and easier to implement.

Kurtzman and MacDonald (1993). There is a more recent psycholinguistic study on the scope assignment process performed by Kurtzman and MacDonald (1993). In Poesio (1994, pp. 24) a good summary can be found. Kurtzman and MacDonald designed experiments to test the predictions of – among others – the linear order principle, Ioup's grammatical function hierarchy and the topic principle. They also check the influence of other factors like the choice of the predicate (e.g. "action", "perception", "stative" etc. predicates) and the difference between active and passive sentences. Unfortunately, they restrict their observations to examples with "simple" singular quantifiers such as *every* and *a*.

Kurtzman and MacDonald found a strong preference for the subject in active sentences to take wide scope. The preference is stronger if the subject is an indefinite NP and the object is a universally quantified NP (contrary to Ioup's predictions of the quantifier hierarchy).

Furthermore, Kurtzman and MacDonald showed that all of the suggested principles were subject to exceptions. This contradicts claims made in the literature that these principles definitely disambiguate sentences. Furthermore, Kurtzman and MacDonald found preferences not predicted by any of the principles discussed so far, e.g. there is a preference for the embedded NP in a complex nominal to take wide scope also if this nominal is not universally quantified as in

(28) George owns each picture of an admiral.

This is a further example that contradicts the linear order principle.

Furthermore, they find that agentivity affects the preference for subjects in active sentences to take wide scope. The wide scope reading of *every kid* is stronger in (29)a than in (29)b.

(29) a. Every kid climbed a tree.
b. Every kid saw a tree.

Finally, Kurtzman and MacDonald observed a clear difference between active and passive sentences.

(30) a. Every author wrote a book.
b. A book was written by every author.

In contrast to the active sentence, in the passive sentence there is no clear preference for one or the other scoping. This has also been observed by Ioup.

As a main result Kurtzman and MacDonald propose that in the process of scope disambiguation the discussed 'principles' are always active, but "they behave as *prioritized defaults* that may be overridden by stronger defaults or originate conflicts with defaults with the same strength" (Poesio 1994, p. 26).

In sum, Kurtzman and MacDonald's experiments show the preference for subject NPs to take wide scope in active sentences, the importance of agentivity and little evidence for the linear order principle. They offer no studies on collective/distributive ambiguities.

Studies on Collective/Distributive Ambiguities

The previous studies have mainly been concerned with scope disambiguation. I have discussed them at some length for two reasons: first, plural NPs enter scope relations, and in a practical application these scope relations have to be resolved, secondly, many of the studies discuss factors that I will utilize also for the collective/distributive disambiguation, e.g. the distributivity hierarchy or the grammatical function hierarchy.

I am aware of only one empirical study (Gil 1982) that is solely concerned with collective/distributive disambiguation, and – as indicated above – this study is very specific in its choice of examples, although it offers an interesting methodological discussion about the interpretation of empirical studies. Gil was mainly investigating sentences containing two numeric NPs. More concretely, he reports results for the examples

(31) a. Three boys saw two girls.
b. Two girls were seen by three boys.

He distinguishes *asymmetric* and *symmetric* readings. Asymmetric readings occur when the NPs have different scope, e.g. one NP is interpreted distributively and the other NP is in the scope of the first NP. Symmetric readings occur when both NPs refer independently. In my terminology symmetric readings are the doubly collective readings. Within the symmetric readings Gil distinguishes *weak symmetric* and *strong symmetric* readings. In Kempson and Cormack (1981) these are called the *incomplete* and the *complete group interpretations*. Strong symmetric interpretations occur when every element of the first group bears the appropriate relation to every element of the second group. Weak symmetric interpretations occur when every element of the first group bears the appropriate relation to at least one member of the second group, and, conversely, every member of the second group bears the appropriate relation to at least one member of the first group.

In his experiment, Gil presented the informants different states of affairs in diagrammatic form (analogous to Figure 1 on page 18 of this thesis). Each informant was asked to judge whether or not the appropriate sentence was true with respect to the presented state of affairs. Gil checked the sentences with 49 speakers of Dutch, 141 speakers of Hebrew and 29 speakers of Bengali.

The following are his main results:

(32) Results of Gil's empirical study (Gil 1982, pp. 423)
a. Sentences with two numerically-determined NPs can have four interpretations: two asymmetric interpretations (where one NP has wider scope than the other), and two symmetric interpretations – strong symmetric interpretations and weak symmetric interpretations.
b. Symmetric interpretations are preferred over asymmetric interpretations. Strong symmetric interpretations are the most highly preferred of the four classes of interpretations.

Gil also finds evidence that the preferences within the non-symmetric interpretations can be explained by Ioup's hierarchy of grammatical relations, that means a subject has wider scope than a direct object etc. And in passive sentences Gil observes that a prepositional (*by*-) phrase has wider scope than the derived subject

Gil raises a number of issues that are problematic for the interpretation of questionnaires of the type he used in general. First, it is not immediately apparent whether the informant's judgments are semantic or pragmatic in nature. Do the answers reflect the speakers' semantic judgments, i.e. judgements on possible or preferred truth-conditions, or are the answers pragmatic in nature, i.e. do they reflect whether a sentence is an appropriate or cooperative description of the presented state of affairs? Gil made a number of tests to assure that the judgements were indeed semantic in nature, with one exception. The interpretation of numerals as *exactly n* is performed according to pragmatic appropriateness conditions (see section 3.3.3.2).

According to my point of view, there are several problems with Gil's experiments. First, he does not vary the predicates in his experiment. He uses a stative perception verb (*see*), but does, for example, not use agentive verbs (e.g. *write*, *compose*, *lift*) in his sentences. Also he does not use predicates that can reasonably get a strict collective reading (e.g. *lift*). Also, he does not test the preferences in syntactic positions other than subject and object. Furthermore, he does not vary the numbers, e.g. giving the first NP a large number and giving the second NP a low number, and vice versa. Therefore, I doubt that Gil's results are general enough to count as an empirical basis. I agree that in many cases two numerals lead to symmetric interpretations, but I doubt that there is a significant general preference for strong symmetric interpretations. In my setting symmetric readings are represented by doubly collective readings. Whether a relation is strongly or weakly symmetric or is realized by any other constellation is not part of the semantic representation. The information can e.g. be inferred from the semantics of the verbs. What is more, Gil has not checked other types of plural quantifiers and their preferred readings.

5.2.2 Computational Approaches

5.2.2.1 Introduction

In the literature on representation and disambiguation of plurals in computational semantics we find mostly representational considerations that allow to integrate different types of disambiguating factors. However, implementing these factors has hardly be performed. I briefly summarize two approaches that make suggestions for the implementation of disambiguation factors: Schütze (1989) who builds upon Pafel (1988), and Aone (1991). I will only briefly consider several other approaches that touch the issue of disambiguation but are not worked out in enough detail.

There has been much more work on the implementation of *scope* disambiguation. Many of the above discussed disambiguation principles are indeed used in computational scope disambiguation (see Moran and Pereira 1992). The implementation of scope disambiguation is based on choosing a number of disambiguation factors, assigning them a relative weight and using them to score the readings compatible with a set of syntactic and semantic constraints. For example, first all readings consistent with the constraints are generated, then each of these possible readings is assigned a score depending on the number and the score of the disambiguation factors that are applicable, finally, the reading with the highest score is picked.

As I have discussed in section 2.3.7 this strategy is potentially subject to the problem of combinatorial explosion, which, however, in practice is often not a real problem, since only a few readings at a time are usually generated. As Poesio (1994) states the real problem of these approaches is “the lack of theoretical understanding of the scope disambiguation process: without such an understanding, the designers have no ways of choosing among the many existing principles, or of assigning them a relative weight, except than by trial and error” (Poesio 1994, p. 23). Thus it is not guaranteed that the principles and weights that were optimized for a certain domain or a certain set of data will generalize to other domains or to different types of data.

Although I am a aware of this criticism I will also choose a weighting algorithm for my computational approach to plural disambiguation. Moreover, I will borrow techniques developed for scope disambiguation.

5.2.2.2 Scope and Plural Disambiguation

Pafel (1988)

Within the LILOG project (see Herzog and Rollinger 1991 for the final report) Pafel (1988) develops a system to determine the relative quantifier scopes of German sentences. He builds upon Ioup’s (1975) findings on the grammatical function hierarchy and – similar to VanLehn (1978) – reinterprets her quantifier hierarchy as a distributivity hierarchy. Additionally, Pafel supports that – in German – the linear order of quantifiers influences their scoping, and – what has not been investigated in Ioup – he proposes that the syntactic structure of the quantified noun phrase plays a role. In particular it is important whether the restriction of the determiner contains another full noun phrase or just a noun.

In Pafel’s proposal the factors **grammatical function**, **distributivity** and **syntactic structure** enter the determination of the relative scope with equal weight, whereas **linear order** is assigned double weight. To calculate the relative scope of quantifiers Pafel uses four scales with numeric values. The higher the numeric value the higher the tendency to receive wide scope. Note that concerning the linear order in German sentences Pafel distinguishes the “Vorfeld” position. The “Vorfeld” in German sentences is a distinguished position and means the constituent before the finite verb. I will ignore this component for the investigation of English sentences. Following are Pafel’s scales with associated numeric values:

(33) Scale of linear order
5 Preceding in Non-“Vorfeld” position
3 Preceding in “Vorfeld” position
0 Non-Preceding

(34) Scale of grammatical functions
5 Subject
4 Indirect Object
3 Preposition Object
2 Direct Object

(35) Scale of inherent distributivity

5	inherently distributive	(e.g. *jed-*, *beid-*	= *every*, *both*)
4	dominant distributive	(e.g. *all-*, *d- meist-*, *viel-*	= *all*, *most*, *many*)
3	dominant non-distributive	(e.g. *einig-*, *numerals*	= *some*, *two*)
2	inherently non-distributive	(e.g. *ein-*, *the* (sg)	= *a*, *the*)

(36) Scale of syntactic structure
5 Restrictor contains a maximal NP
3 Restrictor does not contain a maximal NP

The relative scopes in a sentence containing more than two noun phrases can be determined by a pairwise comparison of the noun phrases. The comparison is based upon a combination of factors as follows: First, the numeric values of the respective noun phrases within the above scales are determined. To determine the linear order value the preceding NP_1 gets the value 5 if it occurs in a non-“Vorfeld” position and precedes another NP_2, or the value 3 if NP_1 occurs in “Vorfeld” position and precedes another NP_2. The succeeding noun phrase gets the linear value 0. The determination of the distributivity value is lexically given, the grammatical function and the syntactic structure have to be derived from the syntactic analysis of the sentence. Thus for each of the two noun phrases that are compared you receive four values: the value for the linear order, the value for the grammatical function, the value for the distributivity and the value for the syntactic structure. The *scope value* of an NP_1 with respect to another NP_2 is then is calculated as follows:

(37) Scope Value: =
2 × linear order + grammatical function + distributivity + syntactic structure

Note, that the scope value of a noun phrase is a relative value since it depends on the other noun phrases with respect to which the linear order has been determined. Assume two NPs, NP_1 and NP_2, are compared. Their scope values are denoted by $\texttt{scopevalue}(NP_1)$ and $\texttt{scopevalue}(NP_2)$. Then Pafel (1988, pp. 28) determines their relative scopes as follows:

(38) Relative Scopes of German Quantifiers
1. $\texttt{scopevalue}(NP_1) = \texttt{scopevalue}(NP_2)$
Both readings possible, no preferences.
2a $|\texttt{scopevalue}(NP_1) - \texttt{scopevalue}(NP_2)| < 5$
The sentence is ambiguous, the wide scope reading of the NP with the higher

scope-value is preferred.

2b $|\texttt{scopevalue(NP}_1\texttt{)} - \texttt{scopevalue(NP}_2\texttt{)}| = 5$ and the higher NP has maximally the value 15: see 2a.

3a $|\texttt{scopevalue(NP}_1\texttt{)} - \texttt{scopevalue(NP}_2\texttt{)}| > 5$

The NP with the higher scope value has wide scope, no other readings.

3b $|\texttt{scopevalue(NP}_1\texttt{)} - \texttt{scopevalue(NP}_2\texttt{)}| = 5$ and the higher NP has a value greater 15: see 3a.

Pafel gives several examples in German to support his rules. For the English examples I will ignore the extra points given for precedence in non-"Vorfeld" position. For example in

(39) Each of the men lifted one of the tables.

the NPs are assigned the following values:

(40)

	each of the men	one of the tables
linear order	$2 \times 3 = 6$	0
function	5	2
distributivity	5	1
syntactic structure	5	5
scope value	**21**	**8**

Since the difference between the scope values is greater than 5 the wide scope reading of *each of the men* is the only possible reading. The sentence

(41) A man loves every woman.

triggers the following scope values:

(42)

	a man	every woman
linear order	$2 \times 3 = 6$	0
function	5	2
distributivity	1	5
syntactic structure	3	3
scope value	**15**	**10**

Since the scope difference equals 5 and the higher NP has maximally the value 15 the sentence is ambiguous, however, the wide scope reading of *a man* is preferred. Pafel also discusses examples with plural determiners. In

(43) Most men like a woman.

the NP *most men* has the scope value 17 ($= 2 \times 3 + 5 + 3 + 3$) and *a woman* the scope value 6 ($= 0 + 2 + 1 + 3$), which means that the sentence is predicted to have just the wide scope (distributive) reading of *most men*. The following sentence contains three NPs and induces the following relative scope values.

(44) Every man told many children several stories.

every/many	19:10
every/several	19:7
many/several	16:7

As a result the sentence is assigned just one reading with the quantifier order *every/many/several*. Whether *many children* is interpreted collectively or distributively is not resolved by this algorithm. Sentence (45) with direct and indirect object reversed triggers the following scope values:

(45) Every man told several stories to many children.

every/several	19:13
every/many	19:10
several/many	13:10

This means that the sentence is ambiguous between the quantifier ordering *every/several/many* and the ordering *every/many/several* where the first reading is slightly preferred. Again nothing is said about collective-distributive ambiguities. In the sentence

(46) Every man told a story to many children.

every/several	19:6
every/many	19:10
a/many	9:10

the algorithm predicts a preferred reading where *many* has scope over *a*.

Pafel's method can be used to find out possible and plausible quantifier scopings in a sentence. His algorithm has been fine-tuned and tested for German sentences; English examples were not explicitly investigated. In general, Pafel shows how computationally available disambiguation factors can be integrated into a real system. In his report he gives a brief but very useful documentation of his implementation. His method, however, does not attempt to give theoretical explanations of the preferences. The weighting is fine-tuned for his small set of examples but not linguistically investigated or motivated. Still I think that Pafel's method is a good starting point for the implementation of scope disambiguation in practical systems. Though he does not elaborate on collective/distributive ambiguities the basic idea of his method will also be useful for my proposal of a practical disambiguation of plural sentences.

Other Computational Scoping Algorithms

In Allen (1995, pp. 349) a further scoping algorithm for practical applications is discussed. Furthermore the article "Quantifier Scoping" by Moran and Pereira (1992) discusses quantifier scoping in the "Core Language Engine" (Alshawi 1992). Moran and Pereira suggest further constraints and preferences that cover a wider range of syntactic constructions. Their implementation is also based on a pairwise comparison of all quantifiers in a sentence, and on giving the rules and preferences certain weights. Moran and Pereira do not discuss the influence of plural noun phrases.

Schütze (1989)

In his diploma thesis Schütze (1989) builds upon Pafel's results and uses it for a more elaborate system on plural disambiguation. He reconsiders Pafel's scale of distributivity by a close investigation of a number of German quantifiers. Schütze's investigations suggest that distributivity of determiners is the primary phenomenon and that the tendency of these determiners to take wide or narrow scope is only derived from the internal distributivity. To determine the internal distributivity Schütze tests plural determiners in different contexts. First, in contexts where the collective reading is enforced by additional modifiers:

(47) DET strong men together carried the piano into the second floor.

Second, in contexts with collective verbs:

(48) DET Swiss students scattered after a demonstration.

Third, in contexts where both readings are – in principle – possible:

(49) DET strong men lifted the table.

Schütze finds, that some determiners easily allow for collective readings in the sentences (47) and (48), e.g. the German equivalents of *two*, *several*, *a few*, *some*. Other determiners like *at most three*, *all* are easily read collectively in (47) but more difficult to read collectively in (48). In (47) and (48) determiners like *most*, *no* tend to induce a partitioning of a set of men into groups of men that together carry the piano. The third sentence (49) distinguishes determiners in a neutral context as to their tendency to be interpreted collectively or distributively. Schütze uses the following abbreviations in his resulting classification of determiners:

(50) Abbreviations

c	collective reading
d	distributive reading
>>	strongly prevails
>	prevails

That means the category 'c >> d' expresses that the respective determiner strongly prefers a collective reading. Schütze suggests the following classification categories with respective numeric values similar to Pafel:

(51) Schütze's Distributivity Scale

d	5	*every*, *both*, *no*
d >> c	4	*many*, *few*, *numerous*
d > c	3	*exactly n*, *at most n*, *at least n*, *all*, *all the*
c > d	2	*two*, *three*, …, *some*, *several*, *a few*, *the* personal pronouns (*they*) and possessive pronouns (*their*), possessives (*John's books*)
c >> d	1	null quantifier (e.g. in the noun phrase *men*)

Schütze then checks whether the distributivity scale correctly predicts the tendency of the determiners to take wide scope. Concerning the influence on scope preference Schütze finds no evidence for the distinction between the categories ‘d >> c’ and ‘d’. Therefore he suggests the following scale concerning the scopal strength of determiners:

(52) Schütze’s Scope Values

5	*every*, *both*, *no*, *many*, *few*, *numerous*
3	*exactly n*, *at most n*, *at least n*, *all*, *all the*
2	*two*, *three*, …, *some*, *several*, *a few*, *the*
1	null quantifier, singular *a, one*

Schütze does not unify the two scales since he is not only concerned with scope ambiguities but also with collective/distributive ambiguities. The distributivity scale is needed to determine collective/distributive preferences. Since e.g. *many* and *few* can marginally get a distributive reading they have to be distinguished from *every* or *no*. This distinction has been abandoned in the scoping scale (52) whereas it is maintained in the distributivity scale (51).

Schütze also investigates the influence of other elements of the sentence to the interpretation of plurals. He distinguishes three types of verbs, collective verbs which trigger a collective reading, distributive verbs that trigger a distributive reading and mixed verbs that can have both readings. He suggests that collective verbs trigger a collective reading if they are combined with noun phrases that allow for a collective reading (i.e. noun phrases of at least the type ‘d >> c’). Combinations with singular non-collective noun phrases (like *a man*) are not grammatical, and combinations with distributive plural noun phrases lead to a partitional interpretation. Combined with distributive verbs all noun phrases are interpreted distributively, and combined with mixed predicates noun phrases of type ‘c >> d’ and of type ‘c > d’ get a *default* collective reading, noun phrases of type ‘d > c’ and ‘d >> c’ get a *default* distributive reading, and distributive noun phrases get only a distributive reading.

(53)

NP type	VP type	Reading	Example
c >> d, c > d	collective	c	the men meet
	mixed	default: c	the men lift a table
	distributive	d	the men sleep
d >> c, d > c	collective	c	all/many men gather
	mixed	default:d	all/many men lift a table
	distributive	d	many men sleep
d	collective	(Reinterpretation)	?both men gather
	mixed	d	both men lift a table
	distributive	d	both men sleep

The problem for computational applications is how to get the respective classification of verbs. I will come back to this problem below in section 5.3.2.

Like other authors, Schütze also points out that collective/distributive ambiguities not only occur in combination with noun phrases and verbs but also within noun phrases, e.g. with adjectives, with *of*-genitives and with other prepositional phrases:

(54) the expensive telephones
(55) the papers of the students
(56) the tarpaulins on the pallets

(54) can mean that each telephone is expensive, or that the telephones together are expensive. (55) can denote for each student his or her papers or it can denote the sum of the papers of the students together. (56) can mean that each pallet is protected by its own tarpaulin, or in the collective reading that the tarpaulins together protect the pallets as a whole.

Schütze treats the distribution of adjectives to nouns analogous to the distribution of verb phrases to noun phrases. He assumes that there are three types of adjectives: collective (c), distributive (d) and mixed (m) adjectives. Simple fully collective adjectives are not so frequent, examples could be *gathering*, *assembling*. Furthermore, "reciprocal" adjectives could be seen as collective adjectives, e.g. *parallel*, *successive*. Mixed adjectives are typically measure adjectives like *heavy*, *long*, *expensive*. Most adjectives are distributive and are – like non-collective count nouns – true of individuals, e.g. *tired*, *red*. Schütze suggests default interpretations if plural noun phrases contain adjectives. These defaults are slightly different from those for verbs in that distributivity starts earlier.

(57)	NP type	Adjective type	Default Reading	Example
	c >> d, c > d	collective	c	the gathering men
		mixed	default: d	the heavy men
		distributive	d	the tired men
	d >> c, d > c	collective	partitioning	few gathering men
		mixed	d	few heavy men
		distributive	d	few tired men

There are no explicit rules for other NP internal ambiguities as in (55) and (56). Schütze points out that very often – if there is a distributive interpretation – the head noun phrase is distributed to the modifying noun phrases, e.g. in (55) papers are distributed to students (not vice versa) and in (56) tarpaulins are distributed to the pallets.

Like Pafel, Schütze works with German examples. So, strictly speaking, his findings are only valid for German. Still, Schütze presents a well worked out classification of some factors that influence the interpretation of plural noun phrases. He establishes a number of relevant features that interact to determine a default interpretation. Furthermore, Schütze clearly distinguishes the problem of scope from the problem of plural ambiguities. However, he does not make clear how exactly plural disambiguation and scope disambiguation interact in a practical applications. Furthermore, Schütze faces the problem how the additional information (e.g. whether verbs are collective or distributive) are automatically collected. Still, Schütze's approach is a computationally manageable approach since it does not rely on external information sources like domain knowledge. This is different in Aone's approach that I will discuss next.

Aone (1991)

General

Whereas the above disambiguation algorithms were mainly based on structurally available knowledge Aone (1991) proposes a plural disambiguation algorithm which is mainly based on the formalization of external knowledge sources. Aone develops a reasoning module to disambiguate collective/distributive ambiguity involving two numerically specified NPs. His prototypical examples are

(58) Five diskservers are used by four clients.
(59) Five students ate four slices of pizza.
(60) Five wilders attacked four joggers.
(61) 5 dogs had (a litter of) 4 puppies.
(62) 500 children shared 400 lbs. of cookies.
(63) Five alarms were installed in 6 buildings.
(64) Five piglets were born to 6 pigs last month.

Aone proposes a special representation language **CDCL** (Collective-Distributive Constraint Language) that is used in the reasoning process. This representation is computationally suitable as input for the reasoner. Aone's algorithm follows the strategy "generate and test". English input is parsed by the parser, and the result is mapped to DRSs (with events) by the semantics module. The semantics module can generate all the possible readings of an input sentence in DRS forms, and thus is a **hypothesis generator**. These readings can be nonsense, because they are generated without any domain knowledge. If a sentence is ambiguous, each of its possible DRSs is then translated into an CDCL representation in order for the reasoner to work on. If the reading is not consistent with the knowledge of the domain which the reasoner can utilize, the reading is rejected (**hypothesis testing**). The reasoner is thus acting as a **hypothesis filter**. Only consistent readings are further considered and stored in a knowledge base.

How does the filtering work? Aone formulates two main sources for disambiguation:

- domain-specific knowledge formulated as **constraints** in CDCL,
- domain-independent knowledge formulated as **axioms** in CDCL.

Constraints

Each predicate (e.g. `is_installed_in`, `is_born_to`, `share`) which is defined in the knowledge base has its associated constraints. There are two types of constraints, **type constraints** and **numerical constraints**.

Type Constraints. Type constraints put restrictions on types of arguments of predicates (i.e. constraints on whether the arguments should be read collectively or distributively). The following sentence illustrates a **type constraint**:

(65) The children shared 400 pounds of cookies.

The verb *share* requires both of its arguments to be read collectively. A constraint is formulated in CDCL that both arguments of *share* are to be read collectively. The form of these constraints and a complete list is introduced in Aone (1991, pp. 91). His constraint C2 – expressing that both arguments are collective – gives an impression of how constraints look like in CDCL:

(66) C2 (1 (?p:set ?a:arg → ?q:set ?b:arg)) :⇒ inconsistent

Predicate constraints are represented as rules for reasoning. The constraint above is to be read as "anti-rule". That is, if a reading does not meet a constraint in the antecedent, the reading is considered inconsistent.

Aone divides type constraints into constraints that hold for the predicates in all domains, and constraints that are specifically formulated for a certain domain.

Constraints that hold for the predicates in all domains:

- C1: Both arguments are distributive. (`disk-used-by a0 a1`, `eat a0 a1`)
- C2: Both arguments are collective (`share a0 a1`)
- C3+C4: The first argument is distributive and the second collective. (e.g. `deliver-off-spring a0 a1`)
- C5+C6: The first argument is collective and the second distributive (e.g. `attack a0 a1`).

Domain specific type constraints:

- C7+8: Each client has at most one diskserver connected to it, which means that the predicate `disk-used-by a0 a1` has to obey the following constraint
  (?p:num (1 a1 → ?q:num a0)) ⇒ (<= ?q:num 1)
  The constraint expresses that a0 and a1 are to be read distributively and that the number of a1 (i.e. the number of disks used by a client) is smaller or equal to 1

Domain specific type constraints can be read as ordinary rules: If they succeed the consequents are asserted, if they fail, nothing is done.

Numerical Constraints. Numerical constraints restrict numerical relations between arguments of predicates (e.g. a relation from argument 1 to argument 2 is a function).

Examples for numerical constraints are:

- C9: A relation from a0 to a1 is a function (e.g. `installed-in a0 a1`).
- C10: A relation from a1 to a0 is a function (e.g. `eat a0 a1`).
- C11: A relation from a0 to a1 is a function whose domain is a set of sets (e.g. `born-to a0 a1`)
- C12: A relation from a1 to a0 is a function whose domain is a set of sets (e.g. `deliver-offspring a1 a0`).

Axioms

Axioms can be considered as a special kind of constraints; predicate constraints hold only for particular predicates, axioms hold regardless of concrete predicates. Therefore axioms are **domain independent** knowledge. Aone distinguishes two kinds of axioms, **constraint axioms** and **reading axioms**.

Constraint axioms. Constraint axioms are constraints about predicate constraints. Constraint axioms derive more constraints if certain constraints are associated with the predicates.

An example is constraint axiom CA1 (Aone 1991, pp. 94) which is concerned with the number of objects in the range of a relation denoted by a predicate. The constraint says that if a relation is a function from A to B where the domain is not a set of sets, the number of the objects in the range is less than or equal to the number of objects in the domain.

(67) Five students ate four slices of pizza.
Predicate: (eat a0 a1)
Predicate constraint: C10 (a relation from a1 to a0 is a function).

With the help of the constraint axiom CA1 we can derive the additional constraint that the number of pizzas has to be smaller than the number of students. If at any point of the reasoning process this constraint is not true the reading that lead to the contradiction has to be marked as inconsistent.

Reading axioms. Reading axioms are axioms about certain assertions that represent particular readings of a sentence. Reading axioms derive more assertions from existing assertions, e.g. assert the number of all objects in the domain of a relation given an assertion e.g. of the form where first argument is distributive:

(68) Five students ate three apples.

Given the reading where both arguments are distributive and *five students* has wide scope, the reading axiom RA1 is used to derive that the number of students in the domain of the relation *eat* is 5.

In Aone's system reading axioms are also necessary to reason about the consistency of cumulative readings when numerical constraints are associated with the predicates in the readings, e.g. *eat* is associated with the constraint that one object can only be eaten by one person. That means if the number of apples is smaller than the number of students the cumulative reading is eliminated. A problem with this example is that the plausibility of cumulative readings sometimes depends on the divisibility of objects denoted by nouns. For example, in

(69) Five students ate four slices of pizza

it is possible that there are less slices of pizzas than there are students. This example already shows that the precise formalization of constraints for predicates is very difficult since many other knowledge sources influence the plausibility of a certain reading. And this knowledge would have to be formalized as well.

Aone compares his axioms with the human reasoning process when sentences are disambiguated. Constraint axioms simulate how humans make comparisons between two groups of objects related by a predicate that has a numerical constraint. Reading axioms simulate how we do such a calculation given a certain reading and its associated constraints.

Reasoning

Aone's reasoner is based on the theory of model-based reasoning. The semantics module is a hypothesis generator for collective and distributive readings of a given sentence. What reasoning does is hypothesis filtering given a model. The model consists of four kinds of knowledge sources:

- predicate constraints
- constraint axioms
- reading axioms
- simple mathematical knowledge (e.g. that 5 is smaller than 7)

If a hypothesis does not satisfy a constraint in the model, it is rejected as an inconsistent reading. Each reading is stored as a hypothesis in a reading record. At the beginning the reading record just stores assertions that represent the reading, and records that it is a consistent reading by default. As reasoning proceeds and more information is asserted using the model, the reading record is updated. When inconsistency arises, the reading record is marked as inconsistent and the hypothesis is filtered out. In his dissertation, Aone shows how his reasoning module derives the most plausible readings for the following sentences. The notation 'Ca0 Da1' means that the argument a0 is collective and that the argument a1 is distributive, furthermore that the first argument a0 has wide scope.

(70) Five diskservers are used by four clients.(Da0-Da1)
(= Each of the five diskservers is used by 4 persons one at a time.)

(71) Five students ate four slices of pizza.(Da0-Da1)
(= Each of the five students ate 4 slices of pizza one at a time.)

(72) Five wilders attacked four joggers.(Ca0-Da1, or cumulative, or Da1-Ca0)
(Ca0-Da1 means: Each of the four joggers were attacked, on different occasions, by the same group of four wilders).

(73) 5 dogs had (a litter of) 4 puppies. (Da0-Ca1)
(= Each of the five mother dogs delivered a litter of 4 puppies.)

(74) 500 children shared 400 lbs. of cookies.(Ca0-Ca1)
(= There were 500 children and 400 pounds of cookies, and the children shared the cookies in some way.)

(75) Five alarms were installed in 6 buildings.(Da1-Da0)
(= Each of the 6 buildings are installed with 5 alarms.)

(76) Five piglets were born to 6 pigs last month. (Da1-Ca0)
(= Each of the 6 pigs had 5 piglets last month.)

Aone does not consider syntactic factors that lead to the plausibility of certain readings.

Aone's approach heavily relies on the formalization of domain knowledge and of lexical knowledge. For computational applications this precision is only possible for small restricted domains but the approach is not feasible for large domains.

Other Approaches

Many computational approaches to plural semantics integrate a very simple disambiguation mechanism that I do not want to comment in detail. For example, Graham (1994) triggers distributive readings only if an explicit disambiguation marker like *each* is put after the noun phrase. In an approach by Scha and Stallard (1988) "distributive" (*walk*) and "collective" (*gather*, *disperse*, *meet*) verbs are distinguished and meaning postulates are formulated for them. There are, however, several problems with meaning postulates since the variability in distribution depends on several factors. First, world knowledge influences the distribution patterns. Meaning postulates dictate the same distribution pattern for any verb, yet it does not seem plausible that one could finally decide what this should be, since the beliefs and knowledge about the world from which the postulates are derived are subject to variation from speaker to speaker. Second, context imposes a variability of distribution.

(77) The squares contain the circles.
(78) The children ate the pizzas.

This variability contradicts the idea of meaning postulates which are stipulated to be true in all models. That means it is logically incoherent to have several, mutually incompatible meaning postulates for the same verb. Alternatively, meaning postulates could be seen like conventional implicatures, which are "usually" or "often" true. The problem is, however, that the semantics should state specific truth conditions. Scha and Stallard's solution is that they propose an open-ended ambiguity of lexical items. This ambiguity is not one of syntactic type, that means it makes no sense to multiply lexical entries. Instead they introduce a level of representation in which these distributional issues are left open to be resolved by a later stage of processing. That means two levels of semantic interpretation are introduced. One "ambiguous" level EFL ("English-Oriented Formal Language") with expressions viewed as schemata abbreviating sets of possible instance-expressions and a second level WML ("World Model Language") which is unambiguous. A set of translation rules relates each ambiguous constant of EFL to a set of WML expressions representing its possible meanings. EFL and WML are instantiations of a higher order logic with a recursive type system. Each expression has a type that is computed from the type of its subexpressions. Expressions that are constructed from subexpressions of inappropriate type are not meaningful and are called "semantically anomalous", they get a special type NULL-SET.

Concerning disambiguation, Scha and Stallard's concern about disambiguation is limited to type checking, i.e. their representation language employs a recursive types system and arguments of verbs have type restrictions. For example, the subject of a predicate *eat* is of individual type, and therefore when it combines with an NP of set type, or set of sets type, etc., the expression gets rejected by the system. This is similar to what Aone's reasoner does using typing constraints. Beyond type checking, however, Scha and Stallard's system does not offer any reasoning capability. In a way, Scha and Stallard offer with their EFL an early form of an underspecified representation.

Other systems that contain a computational plural processing component are more concerned with representational issues. Covington (1996) offers suitable plural representations for model-based question answering based on Prolog. The DORIS system (Bos 2001) parses a limited set of plural construction but does not distinguish collective and distributive readings systematically. The Core Language English (Alshawi 1992) offers underspecified representations for collective and distributive readings but does not deal with their automatic disambiguation. Reyle (1994, 1995) proposes underspecified DRSs for plurals and suggests possible rules for disambiguation. But his remarks remain on a rather theoretical and representational level.

5.2.2.3 Evaluation of Computational Approaches

In the following, I will briefly evaluate and compare the approaches of Schütze and Aone with respect to plural disambiguation. I will focus on the computational aspect of their suggestions.

Both, Schütze and Aone choose a DRS based semantic language for the representation of plurals. Schütze's DRS language is more expressive in that he also considers non-standard quantifiers and discusses their relation to first-order logic.

Schütze offers a simple but computationally manageable approach to plural disambiguation. The approach is based on lexically and structurally available knowledge encoded as features. Schütze's algorithm predicts default readings that can – if contradicting knowledge is available – be overridden. How this is concretely realized is not discussed in Schütze's thesis. Schütze's algorithm can be combined with Pafel's scoping algorithm. Both algorithms have a computationally manageable foundation in that only structurally and lexically available knowledge is used for the disambiguation. Unlike Aone, Schütze does not make detailed predictions about possible distribution schemata. Schütze's feature-based approach is suitable to be extended by other disambiguation factors.

Aone's algorithm is based on a detailed encoding of lexical knowledge of verbs and of domain knowledge. His approach predicts possible distribution schemata for a number of sentences containing two numerically specified noun phrases. Aone does not consider the disambiguation of other plural constructions. His approach is problematic in that both, manual or automatic encoding of the precise constraints is problematic if not impossible for large scale applications. Furthermore, I doubt that verbs in general can be clearly classified according to the constraints Aone suggests. The eating example (59) shows that not only the semantics of verbs influences possible and plausible distribution schemata but also other factors like our knowledge about the participating objects.

Furthermore, I am not sure how universal the character of other constraints in Aone's system is. For example, `installed-in` has a constraint of a physical nature: one and the same object cannot be installed in more than one place at the same time, for example this results in a many-to-one function from alarms to buildings in (75). However, what are the identity criteria for objects. How about the operating system OS X being installed on different computers at the same time?

A related problem of Aone's approach is that it is centred around the predicate. There are, however, many semantically neutral predicates that do not indicate possible or plausible constellations. Furthermore Aone does not integrate other disambiguation factors, e.g. the influence of determiners or the influence of the syntactic structure and surface structure. The extension of Aone's suggestion with these additional factors is not straightforward.

In Schütze's approach disambiguation is triggered at a very early stage of text processing. Many disambiguation factors already influence the semantic construction. In contrast, Aone's approach requires a – possibly costly – postprocessing of semantic representations within his reasoning component. This also involves that Aone first has to generate all possible readings and has to filter them later ("generate and test") which may lead to a combinatorial explosion of readings. Schütze, in a first step, only generates the plausible default reading that is consistent with his feature combinations.

Schütze's approach always predicts a default reading, whereas in Aone's approach, filtering does not necessarily lead to a single meaning. It is not clear which reading is chosen by Aone's algorithm if an automatic choice has to be performed in computational applications.

Concerning the reasoning component itself, Aone's approach is more elaborate than Schütze's. It is important to have an additional reasoning component that can manage additional knowledge if it is available (e.g. for a restricted domain). In Schütze's approach no such reasoning component is explained which is why he cannot make predictions that go beyond assigning collective and distributive readings. Aone's reasoning component relies on a non-standard language to formalize the additional knowledge and it uses a non-standard reasoner especially developed for the purpose of plural disambiguation. In my approach I will prefer the encoding of additional knowledge in standard-first order logic which allows me to rely on standard first-order theorem provers that can be reused in different applications.

Crucial constructions that are very hard to disambiguate for both, Schütze's and Aone's approach are for example

(79) 33 signals of 4 types
(80) 205 miners of two mines
(81) four packs of eight cells
(82) Tissue samples were taken from 10 defined points of 6 organs of 96 deceased people.
(83) Five organs of three newborns and one infant.

In general, there is no satisfactory fully worked out plural disambiguation algorithm that is based on a stable set of empirical data.

5.3 Accessible Disambiguation Information

5.3.1 Overview

In the following sections I will develop a plural disambiguation algorithm that is based on computationally accessible information sources. The algorithm is prototypically implemented within the system DRoPs (Disambiguating and Reasoning with Plurals) that I have developed to practically test the proposals of this thesis. The main question will be whether a feasible *structural* solution to plural disambiguation can be found, i.e. how much disambiguation information can be extracted from the syntactic structure. Furthermore, predefined lexical knowledge that can be extracted from machine readable lexica is used. Additional knowledge like world-knowledge or contextual knowledge can be added via first-order axioms (see chapter 6) but is not prerequisite for the disambiguation algorithm to work. In section 5.3.2 I will classify the accessible disambiguation information that I am using in my algorithm. Many of the disambiguation sources have been used in various approaches that I have discussed in section 5.2 above. I will therefore not comment them again in detail.

In my implementation of the algorithm the accessible disambiguation information is encoded as feature structures within a unification-based phrase structure grammar. The grammar is written in ProFIT (Prolog with Features, Inheritance and Templates) (Erbach 1994, Erbach 1995). ProFIT is an extension of Prolog which allows to declare an inheritance hierarchy, features and templates. Sorted feature terms can be used in ProFIT programs together with Prolog terms to provide a clearer description language for linguistic structures. Among other linguistic information the grammar encodes the information entering the disambiguation process as feature structures. In Table 8 on page 237 these feature structures are summarized. The features will be commented in the following sections.

5.3.2 Types of Disambiguation Information

My disambiguation algorithm contains several types of disambiguation information:

- linguistic information
  - explicit disambiguation triggers (e.g. floated quantifiers)
  - lexical information (e.g. lexical information about quantifiers, verbs, adjectives)
  - structural information (e.g. linear order, grammatical function, syntactic structure)
- non-linguistic information
  - world- and domain knowledge
  - pragmatic knowledge

Linguistic disambiguation information is derived from two sources: from the structure of a sentence or from our lexical knowledge about the semantics of words. The structure of a sentence shows for example the linear order of noun phrases, the grammatical function of noun phrases (subject, direct object etc.), the syntactic structure of noun phrases etc. Lexical infor-

<table>
<tr><td>syn</td><td>index</td><td colspan="4">Variable</td></tr>
<tr><td></td><td>v_dist_type</td><td>arg1</td><td colspan="3">d | c | m</td></tr>
<tr><td></td><td></td><td>arg2</td><td colspan="3">d | c | m</td></tr>
<tr><td></td><td></td><td>arg3</td><td colspan="3">d | c | m</td></tr>
<tr><td>sem</td><td>object_index</td><td colspan="4">Variable</td></tr>
<tr><td></td><td>event_index</td><td colspan="4">Variable</td></tr>
<tr><td></td><td>log_rel</td><td colspan="4">(determined by lexicon, e.g. man | card | …)</td></tr>
<tr><td></td><td>object_type</td><td colspan="4">person | time | object</td></tr>
<tr><td></td><td>v_info</td><td>default</td><td>dist_type</td><td colspan="2">d | c | m | ‘d > c’ | ‘c > d’</td></tr>
<tr><td></td><td></td><td>c_class</td><td colspan="3">vollective | intrans_recip | …</td></tr>
<tr><td></td><td></td><td>eventl</td><td colspan="3">state | event</td></tr>
<tr><td></td><td></td><td>v_class</td><td colspan="3">measurment | perception | motion | ingesting | body | …</td></tr>
<tr><td></td><td>n_info</td><td>collective</td><td colspan="3">yes | no</td></tr>
<tr><td></td><td>quant</td><td>quant_type</td><td colspan="3">pn | def | indef | card | quant | no</td></tr>
<tr><td></td><td></td><td>monotone</td><td colspan="3">up | non-up | right-up</td></tr>
<tr><td></td><td></td><td>quantity</td><td>count_unit</td><td colspan="2">count_unit | kg | m | l |</td></tr>
<tr><td></td><td></td><td></td><td>dimension</td><td colspan="2">cardinality | weight | length | volume | …</td></tr>
<tr><td></td><td></td><td></td><td>value</td><td colspan="2">0 | 1 | 2 | 3 | …</td></tr>
<tr><td></td><td></td><td></td><td>num_rel</td><td colspan="2">eq | leq | geq | less | greater | …</td></tr>
<tr><td></td><td></td><td></td><td>vague_spec</td><td colspan="2">most | most_of | many | many_of | few | few_of |
more_than_half_of | no</td></tr>
<tr><td></td><td></td><td>disambig</td><td>default</td><td>dist_type</td><td>'c>>d' | 'c>d' | 'd>>c' | 'd>c' | 'd'</td></tr>
<tr><td></td><td></td><td></td><td>reading</td><td>result</td><td>distr | coll</td></tr>
<tr><td></td><td></td><td></td><td></td><td>stability</td><td>pref | unpref | fix</td></tr>
<tr><td></td><td></td><td></td><td>scop_lex</td><td colspan="2">1 | 2 | 3 | 5 | …</td></tr>
<tr><td></td><td></td><td></td><td>syn_struct</td><td colspan="2">noun | partitve | pp | poss | rel_clause</td></tr>
<tr><td></td><td></td><td></td><td>gra_fct</td><td colspan="2">subj | do | io | po</td></tr>
<tr><td></td><td></td><td></td><td>subcat</td><td colspan="2">iv | tv | dv | prep | adj</td></tr>
<tr><td></td><td></td><td></td><td>marker</td><td>type</td><td>disambiguation | elaboration</td></tr>
<tr><td></td><td></td><td></td><td></td><td>lex</td><td>together | each | at_the_same_time | simul-
taneously | any_temporal_order |
one_by_one | …</td></tr>
<tr><td></td><td></td><td></td><td></td><td>dim</td><td>group_structure | time | space</td></tr>
<tr><td></td><td></td><td></td><td></td><td>value</td><td>coll | distr | same | different</td></tr>
<tr><td></td><td></td><td></td><td>scope_value</td><td>local</td><td>(numeric value calculated by algorithm)</td></tr>
<tr><td></td><td></td><td></td><td></td><td>global</td><td>(numeric value calculated by algorithm)</td></tr>
<tr><td>drs</td><td>out</td><td colspan="4">drs(U,Con), where U is a list of variables, and Con is a list of DRS conditions</td></tr>
</table>

Table 8 Feature Specification of Linguistic Disambiguation Information of NPs

mation includes the classification of determiners, verbs and other elements. The borderline between the different types of information is not always clear-cut and interactions occur. For example, the use of disambiguation triggers requires a combination of structural and lexical knowledge since certain words in certain positions function as explicit disambiguation triggers (e.g. the floating quantifier *each*). Non-linguistic information cannot be derived from our linguistic knowledge alone. It contains world-knowledge, knowledge about a particular domain or knowledge about a particular utterance context. I will also classify pragmatic knowledge as non-linguistic. Pragmatic knowledge contains for example knowledge about the utterance situation, knowledge about conversation principles etc. Non-linguistic knowledge is difficult to encode and in particular to automatically extract. My implementation contains the possibility to add first-order axioms that encode word-knowledge and domain knowledge. These axioms, however, have to be encoded manually and are not automatically given.

5.3.2.1 Explicit Triggers

Ambiguity and indeterminacy of plural constructions are often removed by adding extra linguistic material. I consider two types of additional cues:

- disambiguation markers (e.g. floating quantifiers)
- elaboration markers (e.g. part-structure modifiers)

Disambiguation Markers

Floating quantifiers like *each*, *both* trigger a distributive interpretation of a preceding individual denoting plural noun phrase.

(84) The customers each enter a card.

I consider only those constructions where the floating quantifiers occurs directly after the plural noun phrase (the *licensing* NP), but ignore constructions like

(85) The customers enter a card each.

where the floating quantifier does not occur directly after the noun phrase to which it relates.

The information added by disambiguation markers will be encoded in the lexical entry of the marker and will be unified with the `sem:quant:disambig:marker` feature of the noun phrase that is modified by the marker. For example the disambiguation marker *each* will instantiate the marker disambiguation feature of the licensing noun phrase as follows:

(86) Disambiguation information added to noun phrase modified by *each*

$$\left[\ \texttt{sem:quant:disambig:marker}\ \left[\begin{array}{l}\texttt{type: disambiguation}\\ \texttt{lex: each}\\ \texttt{dim: group\_structure}\\ \texttt{value: distr}\end{array}\right]\right]$$

That means the marker *each* triggers a distributive (`distr`) reading of the preceding noun phrase with respect to the lexical element of which the complete NP is an argument. See also section 4.5.4.4 and section 4.5.5.2 for an explanation of the other features. Disambiguation markers have to be manually classified. Currently, I consider *each* as an explicit marker that triggers the distributive reading. Additionally, partitive constructions with *each of* trigger the distributive reading.

Elaboration Markers

In section 4.5.5.2 I have discussed that if a speaker wants to express additional information about possible constellations of a (complex) event explicit modifiers and other markers are used.

(87) A man lifts two tables as a whole.

As indicated the semantics of these markers is less clear than that of floating quantifiers. I discussed that there are several dimensions along which an event can be detailed. Currently, I am considering the dimensions `group_structure`, `space` and `time`. Table 6 on page 192 summarized the dimensions and possible examples. Again, the markers have to be classified manually in the lexicon. Elaboration markers relate an object with and event. The object and the event variable are encoded in the feature structure. Furthermore, the marker feature is instantiated. To process elaboration markers the following features get instantiated:

(88) Disambiguation information added by the elaboration marker *as a whole*

$$\begin{bmatrix} \texttt{sem:} \begin{bmatrix} \texttt{object\_index: O} \\ \texttt{event\_index: E} \\ \texttt{quant:} \begin{bmatrix} \texttt{disambig:} \begin{bmatrix} \texttt{marker:} \begin{bmatrix} \texttt{type: elaboration} \\ \texttt{lex: as\_a\_whole} \\ \texttt{dim: group\_structure} \\ \texttt{value: coll} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

For a restricted domain a set of predefined elaboration markers has to be lexically predefined. The problem is, how to automatically recognize adverbials that function as elaboration markers. One approach is to search for synonyms of existing elaboration markers in machine readable lexica like WordNet (Fellbaum 1998, http://www.cogsci.princeton.edu/~wn/). For example for the elaboration adverbials *collectively* and *individually* WordNet lists the following synonyms:

(89) WordNet synonyms of *collectively*
jointly, collectively, conjointly, together, putþtogether

(90) WordNet synonyms of *individually*
individually, separately, singly, severally, one by one, on an individual basis

Other elaboration markers like *at the same place/time*, *in groups of 1/2/3*, *as a group/whole* can by constructed from the constituting components.

5.3.2.2 Lexical Information

Determiners

Similar to the existing approaches discussed in sections 5.2.1 and 5.2.2 I assume a classification of determiners. The following criteria are distinguished. The corresponding feature name can be found in brackets:

- general semantic classification
  - quantifier type (`quant_type`)

- monotonicity (`monotone`)
- quantity information (`quantity`)
- disambiguation information
    - distributivity type (`dist_type`)
    - scopal strength (`scop_lex`)

Similar to Schütze's approach the scopal strength of a quantifier will be related to the distributivity type.

Semantic Information. The semantic classification of determiners has already been discussed in section 4.5.2. I assume that there are two types of noun phrases, individual denoting and quantificational noun phrases. Only the former can receive a collective reading. Individual denoting noun phrases are further classified into definite (`def`), indefinite (`indef`), cardinality NPs (`card`) and proper nouns (`pn`). Quantificational noun phrases are subclassified into quantifying noun phrases (`quant`) and negated noun phrases (`no`). Furthermore each noun phrases is classified according to its monotonicity property. I only distinguish three types of monotonicity: left and right upward monotonicity (`up`) and right-upward monotonicity (`right-up`), but not left-upward monotonicity, and non-upward monotonicity (`non-up`). Furthermore, quantity information as discussed in section 4.5.2 is encoded. The feature `vague_spec` encodes whether the quantifier contains a vague specification of the number, if not the value is `no`.

For example the semantic contribution of the determiner *at least two* within the noun phrase *at least two men* will be instantiated as follows:

(91) Semantic contribution of the determiner *at least two* to a noun phrase

$$\left[\begin{array}{ll} \texttt{sem:quant} & \left[\begin{array}{ll} \texttt{quant\_type: card} & \\ \texttt{monotone: up} & \\ \texttt{quantity:} & \left[\begin{array}{l} \texttt{count\_unit: count\_unit} \\ \texttt{dimension: cardinality} \\ \texttt{value: 2} \\ \texttt{num\_rel: geq} \\ \texttt{vague\_spec: no} \end{array}\right] \end{array}\right] \end{array}\right]$$

Analogously, the semantic contribution of the determiner *few* within the noun phrase *few men* will be instantiated as:

(92) Semantic contribution of the determiner *few* to a noun phrase

$$\left[\begin{array}{ll} \texttt{sem:quant} & \left[\begin{array}{ll} \texttt{quant\_type: card} & \\ \texttt{monotone: non-up} & \\ \texttt{quantity:} & \left[\begin{array}{l} \texttt{count\_unit: count\_unit} \\ \texttt{dimension: cardinality} \\ \texttt{value: 2} \\ \texttt{num\_rel: geq} \\ \texttt{vague\_spec: few} \end{array}\right] \end{array}\right] \end{array}\right]$$

Distributivity Type. Similar to Schütze (1989) I will classify noun phrases as to their inherent tendency to be read distributively or collectively. Recall, that the inherent distributivity of a noun phrase can be tested in different contexts. If a noun phrase easily combines with collective predicates like *meet* or *lift a table together* it is not inherently distributive:

(93) Most/at least two men gather/lift a table together.
(94) Two/some men gather/lift a table together.

The quantifiers *most* or *at least two* are difficult to interpret with a collective predicate whereas *two* or *some* easily combine with collective predicates. As a further test we can combine the noun phrases with predicates like *lift a table* that allow for both readings (in a neutral context). Depending on the determiner we observe different preferences as to a distributive or collective interpretation.

(95) Most/at least two men lifted a table.
(96) Two/some men lifted a table.

Most and *at least two* strongly prefer a distributive interpretation whereas *two* and *some* prefer a collective interpretation. See the comments in section 5.2.2.2 above.

In my account all individual denoting noun phrases can in principle get a collective reading, only quantifying noun phrases are always read distributively. That means NPs like *few man*, *at most two men*, *most men* can in principle be read collectively. Apart from this my classification will be similar to Schütze's proposal.

(97) Distributivity Scale

`d`	every, each, no
`d >> c`	many, many of, few, few of, most, most of, more than half of the, both
`d > c`	at least three, exactly n, at most n, all, all the
`c > d`	numerals (two, three, …), some, several, a few, *the*$_{\text{pl}}$, personal plural pronouns, coordinations of NPs of this type
`c >> d`	ø
`c`	singular individual denoting NPs (a, one, at least one, …)

Note, that there are no plural noun phrases that definitely trigger a collective reading. For singular noun phrases collective and distributive readings are not relevant. Nevertheless, I will give them the dummy classification `c`, that I will need for the DRS construction procedure (no distribution takes place).

The distributivity is encoded with the feature `dist_type` as follows. For example *at least two* will instantiate the feature as follows:

(98) Distributivity type of *at least two*:

[`sem:quant:disambig:default:dist_type: 'd>>c'`]

The classification is of course only a tendency in neutral contexts as discussed above.

Scoping Preferences. Quantifying noun phrases and distributively interpreted individual denoting noun phrases have scopal effects. I agree with Schütze (1989) and VanLehn (1978) that the higher the degree of distributivity the higher is the tendency of a noun phrase to take wide scope. For singular noun phrases the scope value has to be determined separately. The different types of noun phrases are assigned numeric scope values that will help to calculate relative scopings. The following scopal hierarchy is proposed. I will adopt the numeric classification suggested by Pafel (1988) and modified by Schütze (1989):

(99) Scoping Scale

5	every, each, no
5	many, many of, few, few of, most, most of, more than half of the, both
3	at least three, exactly n, at most n, all, all the
2	numerals (one, two, three, …), some, several, a few, coordinations of NPs of this type
1	ø, singular indefinites

Although the scope values can mostly be derived from the distributivity type and the quantifier type I will encode the scope value using the extra feature `scop_lex`. This helps to fine-tune the system for new quantifiers. An example for the encoding of the scope value is:

(100) Scope Value of *many*

[`sem:quant:disambig:default:scop_lex: 5`]

Note that (singular and plural) definites and proper nouns are not assigned scopal values. Proper nouns always get topmost scope. And the interpretation of definites follows special rules. Definites are often interpreted anaphorically which is why the scope value is not relevant. If they are not interpreted anaphorically their reference has to be independently given. Often they are also given topmost scope. However, there are exceptions to these rules for example in the construction *the title of every book* where the definite noun phrase *the title* is within the scope of *every book*. I have not developed a theory of definites in this thesis which is why I will not further discuss this here.

Verbs

Lexical entries of verbs contain syntactic and semantic disambiguation information. The *syntactic* information distinguishes full verbs from copular verbs and states the syntactic category of the verb's arguments. I am not further commenting on *syntactic* information like agreement here. Concerning the *semantic* information I was mainly interested in the lexical contribution of verbs to plural disambiguation. My hypothesis was – and still is – that verb semantics plays a crucial role in plural disambiguation, yet I found it very difficult to systematically isolate relevant disambiguation triggers. I do not consider this primarily as a problem of automatic classification but even more as a theoretical problem since the verb semantics is very complex. As already discussed in the criticism of Aone's distribution patterns (see section 5.2.2.2, pp. 229) the factors deciding about possible and plausible readings and distribution patterns involve more than lexical knowledge about verbs.

As a starting point I collected the following lexical information about verbs that I consider to be relevant for the process of disambiguation and for the process of determining subentailment properties of verbs. Most of this information can be automatically collected from the existing lexica COMLEX (see http://nlp.cs.nyu.edu/comlex/) and WordNet (see http://www.cogsci.princeton.edu/~wn/), and additionally from the concise verb classification given in Levin (1993). See appendix D for more details about the lexica. I have not yet fully included these information sources into my prototypical implementation, however, the current proposal lays the foundation for possible extensions.

Every lexical entry of a verb contains syntactic and semantic information needed for disambiguation. In Table 9 I will list the relevant features. The syntactic information (`syn`) contains

Verbs				
syn	index	E (Variable for Event Index)		
	cat	iv | tv | dv		
	subcat	arg1	index	I1
			cat	np
		arg2	index	I2
			cat	np
		arg3	index	I3
			cat	np | pp | …
			pval	to | of | for
	agr	number	sg | pl	
	v_form	base | fin		
sem	log_rel	(determined by lexicon, e.g. `enter`)		
	v_dist_type	arg1	d | c | m | ‘d>c’ | ‘c>d’	
		arg2	d | c | m | ‘d>c’ | ‘c>d’	
		arg3	d | c | m | ‘d>c’ | ‘c>d’	
	c_class	vcollective | intrans_recip | …		
	evntl	state | event		
	v_class	measurment | perception | motion | ingesting | body | combining | …		

Table 9 Feature Specification of Syntactic and Semantic Verbal Disambiguation Information

the category (`cat`) of the verb which – for the English fragment considered here – can be intransitive (`iv`), transitive (`tv`) or ditransitive (`dv`). This information will be needed to distinguish external and internal arguments. Furthermore, the category (e.g. `syn:subcat:arg1:cat:np`) and the index (e.g. `syn:subcat:arg1:index:I`) of the subcategorized element is stored plus – if the argument is a prepositional phrase – its associated preposition (e.g. `pval:to`). The index is necessary for the identification of the argument within the DRS construction, and for the relevant feature unifications.

The semantic (`sem`) information of the lexical entry of a verb currently contains the following elements. For each argument of the verb possible defaults for the argument interpretation with respect to collectivity and distributivity are stored (`v_dist_type`). The information is actually derived from other semantic entries of the verbs as follows. First, I adopt a distinction of the COMLEX lexicon according to which verbs are classified as to whether they require plural arguments, mass arguments or group NPs. COMLEX distinguishes "intrans-reciproc" and "VCOLLECTIVE" verbs. I will store this in the feature `c_class` (for collectivity class). The class of intransitive reciprocal verbs (`intrans_recip`) contains verbs that only occur intransitively when they have a plural subject (*some men*), a singular noun phrase built from a collective noun (*a group*), a mass NP (*water*), or a subject consisting of coordinated NPs. Verbs of this subclass occur with *each other* or a prepositional phrase consisting of *with* or *from* followed by *each other* as complements, although complementation is not mandatory.

(101) Examples for intransitive-reciprocal verbs:
agree, *argue*, *associate*, *combine*, *confer*, *conflict*, *differ*, *meet*, *part*, *separate*, …

If an intransitive-reciprocal verb is used intransitively I will give the subject a collective reading, indicated by instantiating the `v_dist_type` feature of the subject with the value `c` (`sem:v_dist_type:arg1:c`). This excludes a distributive reading for the subject. Further subentailments triggered by the use of intransitive-reciprocal verbs will require the inference component (chapter 6) which could make use of the feature `intrans_recip`. There are also non-reciprocal verbs that require plural, collective or mass objects as arguments. In COMLEX these verbs are classified as "VCOLLECTIVE". Verbs of this class can be used transitively and intransitively. If used transitively a VCOLLECTIVE verb requires as a direct object a plural NP, a mass NP or a collective NP. If used intransitively these types of arguments are required in subject position.

(102) Examples for "VCOLLECTIVE" verbs:
accumulate, *assemble*, *collect*, *convene*, *disperse*, *cluster*, *scatter*

Again, if used intransitively verbs of this class will instantiate the `v_dist_type` feature of the subject with the value `c` (`sem:v_dist_type:arg1:c`), if used transitively the direct object (`arg2`) gets the feature `c` (`sem:v_dist_type:arg2:c`). If the verb contributes no information about the distributivity type of its arguments the `v_dist_type` feature is instantiated to `m` for "mixed". Other instantiations like `'c>d'` or `'d>c'` may be derived from the verb class or from other semantic information of the verb. However, I have not yet found systematic lexical factors that lead to this instantiation. Also, the classification of verbs as distributive verbs requires additional information, as discussed in the next sections.

The semantic part of the verb lexicon also stores whether the verb denotes a state or an event. Admittedly, this information is difficult to extract automatically. The subentailments of state verbs often differ from the subentailments of event verbs (see Schwertel 2000). Finally, I will leave a slot (`v_class`) to store some verbal classifications proposed by Levin (1993). Levin classifies over 3000 English verbs according to shared meaning and behaviour. She presents classes that share a kernel of meaning. I expect that some of the classes allow for systematic inferences concerning collective/distributive ambiguities and concerning possible distribution patterns. For example, certain perception verbs in Levin (1993, pp. 185) tend to trigger a distributive reading of the subject noun phrase (e.g. *see*, *hear*, *notice*, *smell*, *taste*, *gaze*, *glance*, *look*, *stare*, *feel*). This tendency would have to be marked as `'d>c'` or even `'d'` in the distributivity type (`v_dist_type`) of the corresponding argument. Verbs of ingesting (Levin 1993, pp.

213) could – similar to Aone's proposal (see section 5.2.2.2, pp. 229) – trigger a default inference pattern according to which the same object cannot be consumed twice. Examples of ingesting verbs are *drink*, *eat*, *chew*, *swallow*, *consume*, *devour*. Thus the additional storage of the verb class can help to disambiguate collective/distributive readings and furthermore to predict plausible distribution patterns and distributive subentailments. A more concise study would have to find out how far we get with the classifications. I expect, however, that considerable manual postprocessing is necessary to adapt the proposed classification for plural disambiguation. Furthermore, the predictions derived from the verb classes should most probably not be seen as constraints in the sense of Aone but rather as defaults that help to prefer one alternative but that do not rule out other readings. Furthermore, empirical validations of the predictions derived from the verb classes should be made, a task that goes beyond the scope of this thesis. Also, lexical ambiguity of verbs is not solved by this approach.

Unfortunately, in the current proposal, for the majority of verbs there are no automatic means to extract preferred readings. One problem is that preferences for certain readings depend also on the type and the semantics of the arguments (recall the pizza eating example (69)). For these verbs the lexical entry will admit both readings. A next task is to formulate inference rules for certain verb classes that capture desired inferences, e.g. subentailment properties. For example, for the majority of non-collective intransitive verbs we observe that the subject gets an indeterminate interpretation. The sentence

(103) Five men disappear.

means that each of the men disappears, either alone or together with others. I will therefore take as a default an (indeterminate) collective reading and encode the entailments with the help of inference rules (see chapter 6). A similar observation can be made for the internal argument of transitive verbs:

(104) John lifted two tables.

Again, inference rules trigger the distribution to the individual members of the internal argument (Schwertel 2000). Again this is only a starting point and needs further investigation.

In my approach I deliberately propose a coarse, but tractable semantic classification of verbs (e.g. into event and state verbs), and achieve already good results. I observed that to perform a more precise classification of verbs one immediately faces the problem that the interpretation of verbs is highly context dependent. Classifying verbs often requires specific information about individual verbs, and additional knowledge about their arguments. A finer classification of verb semantics would allow us to infer more precisely what happened to the individuals involved in a "collective" action, and possibly would also improve the disambiguation algorithm. For limited domains with a clearly defined subject area the necessary additional information could perhaps be made reasonably manageable and improve the disambiguation process.

Adjectives

Adjectives can also lead to collective/distributive ambiguities. These ambiguities can occur in attributive and in predicative position (see pp. 227 above). For example, the type of the adjective in predicative position influences the interpretation of the subject. Most adjectives lead to a distributive interpretation, e.g.

(105) The balls are red.
(106) Five customers are ill.

There is a limited class of adjectives, however, that leads to a collective/distributive ambiguity:

(107) The books are expensive.
(108) Five tables are heavy.

Other examples are *heavy*, *light*, *dense*, *heavyweight*, *massive*, *middleweight*, *ponderous*, *thick*, *thin*, *wide* etc. A further class of adjectives can only be interpreted collectively: *numerous*, *equal*, *identical*, *adjacent*, *compatible*, *consistent* etc. Similar to the classification of verbs I propose to add into the lexical entry of adjectives its distributivity type. The features listed in Table 10 are used for intransitive adjectives.

<table>
<tr><th colspan="4">Adjectives</th></tr>
<tr><td>syn</td><td>index</td><td colspan="2">I</td></tr>
<tr><td>sem</td><td>log_rel</td><td colspan="2">e.g. red</td></tr>
<tr><td></td><td>adj_dist_type</td><td colspan="2">d | c | m</td></tr>
<tr><td></td><td>reading</td><td>result</td><td>d | c</td></tr>
<tr><td></td><td></td><td>stability</td><td>pref | unpref | fix</td></tr>
</table>

Table 10 Adjective Disambiguation Information

The feature `adj_dist_type` classifies adjectives into distributive (`d`), collective (`c`) and ambiguous (`m`) adjectives. The feature `reading` encodes the value (`result`) calculated by the disambiguation module plus the “stability” (`stability`) of the reading. This information is not lexically given but calculated during disambiguation.

I found no method to classify the adjectives automatically according to their distributivity type. However, since the class of non-distributive adjectives is small, I assume a manual classification taking into account synonym lists offered e.g. by WordNet.

Adverbs

The disambiguation information provided by adverbs is discussed in the section about explicit disambiguation triggers (see pp. 238).

Nouns

The lexical information of common nouns contains information that is relevant for the construction of the semantic representation and also – in a restricted sense – needed for disambiguation. Since some of the information is needed for both syntactic agreement and semantic construction the information is split between the `syn` and the `sem` features. Common nouns are subdivided into different classes. I will distinguish “normal” common nouns (*man*, *card*, *gold*) from nouns denoting measurement units (*kg*, *ounce*, *cent*), and nouns denoting measurement dimensions (*length*, *area*, *weight*) in certain contexts. These three types of noun phrases are assigned different syntactic categories. Some of the nouns are ambiguous and can be used in more than one function. For example in

(109) John drops two dollars.

the noun *dollars* is used as a normal countable plural noun. However, in

(110) The book costs two dollars.
(111) The prize of the book is two dollars.

the noun *dollars* is used as a count unit with the dimension prize. For the applications pursued in this thesis this ambiguity has to be controlled by the users. More concretely, count units and dimensions are predefined in the lexicon. Furthermore, certain syntactic constructions disambiguate which meaning of the noun is chosen.

The information for “normal” common nouns is encoded using the features in Table 11.

Nouns			
syn	index	I	
	cat	cn	
	agr	number	sg | pl
		countable	yes | no
		collective	yes | no
		gender	masc | fem | masc_fem | neuter
sem	log_rel	e.g. man	
	object_type	person | time | object	

Table 11 Common Nouns

The lexicon distinguishes between countable and uncountable nouns (`countable:yes` vs. `countable:no`). Countable nouns can have a singular (`sg`) and a plural (`pl`) form whereas uncountable nouns are only used in the singular. Uncountable nouns comprise mass nouns (*water*) and so-called uncount nouns (e.g. *intelligence*) although this is a rather coarse simplification. COMLEX distinguishes count (“COUNTABLE”), uncount (PLURAL *NONE*) and mass nouns (“NCOLLECTIVE”). This means the information is present but is currently not used in the prototypical implementation. There are two types of singular countable nouns: nouns that denote non-collective objects (*man*, *card*, *book*) and collective nouns that denote a group of objects (*group*, *couple*, *assembly*, *family*), encoded as (`collective:no` vs. `collective:yes`). Syntactically, collective nouns combine with collective verbs (see page 244 above). In the semantic representation, however, singular NPs formed from collective nouns are treated like atomic objects that – though consisting of other objects – are viewed as a unit and can be counted. Schütze proposes that concerning disambiguation plural NPs consisting of collective nouns (*the families*) when combined with collective predicates (*meet*) prefer a (partially) distributive reading, i.e.

(112) The families gather.

preferably means that each of the families gathers. I have not included this preference into my proposal.

The gender feature (`gender`) encodes the natural gender of NPs. It is needed e.g. for correct anaphora resolution.

Furthermore, for each noun phrase three semantic types (`object_type`) are distinguished: nouns that denote persons (e.g. *woman*, *student*, *borrower*), time (*year*, *afternoon*, *moment*) or everything else (e.g. *table*, *book*, *water*, *intelligence*). The corresponding features are `person`, `time` and `object`. The features `person` and `time` can also be automatically extracted from COMLEX ("NHUMAN" and "NTIME"). The types of the nouns are needed to disambiguate the modification type of adverbials formed with the same preposition, e.g. *in the morning* is a temporal, whereas *in the water* is a locative adverbial.

Measurement nouns encoding count units receive a separate syntactic category (`cn_measurement`). Some examples (extracted from COMLEX) are:

(113) measurement nouns
block, centimetre, century, day, foot, hand, hour, inch, kg, mile, millisecond, pound, row, second, segment, week, year

The lexical information contributed by measurement nouns is listed in Table 12.

Measurement Nouns			
syn	index	I	
	cat	cn_measurement	
	agr	number	sg | pl
		countable	yes
sem	log_rel	e.g. ounce, kg, …	
	object_type	count_unit	
	dimension	implicit	weight | length | area | …

Table 12 Measurment Nouns

Semantically, measurement nouns are typed as `count_unit` and carry the inherent dimension of the count unit (e.g. `weight` for the count unit `kg`). In section 4.5.2.12 I discussed that in constructions like

(114) two kg of apples

the measurement dimension `weight` is implicitly given by the count unit `kg`.

The measurement dimension can, however, also be made explicit by dimension nouns like *weight* in

(115) the weight of the apples is two kg

or with measure verbs like *weigh* in

(116) the apples weigh two kg

or measure adjectives like *long* in

(117) The line is 10 inches long.

Here, I am only interested in dimension nouns. Dimension nouns that make the measurement dimension explicit are given the separate syntactic category `cn_dimension`. Table 13 lists the

Dimension Nouns			
syn	index	I	
	cat	cn_dimension	
	agr	number	sg
sem	log_rel	e.g. weight	
	object_type	dimension	
	dimension	explicit	weight | length | area | …

Table 13 Dimension Nouns

relevant features. Examples for dimension nouns are

(118) dimension nouns
age, altitude, area, breadth, circumference, diameter, height, intensity, length, luminosity, strength, thickness, volume, wavelength, width

It is possible that the explicitly given dimension conflicts with the implicitly given dimension of the count unit.

(119) *The weight of two books is 3 cm.

This conflict is currently not resolved in the implementation but the explicitly given dimension overwrites the implicitly given dimension. The user is responsible for correct use of dimensions and units. However, these semantic conflicts could be resolved within the reasoning component.

Note that count units can be automatically extracted from COMLEX. The implicit dimension of the count unit either has to be manually given or derived from other lexica like WordNet. For example in WordNet *kg* is assigned the hypernym “weight unit”, *cm* is associated with “linear metric unit”, *acre* is an “area unit”. More problematic cases are words like *row* or *cup*. In WordNet *cup* can be a not further specified “amount quantity” but also a normal countable noun. This shows that in precise technical texts units and dimensions should be clearly defined. The example *cup* also shows that nouns are ambiguous in that – depending on the context – they denote a count unit or a normal count noun.

Concerning plural disambiguation the proposed classification of nouns is needed to distinguish measurement NPs from normal plural NPs. For example in constructions like

(120) The apples weigh two ounces.

the NP *two ounces* will always be assigned a collective interpretation. We cannot say

(121) *The apples weigh each of two ounces.

Whether the subject NP is interpreted collectively or distributively is a different issue. Also in a sentences where the quantity of the NP is given by a measurement NP like in

(122) John buys two kg of apples.
(123) Two kg of apples are cheap.

we assign a collective interpretation to the NP *two kg of apples*. Distributive interpretations are not possible.

Note, that the dimension of measure constructions can also be identified by using certain adjectives. For example in

(124) The line is 10 inches long.
(125) A ten inch long line.

In COMLEX these adjectives have the feature “ASCALE”. An adjective is ASCALE if it can occur to the right of a measure sequence consisting of a numerical quantifier and a noun which denotes a count unit. It indicates the dimension in which the measurement occurs (e.g. *width*, *length*, *height*). Measure verbs are not classified in COMLEX, they could be extracted from WordNet as verbs that have as hypernyms “measure”.

Prepositions

There are some prepositions like *among* or *between* that require a plural agreement if used with a noun phrase. Apart from that I do not assume that prepositions contribute disambiguation information.

5.3.2.3 Structural Information

There are several information sources that can be extracted from the grammatical structure of the sentence which are used for plural disambiguation and for scope disambiguation. Most of the structural information sources have been discussed in section 5.2.

Grammatical Function

The grammatical function (`gra_fct`) of a noun phrase is needed for both, plural disambiguation and for scope disambiguation. In my fragment, I will distinguish the following grammatical functions and – for scope disambiguation – associate them with the numeric values proposed by Pafel (1988), see 5.2.2.2 above.

(126) Grammatical Function
5 surface subject (`subj`)
4 indirect object (`io`)

3 prepositional object (`po`)
2 direct object (`do`)

The numeric values are calculated in the disambiguation component and not stored as feature values. The grammatical functions are assigned during parsing. The grammatical function is not only needed for scope disambiguation but also for plural disambiguation in that certain noun phrases prefer a collective reading in internal argument position. See section 5.4.2.2 below.

Ordering Information

The proposed algorithm stores the linear order of noun phrases within an ordered NP list that is generated during parsing. The linear order will be used mainly for scope disambiguation. But it is also relevant for plural disambiguation in that an individual denoting noun phrase that is *not* followed by other noun phrases often prefers a collective interpretation. Since the ordering information is relative to the comparison of two noun phrases there is no feature associated to it in the representation of a single noun phrase. See section 5.4.3 for further details.

Syntactic Structure

Similar to Pafel who distinguished NPs according to the structure of the restrictor I will store the syntactic structure (`syn_struct`) of the NP with respect to the head noun of the NP. I will distinguish the following noun phrase structures, and assign the feature value given in brackets:

- noun (`noun`)
  two men, *two red men*
- partitive (`partitive`)
  two of the men, *each of the men*, *most of the men*
- prepositional phrase (`pp`)
  two books of the men, *three cards of a customer*
- possessive (`poss`)
  the customers' books, *a man's books*
- relative clause (`rel_clause`)
  two cards which belong to a customer, *two cards that a customer enters*

In Pafel's algorithm simple noun phrases were assigned the numeric value 3 that entered the scoping algorithm, all other noun phrases have other NPs in their restrictor and were therefore assigned a higher numeric value, viz. the value 5. I will adopt these values but will use the more fine-grained distinction to assign e.g. partitives a higher potential for distributive readings than the non-partitive counterparts.

Note here, that the current algorithm does only calculate vertical scopings. That means I do not calculate the likelihood of an embedded NP to outscope the embedding NP as is e.g. the case in

(127) a representative of each company

I have not included additional vertical scoping mechanisms since I was mainly interested in plural disambiguation and not in scope resolution. You can find algorithms for vertical scoping for example in Allen (1995) or Alshawi (1992). Also interesting with respect to vertical scoping are the data collected by VanLehn (1978) which were briefly addressed above.

5.3.2.4 Non-Linguistic Information

The first part of the current disambiguation algorithm does not consider pragmatics, world-knowlege or context. The inference component introduced in chapter 6, however, allows to integrate word-knowledge or contextual knowledge in the form of first-order axioms. The offered reasoning will, however, not be defeasible.

5.4 The Plural Disambiguation Algorithm

5.4.1 Overview

Every non-distributive noun phrase can – in principle – be read distributively or collectively. Disambiguation for plurals means that for each plural noun phrase in a sentence it has to be decided whether that noun phrase has to be read distributively or collectively with respect to a verb phrase, to another noun phrase or to other elements of the sentence. If the sentence contains more than one noun phrase, additionally the relative scopes of the noun phrases with respect to each other have to be determined. In my approach the collective reading is indeterminate concerning what actually happened with the individuals that constitute the plural object. Thus cumulative and mixed readings are absorbed by the collective reading which already considerably reduces the number of logical representations. The following disambiguation algorithm aims at restricting the possible collective/distributive ambiguities and additionally restricts possible scope relations. More concretely the algorithm transfers possibly ambiguous English sentences to disambiguated fully specified logical forms using the information introduced in the previous section 5.3. The disambiguation algorithm focuses on collective/distributive ambiguities but also contains a scope disambiguation component. The algorithm currently only deals with sentences in the active voice. Furthermore, the scoping algorithm is only implemented for horizontal scoping. Vertical scoping that occurs e.g. with nested NPs (*two books of a man*) or NPs modified by relative clauses (*two books that a man reads*) has not yet been added.

The disambiguation algorithm is prototypically implemented within the DRoPs system. In this section I introduce the disambiguation part of DRoPs, in chapter 6 the reasoning part is discussed. Figure 4 on page 253 shows an overview of the disambiguation components of DRoPs.

A user inputs English text which is analysed by the parser. The parser collects syntactic and semantic disambiguation information and instantiates the appropriate feature values some of which were introduced in section 5.3.2. The disambiguation information is partly extracted form the lexical entries, partly associated with grammar rules and partly externally given by additional disambiguation rules (constraints and preferences) that are applied after parsing.

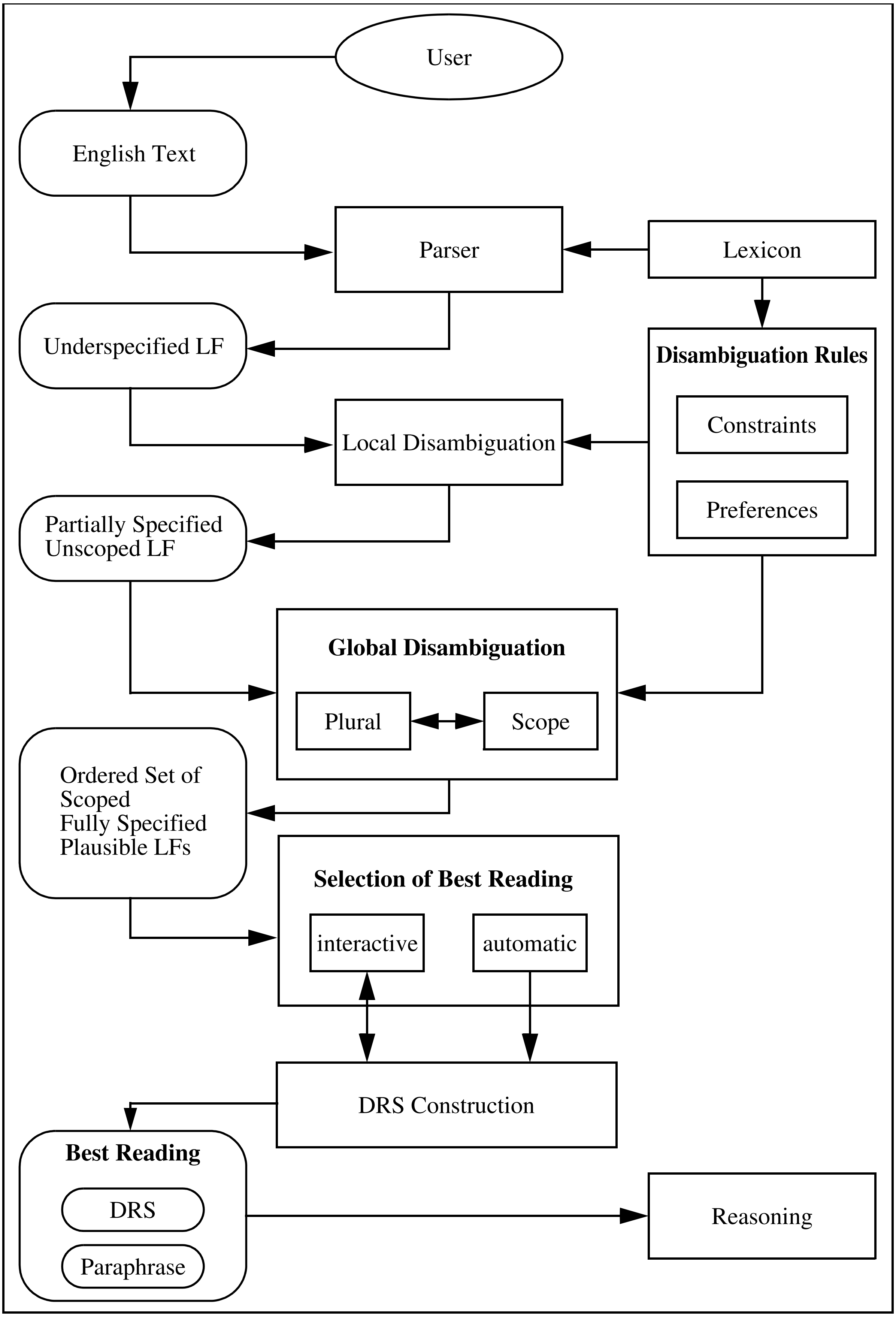


Figure 4 Architecture of the DRoPs Disambiguation Component

Also, information needed for DRS construction is collected by the parser. The parser generates a first semantic representation that underspecifies the collective/distributive interpretation of noun phrases, and the scope of noun phrases and negation. This first underspecified logical form (LF) contains the logical form for the matrix of the sentence, and a list of scope bearing elements, here consisting only of the NPs that are arguments of the matrix predicate. Additionally, the list can contain negation as a scope bearing element. The introduction of an unscoped NP list borrows ideas from the NP storing technique originally proposed by Cooper (1983). See also Blackburn and Bos (2000a, section 3.3) for an overview. In my approach each NP in the list contains the relevant local disambiguation information plus local DRS structures needed for DRS construction. The information is represented as (partially) instantiated feature structures according to Table 8 on page 237. In a next step the locally available disambiguation information is used to calculate a first partially specified unscoped logical form. This calculation is based on a number of constraints and preferences for plural disambiguation. The rules will be introduced in section 5.4.2. Some of the constraints and preferences lead to a default interpretation of NPs, e.g. a default collective interpretation would lead to the following instantiation of the respective semantic feature within the NP:

(128) Instantiation for default collective reading

$$\left[\texttt{sem:quant:disambig:reading:}\begin{bmatrix}\texttt{result: coll}\\ \texttt{stability: pref}\end{bmatrix}\right]$$

This instantiation does not mean that the collective reading is fixed “forever”. The feature value `stability:pref` indicates that the preferred default can be overridden in a later step. Only if the stability value were `fix` no overriding would be possible. The value `fix` occurs e.g. when the NP is modified by a disambiguation marker like *each*, or if the NP is a singular NP like *every man* etc. The partially specified representation, in particular the partially specified NP list, is then input to the global disambiguation component. The component is called “global” because it does not only focus on a single NP but evaluates the NPs with respect to each other. The global disambiguation component has several tasks:

- generate all *plausible* readings with respect to
  - collective/distributive ambiguities
  - scope ambiguities
- rank the readings according to their plausibility

The output of the global disambiguation component is an ordered set of representations for the plausible readings where each representation is associated with a numeric tag that indicates the (relative) plausibility of the reading. The readings are still encoded with the help of NP stores. The NP stores differ from each other with respect to scoping and/or with respect to different instantiations for collective/distributive readings. The algorithm then offers two possibilities to proceed. One possibility is that the best reading, i.e. the reading with the highest numeric value, is automatically selected, and from the NP store the DRS construction component generates a standard DRS and a simple paraphrase indicating the reading in verbal form. A second possibility is that the readings (together with the simple paraphrase) are presented to the user who has to interactively select the intended reading. This second possibility requires that DRSs and the simple paraphrases are generated for each of the plausible NP stores until the user selects one reading as the intended reading. In the next step the fully specified DRS is passed to the reasoning component of DRoPs which will be dealt with in chapter 6.

5.4.2 Disambiguation Rules

5.4.2.1 Disambiguation Constraints

Disambiguation constraints guarantee that only semantically *possible* readings are generated from the underspecified representation. The results of the application of constraints are not defeasible. There are scopal constraints that rule out certain scopings, and there are collective/ distributive constraints that fix the collective or distributive interpretation of an NP or adjective. In the practical implementation, constraints operate on different processing levels. There are lexical constraints, structural constraints and combinations thereof. For example, there is a lexical constraint that noun phrases introduced by *every* always get a distributive reading. Furthermore, the floating quantifier *each* is associated with a constraint that gives the licensing noun phrase a distributive reading. Following is a list of constraints currently used for the DRoPs disambiguation algorithm.

Collective/Distributive Constraints

(C1) Types of Readings
The algorithm distinguishes only collective (`coll`) and distributive (`distr`) readings. There are no semi-distributive (mixed) scoped readings. The collective reading is indeterminate as to how the individuals making up the group are involved in the relation denoted by the rest of the sentence. Possible subentailments are treated by axioms in the inference component.

(C2) Only individual denoting plural NPs, i.e. NPs of type ‘`c>>d`’, ‘`c>d`’, ‘`d>c`’, ‘`d>>c`’ can in principle get a collective reading.

(C3) Plural NPs of type ‘`d`’ always get a distributive reading. This includes partitive NPs starting with *each of*. Also, quantifying NPs never get a collective reading.

(C4) Floating Quantifiers
The floating quantifier *each* triggers a distributive reading of the licensing NP.

(C5) Elaboration Markers
Elaboration markers (*as a whole*, *together*) can mark certain NPs as collective. These NPs cannot get a distributive reading.

(C6) Verbs and adjectives marked as collective in a certain argument position trigger a collective reading of the NP in that position. Distributive subentailments are handled by additional inference rules.

(C7) Distributive NPs cannot combine with collective verbs or collective adjectives.

(C8) NPs interpreted as measurement phrases (see section 4.5.2.12) are always interpreted collectively.

(C9) Logically equivalent readings that are due to a scope reversal of two collectively interpreted NPs are eliminated. Only one reading is kept.

Scoping Constraints

I have only included a few simple scopal constraints. For more information on these constraints see for example Moran and Pereira (1992) or Allen (1995, pp.349).

(C10) Scope of nested NPs
A quantifier from elsewhere in a sentence cannot come after the quantifier associated with a head noun and before the quantifier associated with a noun phrase in the head noun's complement.

The constraint has for example the effect that the sentence

(129) Every representative of a company saw most samples.

cannot get an interpretation with the scopal order in (130)a which would mean something like (130)b.

(130) a. *every - most - a
b. *For every representative it is the case that for most samples there is a company such that the representative of that company saw that sample.

5.4.2.2 Disambiguation Preferences

The application of constraints leads to a (partially) specified semantic representation. The constraints fix or constrain some of the disambiguation features. Preferences are then applied to these partially specified logical forms in order to generate and rank plausible fully specified readings. More concretely, a list of fully specified scoped logical forms ordered according to plausibility is generated. Note, that the preferences cannot override instantiations fixed or restricted by constraints. If, after applying the constraints, the disambiguation values are not yet determined the preference rules are used to instantiate the features. These instantiations are marked as `default` and can be overridden in later disambiguation steps. My algorithm includes preference rules for scoping and preference rules for collective/distributive ambiguities. The scoping preferences are calculated similar to the proposal by Pafel (1988) explained above in section 5.2.2. I will therefore not further motivate the scoping preferences here but focus on the collective/distributive preferences. I will first list the preferences verbally and will then show in section 5.4.3 how the preferences are used to calculate plausible readings.

Collective/Distributive Preferences

The application of preferences presuppose that the collective/distributive reading of an NP has not been fixed by constraints. That means the feature `stability` may not have the value `fix`. The preferences instantiate the reading feature with the corresponding value for either the collective reading (`reading:result:coll`) or the distributive reading (`reading:result:distr`). Additionally, the reading is marked as a default (`reading:stability:pref`) which means that the other reading is still possible, e.g. if the default is collective the distributive reading is still possible and would be represented by the instantiation `reading:stability:unpref`. The overall plausibility of the preferred and unpreferred readings is checked in a later processing step (see below).

There is a set of preference rules that is applied to non-partitive NPs which occur as arguments of verbs such that the position where the NP occurs is not otherwise marked as collective or distributive. Depending on the distributivity type of the NP different defaults are predicted. The distributivity types were defined above in (97) on page 241. The defaults are summarized in Table 14. All preferences presuppose consistency with existing constraints, otherwise the

	iv	tv		dv			prep	cat	Preference Number
dist_type	subj	subj	do	subj	do	io	po	gra_fct	
c>>d	coll	coll	coll	coll	coll	coll	coll	result	(P1)
	pref	pref	pref	pref	pref	pref	pref	stability	
c>d	coll	coll	coll	coll	coll	coll	coll	result	(P2)
	pref	pref	pref	pref	pref	pref	pref	stability	
d>c	coll	distr	coll	distr	coll	distr	distr	result	(P3)
	pref	pref	pref	pref	pref	pref	pref	stability	
d>>c	distr	distr	coll	distr	coll	distr	distr	result	(P4)
	pref	pref	pref	pref	pref	pref	pref	stability	

Table 14 Default Interpretation of Non-partitive NPs as Arguments of Mixed Verbs

preferences are not triggered. The table shows the default instantiations of the features. In the rightmost column the identification number of the preference is listed. For example, the preference (P4) can be paraphrased as follows.

(P4) Noun Phrases of type ‘d>>c’
If a plural NP with the distributivity type ‘d>>c’ occurs as direct object of transitive or ditransitive verbs it gets a default collective reading. Otherwise it gets a default distributive reading.

The other preference rules have to be read accordingly. The table shows that I suggest a finer classification than e.g. Schütze (1989) in that I make different predictions for the distributivity types ‘`d>>c`’ and ‘`d>c`’. Furthermore, in my approach NPs of the same distributivity type can behave differently in different grammatical functions, e.g. plural noun phrases of type ‘`d>c`’ or ‘`d>>c`’ in direct object position do not get a default distributive reading but a default collective reading (despite their distributivity tendency). This preference for collective readings in internal argument position has been described above in section 4.3. Also, I assume a distinct behaviour of the distributivity types ‘`d>>c`’ and ‘`d>c`’ with respect to e.g. the subject position of intransitive verbs. In this position I assign noun phrases of type ‘`d>c`’ like *at least two men* a default collective reading whereas noun phrases of type ‘`d>>c`’ (*most men*, *few men*) are assigned a default distributive reading. The reason is that NPs of the latter type hardly get a collective reading as discussed e.g. in sections 4.5.2.6 and 4.5.2.7 above, whereas noun phrases of type ‘`d>c`’ can have collective readings. Admittedly, the preferences, in particular the collective preferences in subject position of intransitive verbs, are debatable. Together with inference rules that model distributive subentailments I found, however, that these preferences are most suitable to model intuitive reasoning processes.

(P5) Partitives
Partitives with determiners of type ‘`c>>d`’ and ‘`c>d`’ trigger the same interpretation as NPs of type ‘`d>c`’. Partitives with other determiners inherit the defaults from the type of their main determiner.

Preference (P5) predicts that sentences like

(131) Three of the students presented a project.

prefer a distributive reading of the subject noun phrase, although a collective reading is still possible. According to my intuition partitives behave differently from their non-partitive counterparts in this respect. However, this observation needs further empirical investigation. Partitives are considered to be different in that they more likely get an *exactly*-reading and not an *at least*- reading (see section 3.2.2.3). As indicated, in my approach the exactly reading is not modelled unless explicit determiners force it.

(P6) Adjectives
Mixed adjectives that combine with a plural noun phrase get a default distributive reading. Collective readings are also possible.

This rule predicts that in the following sentences the adjective is preferably interpreted distributively (see section 4.5.6.4).

(132) a. Some Sherpas are strong.
b. Five strong Sherpas carried a tent.

Collective readings are not excluded.

(P7) VPs consisting of a copula and an intransitive adjective behave like intransitive verbs, VPs built with transitive adjectives behave like transitive verbs.

(P8) A relative pronoun inherits the distributivity type of the NP it refers to. The collective/distributive preferences are calculated accordingly.

This preference predicts that in the following sentence

(133) John sees at least two climbers who carry a tent.

the relative pronoun *who* gets a default distributive reading since it relates to an NP of type 'd>c' and occurs in subject position of a transitive verb. In contrast, the NP *at least two men* itself gets a default collective reading since it occurs as a direct object.

(P9) Bare Plurals
Bare plurals prefer an existential interpretation with a "one or more" interpretation.

This is a rather simplistic assumption since I have not included an algorithm to also calculate universal or generic readings of bare plurals. This is a complex issue and therefore was not pursued in this thesis. See e.g. Izzo (1993) for the disambiguation of bare plurals.

(P10) The overriding of the default interpretation is more likely in positions that have a higher value on the scale of grammatical functions (see (126)).

The rule predicts that for example, reinterpretation of the subject noun phrase is less costly than reinterpretation of the indirect object, and the reinterpretation of the indirect object is less costly than the reinterpretation of the direct object.

(P11) The more default readings are overridden the less plausible is the corresponding reading.

The above rule (P11) has the effect that in a sentence like

(134) Two men tell a story to several children.

both NPs get a preferred collective reading, the next plausible reading is a distributive interpretation of the subject, followed by a reading where the subject is interpreted collectively, while *several children* is interpreted distributively and outscopes *a story* (see below for more details of this calculation).

(P12) Scope Final NPs
Individual denoting plural NPs occurring in scope final position are less likely interpreted distributively than in non scope final position.

The rule predicts that in sentence (134) the narrow scope distributive interpretation of *several children* is less likely than the distributive interpretation of several children in a sentence having the following structure:

(135) Two men tell several children a story.

The preference is a simplified version of the observation that collective/distributive ambiguities occur more likely if an indefinite or a quantifying noun phrase is in the scope of the non-quantifying plural noun phrase.

I have not included numeric preferences of the type suggested by Aone (1991) (see pp. 229 above). According to these preferences a collective reading for the sentence

(136) Five boys ate three apples.

was dispreferred because the number of boys is larger than the number of apples. Preferences of this type require too much knowledge about the semantics of verbs and the semantics of the participating objects.

Scoping Preferences

In a first step scoping preferences and collective/distributive preference rules are applied independently.

(P13) The relative scope preferences of a set of noun phrases can be calculated by comparing the scope values of each pair of noun phrases in the sentence.

(P14) The relative scope values of two noun phrases are calculated on the basis of the surface order of the noun phrases, the grammatical function, the syntactic structure and the position on the scoping scale of each noun phrase.

(P15) The values for grammatical function, syntactic structure and scoping scale are calculated using the values suggested by Pafel (1988) and repeated in Figure 6 on page 264. The value for the ordering is calculated as follows: if the preceding NP is the subject it gets the ordering value 6, if the preceding NP is the direct, indirect or prepositional object it gets the ordering value 4.

In the sentence

(137) A man tells a story to every child.

the desired scoping is not fully clear. The current algorithm predicts that there is a preference for *every child* to outscope *a story*, and a preference for *a man* to outscope the other two NPs in the sentence. In full natural language there are several techniques to indicate the desired scope ordering more clearly, for example you can use fronting mechanisms to give an NP (or other scope bearing elements) wide scope. For example in

(138) There is a story that a man tells to every child.

a story has wide scope over the other NPs in the sentence. Also in the “logician’s English” sentence

(139) For every child a man tells a story to the child.

the NP *every child* has wide scope over the other NPs. To model this behaviour I add the following scoping rule:

(P16) Fronted NPs always outscope succeeding non-definite NPs.

In my algorithm I have not yet included vertical scopings. I will still add the following preference:

(P17) Complex Noun Phrases
A noun phrase in a prepositional phrase complement of a relational head noun usually outscopes the head noun.

This preference predicts that the sentence

(140) A customer enters a card of every bank.

gets a default interpretation where *every bank* outscopes *a card*. Similarly, in

(141) A customer enters every card of a bank.

the preference (P17) predicts that *a card* preferably outscopes *every card*.

5.4.3 Algorithm and Implementation

The following sections describe how the disambiguation information and the disambiguation rules and preferences are practically realized within the DRoPs disambiguation algorithm. To explain the algorithm I use the following example:

(142) Two men tell a story to several children.

Without a disambiguation component the sentence is predicted to have 6 readings due to different scopings of the 3 NPs and for each of these six different scopings there are 4 different possibilities due to collective/distributive ambiguities (2 plural NPs). This results in 24 theoretically possible readings where, however, some of the readings are logically equivalent, viz. the 6 different scopings where all NPs are interpreted collectively. This reduces the number of different possible readings for (142) to 24–5 = 19. It is evident that a practical system cannot work with such a huge number of possible readings. My disambiguation algorithm will reduce the number of readings to only 4 plausible readings where one reading is predicted to be the “best” reading and the other readings can be generated on demand. The reduction of readings is described in the following sections.

Underspecified Unscoped Logical Form

Every sentence is parsed and translated into a first underspecified representation that has the following simplified structure

(143) `[Matrix, Store]`

where `Matrix` is of the form `[drs(UMatrix,ConMatrix)]` and corresponds to the DRS for the non-scope bearing parts of a sentence. `Store` is a list of terms `np(Features)` or `neg` for VP negation. In the term `np(Features)` the expression `Features` is instantiated to the disambiguation information collected during parsing as introduced in section 5.3.2. In a next step a first instantiation of default collective/distributive readings is applied according to preferences (P1)-(P4) in Table 14 on page 257 above. In our case the first NP *two men* is of type ‘`c>d`’ and occurs in subject position of a ditransitive verb and therefore is assigned a default collective reading. The same default is set for *several children*. Since *a story* is a singular NP its interpretation is `fix`, and abbreviated with `coll`. Figure 5 on page 262 shows this first partially specified underspecified logical form. After the (defeasible) preferences for collective/distributive ambiguities are set the scoping algorithm is applied.

```
Store = [
     [drs([A],[structure(A,event),predicate(A,tell_to,B,C,D)])],
     [np(syn:index:B &
         drs:out:drs([B,A6],[structure(B,group),
                     quantity(B,cardinality,A6,count_unit),
                     value(A6,eq,2),drs([B6],[structure(B6,atomic),
                     part_of(B6,B)])=>drs([],[object(B6,man)])]) &
         sem:quant:quant_type:card &
                   quantity:value:2 &
                            num_rel:eq &
                   disambig:default:dist_type:'c>d' &
                            reading:result:coll &
                                    stability:pref &
                            gra_fct:subj &
                            subcat:dv &
                            syn_struct:noun &
                            scop_lex:2 &
                            scope_value:local:10 &
                   monotone:up &
             string:[two,men] &
             v_info:default:dist_type:m &
             log_rel:man
        ),
      np(syn:index:C &
         drs:out:drs([C,O3],[structure(C,atomic),
                     quantity(C,cardinality,O3,count_unit),
                     value(O3,eq,1),object(C,story)]) &
         sem:quant:quant_type:indef &
                   quantity:value:1 &
                            num_rel:eq &
                   disambig:default:dist_type:c &
                            reading:result:coll &
                                    stability:fix &
                            gra_fct:do &
                            subcat:dv &
                            syn_struct:noun &
                            scop_lex:1 &
                            scope_value:local:6 &
                   monotone:up &
             string:[a,story] &
             v_info:default:dist_type:m &
             log_rel:story
        ),
      np(syn:index:D &
         drs:out:drs([D,B1],[structure(D,group),
                     quantity(D,cardinality,B1,count_unit),
                     value(B1,geq,2),drs([C1],[structure(C1,atomic),
                     part_of(C1,D)])=>drs([],[object(C1,child)])]) &
         sem:quant:quant_type:indef &
                   quantity:value:3 &
                            num_rel:geq &
                   disambig:default:dist_type:'c>d' &
                            reading:result:coll &
                                    stability:pref &
                            gra_fct:io &
                            subcat:dv &
                            syn_struct:noun &
                            scop_lex:2 &
                            scope_value:local:9 &
                   monotone:up &
             string:[several,children] &
             v_info:default:dist_type:m &
             log_rel:child
        )]]
```

Figure 5 Partially Specified Unscoped Logical Form for '*Two men tell a story to several children*.'

Scope Disambiguation (Partially Specified Scoped Logical Forms)

In the next step plausible scope ambiguities are calculated from this unscoped underspecified store. More concretely, plausible scope reorderings are calculated. The scopings are calculated using a modification of an implementation proposed by Pafel (1988). Each pair of NPs in the store is compared and their relative scopes are determined. I use a similar formula like Pafel (1988), however, the ordering values are set differently. See (37) in section 5.2.2.2. The determination of the scope value of a noun phrase uses the parameters listed in Figure 6 on page 264.

More concretely, for each pair of scope bearing elements in the store the following calculations are made. First, for each NP within a scope pair its scope value is calculated as the sum of the values for grammatical function, distributivity type and syntactic structure. Next, to determine relative scopes the preceding NP of each pair gets an additional ordering value. If the preceding NP is in subject position the scope value is raised by 6, if the preceding NP occurs in another grammatical function the scope value is raised by 4. If the NP is not preceding the ordering value is 0. This calculation is summarized as:

(144) Relative Scope Value:
linear order + grammatical function + distributivity + syntactic structure

Like in (38) on page 223 above I assume a pair of NPs, NP_1 and NP_2, to be *not* ambiguous if (P18) in Figure 6 on page 264 holds.

When possible scopings are calculated each NP of the store is compared to all other NPs of the store. The program calculates the local scope value (`scope_value:local`) according to the values in Figure 6 on page 264. The preference rule (P18) decides whether the NP pair is scope ambiguous or not. If the NPs in the NP pair are scope ambiguous with respect to each other both orderings are further processed, otherwise only the ordering reflecting the relative scope is kept. As a result of these calculations the program generates from the underspecified store as shown in Figure 5 on page 262 a list of NP pairs where each NP pair is ordered according to scope. From this list of ordered NP pairs the program generates as many lists of scope-ordered NP pairs as the sentence has scoped readings. Basically, each of these different lists of NP pairs corresponds to a strict order (i.e. an irreflexive, transitive and asymmetric relation) on the set of NPs in the sentence. The greatest element has widest scope and the lowest element has narrowest scope. In our example from the original underspecified store in Figure 5 on page 262 here abbreviated as

(145) Original underspecified store (abbreviated)

```
[
  np(sem:string:[two,men] & >scope_value:local:10),
  np(sem:string:[a,story] & >scope_value:local:6),
  np(sem:string:[several,children]& >scope_value:local:9)
]
```

two lists of NP pairs are generated as shown in (146) and (147) given in Figure 7 on page 266 The feature `>scope_value:local:10` abbreviates the full path `sem:quant:disam-`

Scope Parameters

(P18) A pair of NPs is *not* scopally ambiguous iff

a. scopevalue(NP1) > scopevalue(NP2), and
b. |scopevalue(NP1) – scopevalue(NP2)| > 5
or
|scopevalue(NP1) – scopevalue(NP2)| = 5 and |scopevalue(NP1)| > 15

Otherwise both scopings are possible.

(P19) Relative Scope Value:=
linear order + grammatical function + distributivity + syntactic structure

(P20) Linear Order
6 preceding in subject position
4 preceding in other positions
0 non-preceding

(P21) Grammatical Function
5 subject
4 indirect object
3 preposition object
2 direct object

(P22) Distributivity Type
1 ‘c’
1 ‘c>>d’
2 ‘c>d’
3 ‘d>c’
5 ‘d>>c’
5 ‘d’

(P23) Syntactic Structure
5 Restrictor contains NP
3 Restrictor does not contain NP

Collective/Distributive Parameters

(P24) Reinterpretation Costs
–2 subject (subj)
–3 indirect object (io)
–4 preposition object (po)
–6 direct object (do)

(P25) Last Distributive Costs
–3 if `reading:stabilty:unpref`
–2 if `reading:stabilty:pref`

(P26) If the collective/distributive costs are lower than –5 the reading is discarded.

Figure 6 Parameters for Scope and Plural Disambiguation

`big:scope_value:local:10`. Additionally, for each of the two scopings the relative plausibility of the resulting scopings (called scope strength) is calculated. This is done by calculating for each NP pair the difference between the scope value of the preceding NP and the scope value of the succeeding NP and by summing up these values for each NP pair within one plausible scoping. The scoping with the highest sum is then assumed to be the most plausible scoping. This procedure utilizes the features `scope_value:local` and `scope_value:global`. The local scope value stores the sum of the grammatical function, the distributivity type and the syntactic structure, the global scope value additionally adds the relative linear order with respect to the second NP. The resulting two lists of NP pairs that correspond to the two plausible scopings are given as (146) and (147) in Figure 7

From each list of NP pairs the program then calculates a corresponding list of NPs ordered according to scope yielding the two plausible scopings (148) and (149) in Figure 7.

This shows that the scoping algorithm reduces the theoretically possible six different scopings to only two plausible scopings.

However, these two stores are still not fully disambiguated because the collective/distributive ambiguity of the two plural noun phrases has not yet been considered. Within the stores default values are set but possible or plausible alternatives have not yet been calculated. This will be described next.

Collective/Distributive Disambiguation (Fully Specified Logical Forms)

For each plausible scoping in a first step all possible collective/distributive variants are considered. This means for scoping (148) there are four different possibilities for collective distributive readings, since the NP *two men* and the NP *several children* can – in principle – be read collectively or distributively. Not all possibilities are equally plausible and finally not all possibilities will be generated. We start off with default values for the corresponding NPs, in our example these default values are collective for both plural NPs (see the feature `sem:quant:disambig:reading` in figure Figure 5 on page 262). When alternatives are considered each default reading has to be changed into its non-default variant, e.g. for *two men* the setting

(146) Scoping 1: Scope Strength 18

```
[
  [ np(sem:string:[two,men] &>scope_value:local:10 &
         >scope_value:global:16),
    np(sem:string:[several,children] & >scope_value:local:9 &
         >scope_value:global:9
  ]
  ,
  [ np(sem:string:[two,men] &>scope_value:local:10 &
         >scope_value:global:16),
    np(sem:string:[a,story] &>scope_value:local:6 &
         >scope_value:global:6),
  ]
  ,
  [ np(sem:string:[a,story] &>scope_value:local:6 &
         >scope_value:global:10),
    np(sem:string:[several,children] &>scope_value:local:9 &
         >scope_value:global:9),
  ]
]
```

(147) Scoping 2: Scope Strength 16

```
[
  [ np(sem:string:[two,men] &>scope_value:local:10 &
         >scope_value:global:16),
    np(sem:string:[several,children] & >scope_value:local:9 &
         >scope_value:global:9
  ]
  ,
  [ np(sem:string:[two,men] &>scope_value:local:10 &
         >scope_value:global:16),
    np(sem:string:[a,story] &>scope_value:local:6 &
         >scope_value:global:6),
  ]
  ,
  [ np(sem:string:[several,children] &>scope_value:local:9 &
         >scope_value:global:9),
    np(sem:string:[a,story] &>scope_value:local:6 &
         >scope_value:global:10),
  ]
]
```

(148) Scope order corresponding to (146), Scope Strength 18

```
[ np(sem:string:[two,men],
  np(sem:string:[a,story],
  np(sem:string:[several,children]]
```

(149) Scope order corresponding to (147), Scope Strength 16

```
[ np(sem:string:[two,men],
  np(sem:string:[several,children],
  np(sem:string:[a,story]]
```

Figure 7 Two Preferred Scopings of '*Two men tell a story to several children*.'

(150) Default reading for *two men*

$$\left[\texttt{sem:quant:disambig:reading:} \begin{bmatrix} \texttt{result: coll} \\ \texttt{stability: pref} \end{bmatrix} \right]$$

has to be changed into the setting

(151) Non-Default reading for *two men*

$$\left[\texttt{sem:quant:disambig:reading:} \begin{bmatrix} \texttt{result: distr} \\ \texttt{stability: unpref} \end{bmatrix} \right]$$

Each reinterpretation that changes a preferred into an unpreferred reading is "punished" according to the values listed in (P24) in Figure 6 on page 264. As motivated earlier the reinterpretation of the subject is less costly than e.g. the reinterpretation of the direct object. Also, as a result, the more reinterpretations are performed the less plausible is the respective reading. This corresponds to the observation that in a sentence with more than one individual denoting plural NP preferably only one is read distributively. Additionally, the preference (P25) in Figure 6 on page 264 states that the scope final NP is less likely interpreted distributively which is implemented by punishing a scope final distributive NP by the values in (P25).

If the sum of the collective/distributive punishments for one scoping is below a certain value the reading is discarded. Currently, I have set this value to –5, see preference (P26) in Figure 6 on page 264. The setting can, however, be changed: if it is set larger than –5 less readings are generated and, vice versa, if it is smaller than –5 more collective/distributive readings are generated.

In a final step doubly collective readings are removed, i.e. if two collectively read NPs follow each other the reverse scoping is filtered out since it is logically equivalent.

Each fully specified reading is tagged with a reading value which consists of two numeric values indicating the plausibility of the reading:

(152) Reading Value

`[ScopeValue,PluralValue]`

The `ScopeValue` encodes the scope strength and `PluralValue` encodes the collective/distributive punishment. The generated readings are then sorted whereby the scope strength constitutes the stronger sorting criterion. For our example the program will output the four readings shown in Figure 8.

The first reading in Figure 8 corresponds to the most plausible scoping (148), with no reinterpretation of the individual denoting plural NPs. The second reading corresponds again to the scoping (148) with reinterpretation of the subject NP (hence the punishment –2). The third reading corresponds to the scoping (149) with the indirect object reinterpreted (hence the punishment –3). The fourth reading contains two distributively interpreted NPs. If we set the reinterpretation punishment higher, say to –4, the fourth reading would not be generated. There is no scope final distributive reading of *several children* since a distributive reinterpretation would cost 3 points and the last distributive costs would reduce the plausibility by another 3

```
Reading 1 (Reading Value [18,0])

1. [ two men ]-coll, 2. [ a story ]-coll,
3. [ several children ]-coll

Reading 2 (Reading Value [18,-2])

1. [ two men ]-distr, 2. [ a story ]-coll,
3. [ several children ]-coll

Reading 3 (Reading Value [16,-3])

1. [ two men ]-coll, 2. [ several children ]-distr,
3. [ a story ]-coll

Reading 4 (Reading Value [16,-5])

1. [ two men ]-distr, 2. [ several children ]-distr,
3. [ a story ]-coll
```

Figure 8 Predicted Plausible Readings for '*Two men tell a story to several children.*'

points, summing up to –6 which is below the threshold of –5 as stated in preference (P26). The assignment of the values predict that a scope final individual denoting indirect object will never get a distributive reading, unless the threshold is lowered. This parameter setting may be empirically debatable, but is practically useful.

DRS Construction

In a final step the DRS construction component generates from fully specified scoped stores the final DRS representations that were explained in the previous chapter 4. The DRS construction component puts together the locally available DRS information contained in the store and the matrix to one DRS by a schematic composition depending on feature values set in the NP representation of the store, e.g. whether an NP is quantifying or individual denoting, and whether an NP is read collectively or distributively. A DRS construction at this late stage of processing allows a very flexible encoding of *semantic* ambiguities. My DRS construction technique is different from 'DRS-threading' techniques (Johnson and Klein 1986) where the DRS composition is performed directly within the grammar during parsing. This direct approach would make it very difficult to generate different readings from the same syntactic structure since DRS construction would be directly tied to syntactic analysis. In particular, different scopings cannot be easily encoded by this direct threading, and collective/distributive ambiguities would require a multiplication of grammar rules, e.g. each individual denoting plural NP would require a separate grammar rule for the collective and for the distributive reading (this was proposed e.g. in the implementation of Graham 1994). In my approach, collective and distributive DRS representations of NPs are only generated on demand once the reading features are finally set. And this approach requires just *one* NP rule in the grammar that underspecifies the final reading of the NP and just collects locally available DRS information (as shown in Figure 5 on page 262). This approach also supports my view that neither the NPs nor the VPs are ambiguous but that distributivity can be triggered by different factors (see the discussion in section 3.3) and is thus a ‘global’ phenomenon.

Inference Rules

The DRSs will then be input to the reasoning component of DRoPs as will be discussed in the next chapter 6. Within the reasoning component inference rules operate on disambiguated DRSs, that means inference rules do not reduce the number of possible or plausible readings but explain desired inferences or reduce the indeterminacy of a reading. Also, inference rules are not defeasible. Inference rules are implemented as first-order axioms and require for their application an additional reasoning component, in my case the use of a theorem prover. The additional first-order axioms can encode domain-knowledge, mathematical knowledge, plural axioms, meaning postulates for verb types etc. The concrete implementation of the inference rules will be discussed in the next chapter 6.

5.4.4 Evaluation

The suggested algorithm performs a considerable reduction of semantic scope and collective/distributive ambiguities. The algorithm is based on structurally and lexically available information. For the discussed example (142) here repeated as

(153) Two men tell a story to several children.

the number of readings could be reduced from 19 possible readings to 4 plausible readings. Furthermore, the algorithm proposes a best reading that can be used as a first interpretation for further processing. Other readings are available if necessary, e.g. requested by the user. A further advantage is that my algorithm starts off with an underspecified representation. To generate different readings for a semantically ambiguous sentence the grammar has *not* to be re-run The grammar outputs only one underspecified store enhanced with syntactic, semantic and DRS information. From this underspecified store different readings are generated which is computationally not very costly since stores do usually not contain many (plural) NPs per sentence. A further advantage is that DRS construction is delayed until the NP store is fully specified. Within this thesis, I have not investigated further possible applications of my suggested underspecified representation, e.g. whether it would be possible to perform reasoning directly on these representations. This is an issue of further research.

Within this thesis I have not implemented a suitable (plural) anaphora resolution component. Also, my algorithm does not consider vertical scoping mechanisms which would require the introduction of nested stores (see Keller 1988). However, I assume the techniques developed in this thesis to be straightforwardly extensible to nested stores. Also, I have not considered possible re-interpretations of adjectives. Currently, adjectives are only assigned a first default interpretation according to preference (P6). Furthermore, I have not in detail investigated the interaction of scope-bearing elements besides NPs and negation. This would require an extension of the storing concept with other elements and further techniques to decide plausible scopings. Also extensions to more complicated syntactic structures would have to be investigated. A further important question is how the stores can be used to program an anaphora resolution component. A first test showed that the stores constitute a suitable data structure also for anaphora resolution. However, anaphora resolution is a research topic in its own that I have not further pursued here.

The current settings of the disambiguation and the scoping parameters are still of experimental nature. Fine-tuning of the parameters considering larger set of data are desirable. And vice versa the quality of my predictions with respect to larger data would have to be tested more extensively. For both issues I see, however, the problem described above in 5.2.1.1, viz. that it is very difficult to get consistent experimental data on scope and plural disambiguation. Although there has been much research on this problem I still find the issue of detecting and resolving semantic ambiguities by humans not satisfactorily solved. It is problematic to experimentally get consistent disambiguations which I consider as an argument for the necessity to further investigate the usefulness of underspecified representations in general, and – with respect to plurals – the usefulness of the representations developed in this thesis in particular.

A final criticism often raised against parameter based disambiguation techniques of the type suggested here is that the parameters seem more or less “arbitrary” and that there is no theoretical understanding of *why* certain regularities hold. However, since for the practical applications I was considering in this thesis a fully specified logical form was required my main interest was to develop heuristics to generate a “best” fully specified reading with whichever technique turned out to be suitable. Theoretical understanding of why the heuristics lead to the desired result is a different research objective.

Although some of the techniques used in my approach have been discussed in the literature I am aware of no other approach that combines scope and plural disambiguation in a similar detailed and computationally worked out way. Either the approaches focus on scope disambiguation or they deal with collective/distributive ambiguities. The problems are, however, tightly related and for practical applications both issues require a solution. In my approach I assume that scope disambiguation is basically independent of collective/distributive disambiguation. The implications of this assumption need, however, further investigation. The only rule relating scope and plural disambiguation is (P25) where I punish scope final distributive interpretations.

A simplified version of the DRoPs disambiguation implementation is applied in the project Attempto Controlled English which will be described in chapter 7 of this thesis.

6 Automated Reasoning with Plurals

6.1 Overview

Natural language understanding in practical applications requires an appropriate reasoning component. Depending on the application different reasoning strategies are suitable. The strategies range from shallow reasoning processes that involve simple pattern matching to deep logical analyses based on logical deduction. The intended application of this thesis is to provide techniques for a logical analysis of technical texts that describe a problem domain very precisely. Examples are natural language software specifications or medical documentation texts. A logical analysis of these texts allows users to detect inconsistencies, to check whether a sentence can be logically deduced from the text, to answer queries, to detect logical equivalences, or to identify redundancies or incompletenesses.

Many of these logical processes can be modelled by automated theorem proving techniques using the following strategy. To show that a natural language query (`Query_NL`) is the logical consequence of a natural language text (`Text_NL`) a parsing and semantic construction component translates `Text_NL` and `Query_NL` into their equivalent first-order representations `Text_FOL` and `Query_FOL`, the inference component then tries to deduce `Query_FOL` from `Text_FOL` with the help of a standard first-order theorem prover, and then reports the success or failure of the proof – together with a justification – again on the level of natural language.

Building on this strategy I will show in the following sections how reasoning with technical texts containing plurals can be realized within the DRoPs system. The core reasoning component will thereby be founded on existing theorem proving techniques. Since I am mainly interested in the *application* of these existing techniques to the problem of automated reasoning with plurals I will not give a thorough introduction to automated theorem proving in general but presuppose basic knowledge of the relevant concepts. The interested reader will find a good overview of automated theorem proving in Fitting (1996). A very brief but useful introduction with possible applications and further links can be found in an on-line article by Sutcliffe (1999). A recent survey of the application of automated theorem proving within computational semantics can be found in Blackburn et. al. (2001), Blackburn et. al. (1999), ICoS-2 (2000), or in Blackburn and Bos (2000a). In these latter publications theorem proving and model generation are used to solve problems like presupposition projection or anaphora resolution. Although my approach uses similar logical techniques the goals are different: whereas the mentioned approaches use inference techniques for further linguistic processing the genuine purpose of my approach is to perform logical inferences on natural language texts. In my approach, the application of the theorem provers requires fully specified logical forms, as are for example output but the disambiguation component of the DRoPs system. Whether inference techniques could also be applied to support this disambiguation process has not been investigated within this thesis.

In chapter 4, in particular in section 4.2, I argued for a flat first-order representation of plurals. With respect to automated theorem proving the restriction to first-order logic allows us to rely on the correctness, completeness and efficiency of first-order theorem provers and model generators available off-the-shelf. More concretely, for the reasoning component of the DRoPs system I will use an extended version of the model builder Satchmo (Manthey and Bry 1988) and an extension of the theorem prover Otter (McCune 1994); the core extensions to the two applications have been programmed by Norbert E. Fuchs within the Attempto project (see chapter 7 of this thesis) and are described in Fuchs and Schwertel (2002, 2003). I have added to the versions provided by Fuchs a set of first-order axioms modelling the behaviour of plurals and a number of built-in predicates for mathematical and other operations like permutation of sets.

In section 6.2 I will give an overview of the architecture and the basic working of the DRoPs reasoning component. Section 6.3 will explain the extension of the reasoner with auxiliary first-order axioms that are necessary for automated reasoning with plurals. In section 6.4 practical limitations of the current implementation are addressed and promising topics of further research are presented.

6.2 Architecture of the DRoPs Reasoning Component

6.2.1 Requirements

The reasoner used for the DRoPs system is based on the Attempto Reasoner RACE (Reasoning in ACE) that was developed to support automated reasoning within the controlled natural language Attempto Controlled English (ACE) (see chapter 7 of this thesis). In Fuchs and Schwertel (2002, 2003) the core functionality of the reasoner is presented and a set of requirements for the Attempto reasoner is presented. Since the DRoPs reasoner has similar requirements I will briefly list the main requirements here.

Input and Output in Natural Language. To support the user with the automated analysis of texts but not to burden him with technical details, both the input to the reasoning process, and also the results of the reasoning process should be in natural language.

Generate all Proofs. The reasoner should offer the possibility to generate all proofs to give maximal support. This includes finding all answers to a query, detecting more than one possible inconsistency or – by finding more than one proof – helps to show redundancies in a text. More concretely, the reasoner should find all minimal unsatisfiable subsets of sentences of the text, i.e. sets of sentences that are unsatisfiable and all of which strict subsets are satisfiable.

Justification of Proof. The reasoner should provide a justification of a successful proof, either as a trace of the proof or as a report which sentences of a text were used to prove the theorems. Especially the second alternative is relevant for the logical investigation of a technical texts.

No Knowledge about Theorem Proving Required. Many theorem provers allow users to control proofs through options and parameters. Often these options and parameters presuppose detailed knowledge of the structure of the problem, of the internal working of a theorem prover, or of theorem proving in general, that a typical user, e.g. a domain specialist, will not have. Thus, the reasoner should preferably run automatically, and at most expect users to set familiar parameters like a runtime limit, or the number of solutions found.

Integration of Auxiliary Axioms in First-Order Logic. Deductions on natural language texts require additional axioms that are not directly expressed in the texts. For example, for reasoning with plurals additional lattice theoretic axioms have to be added. Other domain-independent axioms concern mathematical knowledge like the comparison between numbers. This domain-independent knowledge is best expressed in auxiliary axioms using the language of first-order logic. Users may even prefer to state some domain knowledge, for instance ontologies, in first-order axioms instead of using natural language text.

Interface to Evaluable Functions and Predicates. Auxiliary first-order axioms, but also natural language texts can refer to functions or predicates, for instance to arithmetic functions or Boolean predicates. Instead of defining these operations in first-order logic it is much more convenient and certainly more efficient to use the evaluable functions and predicates that are provided by the execution environment, e.g. Prolog built-ins.

Combination of Theorem Proving with Model Generation. Theorem provers and model generators complement each other. Unsatisfiability is semi-decidable and can be detected by a correct and complete theorem prover in finite time. Finding a single contradiction – for instance the always false formula $\perp$, or contradictory formulas $\mathtt{F}$ and $\neg\mathtt{F}$ – suffices to show unsatisfiability. If, however, a theorem $\mathtt{T}$ is *not* the logical consequence of axioms $\mathtt{A}$ then $\mathtt{A} \cup \{\neg\mathtt{T}\}$ is satisfiable, i.e. does have a model. Satisfiability is undecidable, while finite satisfiability is semi-decidable. A correct model generator that is complete for finite satisfiability will detect satisfiability in finite time. Finding a single (finite) model suffices to show satisfiability. Thus theorem provers and model generators complement each other. If the problem is unsatisfiable a theorem prover will find a proof while a model generator has to do an exhaustive – and possibly non-terminating – search to find out that there are no models. If the problem is satisfiable and admits finite models then a model generator will find a finite model while a theorem prover must again do an exhaustive – and potentially non-terminating – search to find that there are no contradictions. Finally, if the problem is satisfiable but does only have infinite models, we can encounter non-termination for both theorem provers and model generators – after all satisfiability is undecidable.

Besides complementing theorem provers, model generators that generate (minimal) finite models offer additional advantages (Bos 2001), foremost the possibility to construct comprehensive answers to queries.

6.2.2 The Underlying Theorem Provers

The computational aspects of first-order logic have been intensively investigated by many researchers. To profit from the accumulated experience of these researchers the reasoner RACE, and consequently the DRoPs reasoner, are based on first-order theorem provers and model generators freely available off-the-shelf. While for theorem provers there is a large variety of sophisticated candidates, for model generators the choice is still limited.

Since the above requirements imply extensions and possibly modifications of the selected tools the programs should be locally available. This decision precludes solutions like MathWeb (see e.g. www.mathweb.org) that farms out an inference task simultaneously to theorem provers and model generators available on the internet and then uses the first result returned. In Fuchs and Schwertel (2002) we discuss several candidates and motivate the decision to finally base the reasoner on the theorem prover Otter (McCune 1994) and on the model generator Satchmo (Manthey and Bry 1988).

Otter. McCune describes Otter the following way: "Otter is a resolution-style theorem-proving program for first-order logic with equality. Otter includes the inference rules binary resolution, hyperresolution, UR-resolution, and binary paramodulation. Some of its other abilities and features are conversion from first-order formulas to clauses, forward and back subsumption, factoring, weighting, answer literals, term ordering, forward and back demodulation, evaluable functions and predicates, and Knuth-Bendix completion."

After nearly 20 years of development Otter is very stable, and – due to its implementation in C – very efficient. For the fine-tuning of proofs Otter offers a rich set of options and parameters that can even be modified interactively by users during a proof. This fine-tuning, however, and the possibility of user-interactions during proofs prevent general statements concerning the correctness and completeness of Otter.

Otter accepts first-order clauses, or first-order formulas that are automatically translated into clauses.

Satchmo. Manthey and Bry (1988) write: "Satchmo is a theorem prover consisting of just a few and simple Prolog programs. Prolog may be used for representing problem clauses as well. Satchmo is based on a model generation paradigm. It is refutation complete if used in a level saturation manner."

Satchmo owes its high efficiency to the power of the underlying Prolog inference engine. Satchmo accepts first-order clauses in implication form, or Horn clauses in Prolog notation. Negation is expressed as implication to false.

If the clauses admit a finite model, Satchmo will find it. Satchmo is correct for unsatisfiability if the input clauses are range-restricted – which can always be achieved – and complete for unsatisfiability if used in level-saturation manner – technically achieved with the help of Prolog's set predicates (Bry and Yahya 1996).

In the currently available implementations, Otter and Satchmo are used independently but the idea is to finally combine them into one integrated inference system. Above I stated the requirement that theorem proving and model generation techniques should be combined. Whereas Otter functions only as a theorem prover, Satchmo can be used both as a theorem prover and a model generator. Satchmo functions as a theorem prover if it detects unsatisfiability and alternatively it can be used as a model generator: if a set of Satchmo clauses is satisfiable and admits a finite model then Satchmo will generate a model that is returned as a list of ground instances of atoms.

6.2.3 The Architecture

In Figure 9 on page 276 the architecture of the DRoPs reasoning component is shown. The user inputs natural language text (`Text_NL`) and possibly a natural language query (`Query_NL`) that will be examined by the DRoPs reasoner. The reasoner can check the consistency of the text, alternatively the user can input a query and the reasoner checks whether the query (`Query_NL`) can be logically deduced from the text. Queries can consist of natural language questions or of declarative sentences. Currently, both are – in principle – treated analogously. Both the input text and the query text are input to the DRoPs parser and disambiugator (as discussed in the previous chapter 5), which outputs a fully specified DRS for the text (`Text_DRS`) and for the query (`Query_DRS`). This representation is in fact a a slightly extended version of the discourse representation structures discussed in chapter 4 and is called a *paragraph*.

For example the text

(1) Every company that buys a standard machine gets a discount. A British company buys a standard machine.

will be translated into the following paragraph:

(2) Internal Representation of (1)

```
paragraph(drs([A,B,C,D,E],[drs([F,G,H,I,J],[structure(G,atomic)-1,
quantity(G,cardinality,F,count_unit)-1, value(F,eq,1)-1,
object(G,company)-1, structure(I,atomic)-1, quantity(I,cardinal-
ity,H,count_unit)-1, value(H,eq,1)-1, property(I,standard)-1,
object(I,machine)-1, structure(J,event)-1, predicate(J,buy,G,I)-1])
=> drs([K,L,M],[structure(L,atomic)-1, quantity(L,cardinal-
ity,K,count_unit)-1, value(K,eq,1)-1, object(L,discount)-1, struc-
ture(M,event)-1, predicate(M,get,G,L)-1]), structure(B,atomic)-2,
quantity(B,cardinality,A,count_unit)-2, value(A,eq,1)-2, prop-
erty(B,'British')-2, object(B,company)-2, structure(D,atomic)-2,
quantity(D,cardinality,C,count_unit)-2,value(C,eq,1)-2,prop-
erty(D,standard)-2, object(D,machine)-2, structure(E,event)-2, pred-
icate(E,buy,B,D)-2]), text([‘Every company that buys a standard
machine gets a discount.’,’A British company buys a standard
machine.’])).
```

The structure `paragraph/2` contains as arguments an extended DRS `drs/2` and a representa-

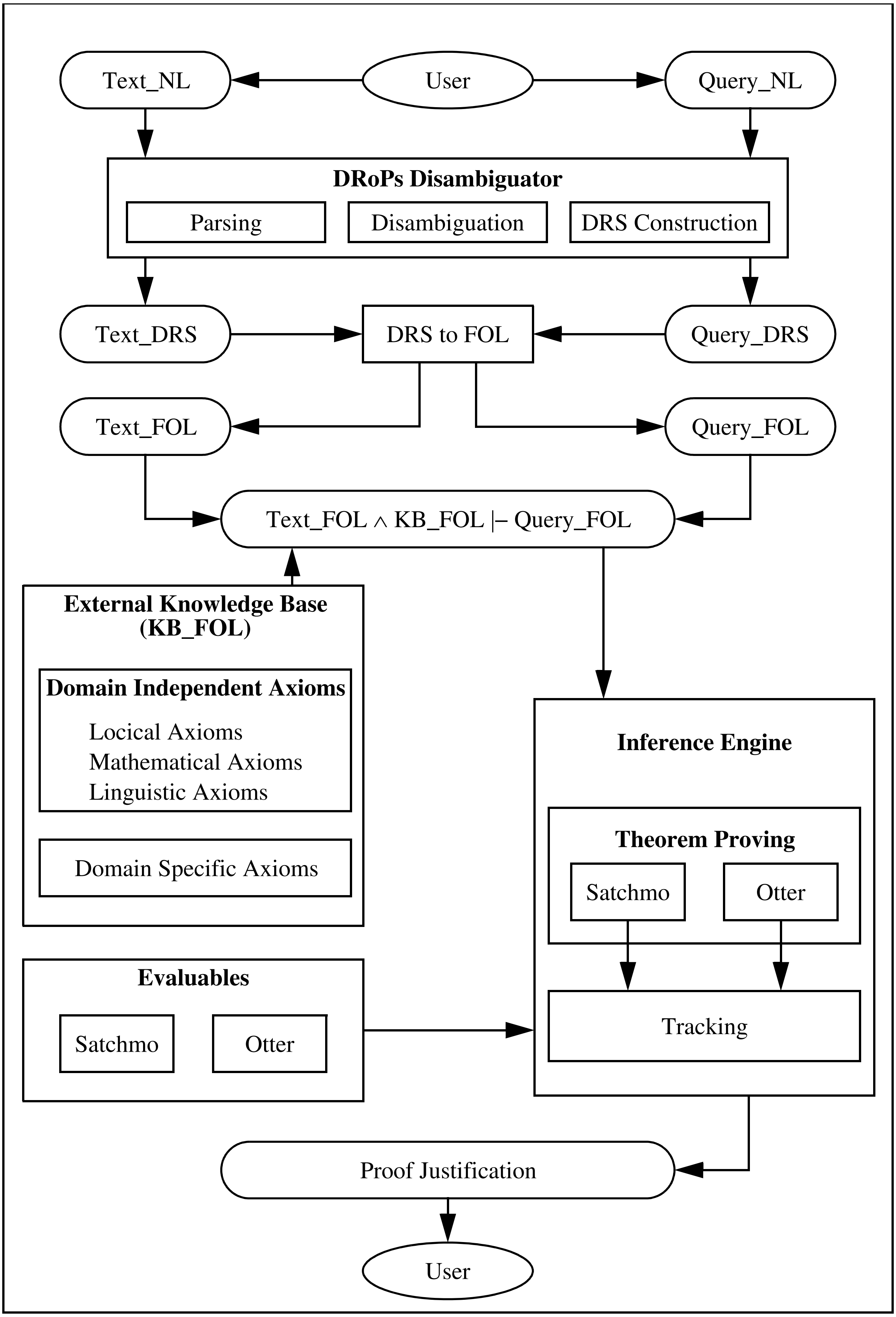


Figure 9 Architecture of the DRoPs Reasoning Component

tion of the input text `text/1`. The structure `text/1` contains a list whose elements are the input sentences represented as character strings. Logical atoms occurring in `drs/2` are actually written as `Atom-I` where the index `I` refers to the `I`'th element of the list in `text/1`, i.e. to the sentence from which `Atom` was derived. This extended DRS notation was introduced because the reasoner generates for each proof a report which subset of the natural language text was used to prove the query. The implementation of this feature relies on the extended version of the discourse representation structures

The `Text_DRS` and a `Query_DRS` are then automatically translated into their first-order equivalents `Text_FOL` and `Query_FOL` using a variation of the program `drs2fol.pl` presented in Blackburn and Bos (2000b). Besides the logical representation of the input text additional knowledge will enter the reasoning process. This external knowledge base (`KB_FOL`) is encoded in first-order logic and consists of domain independent knowledge describing logical, mathematical and linguistic knowledge and – optionally – domain-dependent knowledge. Internally the auxiliary axioms encoding the external knowledge are represented as

```
(3)     fol_axiom(Number,Formula,Text)
```

where `Number` labels the axiom analogous to the index attached to atomic DRS conditions above, `Formula` is a first-order formula, and `Text` is a string describing the axiom verbally. All auxiliary axioms (`KB_FOL`) are conjoined with the first-order formulas derived from the text (`Text_FOL`).

Depending on the desired logical deduction the reasoner checks in a next step either the consistency of the `Text_FOL` and the `KB_FOL` by checking

(4) Consistency

$$\texttt{Text\_FOL} \wedge \texttt{KB\_FOL} \vdash \bot$$

or the reasoner tries – using the same technique – to deduce the `Query_FOL` from the conjunction of the `Text_FOL` with the `KB_FOL`:

(5) Logical Deduction

$$\begin{aligned} & \texttt{Text\_FOL} \wedge \texttt{KB\_FOL} \vdash \texttt{Query\_FOL} \\ \equiv\ & \texttt{Text\_FOL} \wedge \texttt{KB\_FOL} \wedge \neg\texttt{Query\_FOL} \vdash \bot \end{aligned}$$

The FOL representations are then converted into Satchmo and Otter clauses that are passed to Satchmo, respectively to Otter. Satchmo, respectively Otter, is executed, and the output is scanned for the results of the proof that are then reported to the user reusing the original natural language input (see Fuchs and Schwertel 2002, 2003).

Above the requirement was stated that the reasoner should find all proofs. This requirement is solved differently in Satchmo and Otter. To perform the proof in (5) Satchmo proves that $\{\texttt{Text\_FOL}, \texttt{KB\_FOL}\} \cup \{\neg\texttt{Query\_FOL}\}$ is unsatisfiable. The original version of Satchmo is designed to detect the unsatisfiability of an unsatisfiable set of Satchmo clauses as quickly as possible, and Satchmo will stop immediately once it detected that the set is indeed unsatisfiable. The requirement to generate *all* proofs amounts to finding all minimal unsatisfiable subsets of the set of Satchmo clauses. Fuchs has extended Satchmo so that it will find all minimal unsatisfiable subsets of Satchmo clauses, and thus all minimal unsatisfiable subsets of the natural language sentences from which the clauses were derived (Fuchs and Schwertel 2002, 2003). Otter already comes with the functionality to find all proofs of a problem. It provides a parameter `max_proofs` that delimits the number of proofs Otter should try to find. If the parameter is set to –1 then Otter will find as many proofs as it can within other constraints, for instance the setting of the parameter `max_seconds` that delimits Otter's runtime.

Once all results have been found the reasoner reports the result of the proof using again `Text_NL` and `Query_NL`. This tracking uses the indices introduced within the extended DRS and the auxiliary FOL axioms. For more details on the implementation of the tracking mechanism see Fuchs and Schwertel (2002, 2003). Just note that in Otter answer literals (Green 1969) are utilized to perform tracking of a proof, whereas Fuch's modified version of Satchmo collects indices of atoms participating in a proof.

During the proof recourse to evaluable functions and predicates may be necessary, e.g. numeric comparison of two numbers. In Satchmo we can fulfil this requirement by user defined or built-in Prolog predicates. Otter provides the desired functionality less straightforwardly than Satchmo. Otter provides a very limited set of evaluable functions and predicates for integer and floating point arithmetic, Boolean operations and lexical comparisons that allow users to program aspects of the deduction process – for instance list operations – in a style similar to Prolog. To go beyond the limits of Otter's evaluable functions and predicates, there is an option to use Otter's gateway to foreign evaluable functions so that predicates defined externally in Prolog can be accessed. This latter feature is, however, not yet implemented.

Finally, the reasoner outputs the results to the user. This output is different depending on whether the user wants to execute a deduction, asks a query or performs a consistency check.

Assume we have the following natural language input text

(6) `Every company that buys a standard machine gets a discount. A British company buys a standard machine. A French company buys a special machine.`

and the query consisting of a declarative sentence that is to be proved

(7) `A company gets a discount.`

the DRoPs reasoner will prove the query and the user is presented the following output:

(8) Output

```
The reasoner proved that the sentence(s)
  A company gets a discount.
can be deduced from the sentence(s)
  Every company that buys a standard machine gets a discount.
  A British company buys a standard machine.
```

Note that as a consequence of the justification facility we only see the two sentences actually used in the proof.

Given the same three sentences (6) and the natural language question

(9) `Who buys a machine?`

the reasoner generates two answers

(10) Output

```
The reasoner proved that the query (-ies)
  Who buys a machine?
can be answered on the basis of the sentence(s)
  A British company buys a standard machine.

The reasoner proved that the query(-ies)
  Who buys a machine?
can be answered on the basis of the sentence(s)
  A French company buys a special machine.
```

All possible answers are generated, and for each answer we see only the sentences used to derive that answer.

Similarly we can check the consistency of natural language text. If the text is inconsistent, the reasoner will identify *all* its inconsistent subsets. Given the text

(11) `Every company that buys a standard machine gets a discount. A British company buys a standard machine. A French company buys a standard machine. There is no company that gets a discount.`

we get the two results

(12) Output

```
The reasoner proved that the sentence(s)
  Every company that buys a standard machine gets a discount.
  A French company buys a standard machine.
  There is no company that gets a discount.
are inconsistent.

The reasoner proved that the sentence(s)
  Every company that buys a standard machine gets a discount.
  A British company buys a standard machine.
  There is no company that gets a discount.
are inconsistent.
```

showing that the text contains two inconsistent subsets.

The preceding examples demonstrated the basic usage of the reasoner. More advanced deductions are necessary when the sentences contain plurals. These deductions make use of auxiliary first-order axioms and evaluable functions as will be shown in the next section.

6.3 Automated Reasoning with Plurals

6.3.1 Introduction

Plural constructions in natural language often trigger complex inferences. This was already shown in chapter 4. Not all of these inferences are possible on the basis of the semantic representation of the natural language input alone but the inferences need additional external knowledge. There are for example inferences triggered by mathematical knowledge, or inferences that are induced by our linguistic knowledge about the structure and interpretation of plural constructions. Additional knowledge is supplied by our knowledge about the meaning of certain lexical items. In the following sections I will mainly describe how linguistic and mathematical inferences can be modelled by extending the DRoPs reasoning component with auxiliary domain-independent first-order axioms for lattice-theory, equality and integer arithmetic. This includes the integration of evaluable functions and predicates.

The introduction of auxiliary first-order axioms necessarily increases the search space. However – as Fuchs and Schwertel (2002, 2003) noticed during their investigations – the larger search space does not inevitably result in longer runtimes of the reasoners. This is most probably due to the efficient implementations of Otter and Satchmo. After all, as I have addressed in section 4.2.3, to overcome the limited expressive power of first-order logic it is necessary to assume richer ontologies which then requires the introduction of additional axioms that constrain how the newly introduced entities behave. This means, to achieve sufficient expressive power we have no alternative than to introduce additional axioms.

6.3.2 Types of Axioms

There are several types of auxiliary axioms that I have added to the DRoPs reasoner. Currently, only domain independent axioms are considered but it is also possible to add domain-dependent axioms. There are several types of domain-independent axioms: lattice theoretic axioms describing the plural domain, logical axioms including equality axioms, mathematical axioms performing simple arithmetic operations, meaning postulates for certain words or word classes, e.g. meaning postulates for non-standard quantifiers or meaning-postulates describing inferences on certain verb-classes. In chapter 4 I have listed most of the currently implemented auxiliary axioms of the DRoPs system. All these axioms are repeated here in Figure 10, Figure 11 and Figure 12. I will not give detailed comments here but refer to the respective sections of chapter 4. Here, I am mainly concerned with how these axioms are actually integrated into the DRoPs system.

6.3.2.1 Lattice Theoretic Axioms

Various axiom systems governing the behaviour of plurals in natural language have been proposed (Link 1998b, e.g. chapters 2 and 6; Kamp and Reyle 1993 and others). For the practical implementation I had to settle with an axiom system that provides a good trade-off between empirical adequacy and computational tractability. The implemented lattice theoretic axioms can be found in Figure 10 – Figure 12. I will give one example of the practical implementation.

From the natural language text

(13) `Every company that buys a machine gets a discount. Six Swiss companies each buy a machine.`

Type Axioms

(Ax. 1)	$\forall$X(structure(X,group) $\rightarrow$ structure(X,dom))	
(Ax. 2)	$\forall$X(structure(X,atomic) $\rightarrow$ structure(X,dom))	
(Ax. 3)	$\forall$X(structure(X,mass) $\rightarrow$ structure(X,dom))	
(Ax. 4)	$\forall$X(structure(X,event) $\rightarrow$ structure(X,e_dom))	
(Ax. 5)	$\forall$X(structure(X,state) $\rightarrow$ structure(X,e_dom))	

Part-Of Relation

(Ax. 6)	$\forall$X(part_of(X,X))	reflexivity
(Ax. 7)	$\forall$X$\forall$Y$\forall$Z(part_of(X,Y) $\wedge$ part_of(Y,Z) $\rightarrow$ part_of(X,Z))	transitivity
(Ax. 8)	$\forall$X$\forall$Y(part_of(X,Y) $\wedge$ part_of(Y,X) $\rightarrow$ is_equal(X,Y))	anti-symmetry
(Ax. 9)	$\forall$X$\forall$Y(proper_part_of(X,Y) $\rightarrow$ part_of(X,Y) $\wedge$ $\neg$ is_equal(X,Y))	
(Ax. 10)	1. $\forall$X$\forall$C(structure(X,group) $\wedge$ quantity(X,cardinality,C,count_unit) $\wedge$ value(C,geq,2) $\rightarrow$ $\exists$Y$\exists$Q(structure(Y,atomic) $\wedge$ proper_part_of(Y,X) $\wedge$ quantity(Y,cardinality,Q,count_unit) $\wedge$ value(Q,eq,1))	
	2. $\forall$X(structure(X,group) $\rightarrow$ $\exists$Y(proper_part_of(Y,X) $\wedge$ structure(Y,atomic)))	
(Ax. 11)	$\forall$X(structure(X,atomic) $\rightarrow$ $\forall$Y(part_of(Y,X) $\rightarrow$ is_equal(X,Y)))	

Equality

(Ax. 12)	$\forall$X(is_equal(X,X))	reflexivity
(Ax. 13)	$\forall$X$\forall$Y$\forall$Z(is_equal(X,Y) $\wedge$ is_equal(Y,Z) $\rightarrow$ is_equal(X,Z)))	transitivity
(Ax. 14)	$\forall$X$\forall$Y(is_equal(X,Y) $\rightarrow$ is_equal(Y,X))	symmetry
(Ax. 15)	1. $\forall$X$\forall$Y$\forall$O(object(X,O) $\wedge$ is_equal(X,Y) $\rightarrow$ object(Y,O))	
	2. $\forall$X$\forall$Y$\forall$P(property(X,P) $\wedge$ is_equal(X,Y) $\rightarrow$ property(Y,P))	
	3. …	

Mass Domain

(Ax. 16)	$\forall$X$\forall$Y$\forall$O(structure(X,mass) $\wedge$ object(X,O) $\wedge$ proper_part_of(Y,X) $\rightarrow$ object(Y,O))	
(Ax. 17)	$\forall$W$\forall$D$\forall$Q1$\forall$U(structure(W,mass) $\wedge$ quantity(W,D,Q1,U) $\rightarrow$ $\exists$P$\exists$Q2(structure(P,mass) $\wedge$ proper_part_of(P,W) $\wedge$ quantity(P,D,Q2,U)))	

Figure 10 Auxiliary FOL-Axioms for Reasoning with Plurals 1/3

Existence and Quantity of Parts

(Ax. 18) $\forall X \forall C$(structure(X,dom) $\wedge$ quantity(X,cardinality,C,count_unit) $\wedge$ value(C,geq,1)
$\rightarrow \exists Y \exists Q$(structure(Y,atomic) $\wedge$ part_of(Y,X) $\wedge$
quantity(Y,cardinality,Q,count_unit) $\wedge$ value(QP,eq,1))

(Ax. 19) $\forall X \forall C$(structure(X,group) $\wedge$ quantity(X,cardinality,C,count_unit) $\wedge$
value(C,greater,2)
$\rightarrow \exists Y \exists Q$(structure(Y,group) $\wedge$ proper_part_of(Y,X) $\wedge$
quantity(Y,cardinality,Q,count_unit))

(Ax. 20) $\forall X \forall C \forall N$(structure(X,group) $\wedge$ quantity(X,cardinality,C,count_unit) $\wedge$
value(C,geq,N)
$\rightarrow \exists Y \exists Q$(structure(Y,group) $\wedge$ part_of(Y,X) $\wedge$
quantity(Y,cardinality,Q,count_unit) & value(Q,eq,N))

(Ax. 21) 1. $\forall W \forall Q1 \forall N \forall P \forall Q2$(quantity(W,cardinality,Q1,count_unit) $\wedge$
value(Q1,leq,N) $\wedge$ part_of(P,W) $\wedge$ quantity(P,cardinality,Q2,count_unit)
$\rightarrow$ value(Q2,leq,N))

2. $\forall W \forall Q1 \forall N \forall P \forall Q2$(quantity(W,cardinality,Q1,count_unit) $\wedge$
value(Q1,less,N) $\wedge$ part_of(P,W) $\wedge$ quantity(P,cardinality,Q2,count_unit)
$\rightarrow$ value(Q2,less,N))

Numeric Relations

(Ax. 22) 1. $\forall V \forall N$(value(V,eq,N) $\rightarrow$ (value(V,leq,N) $\wedge$ value(V,geq,N)))

2. $\forall V \forall N$(value(V,leq,N) $\rightarrow \neg$ value(V,greater,N))

3. $\forall V \forall N$(value(V,less,N) $\rightarrow \neg$ value(V,geq,N))

4. …

(Ax. 23) 1. $\forall N \forall M \forall N1 \forall M1$(value(N,leq,M) $\wedge$ value(N1,leq,M1) $\wedge$ relation(N,leq,N1)
$\rightarrow$ value(N,leq,M1))

2. $\forall N \forall M \forall N1 \forall M1$(value(N,leq,M) $\wedge$ value(N1,less,M1) $\wedge$ relation(N,leq,N1)
$\rightarrow$ value(N,less,M1))

3. $\forall N \forall M \forall N1 \forall M1$(value(N,geq,M) $\wedge$ value(N1,leq,M1) $\wedge$ relation(N,leq,N1)
$\rightarrow$ value(N,leq,M1))

4. $\forall N \forall M \forall N1 \forall M1$(value(N,geq,M) $\wedge$ value(N1,less,M1) $\wedge$ relation(N,leq,N1)
$\rightarrow$ value(N,less,M1))

Figure 11 Auxiliary FOL-Axioms for Reasoning with Plurals 2/3

Non-Standard Quantifiers

(Ax. 24) $\forall N \forall N1(\neg(\text{property}(N1,\text{few}) \wedge \text{relation}(N,\text{leq},N1)) \wedge \text{property}(N,\text{many})))$

(Ax. 25) $\forall W \forall Q1 \forall P \forall Q2(\text{quantity}(W,\text{cardinality},Q1,\text{count\_unit}) \wedge \text{property}(Q1,\text{few}) \wedge \text{part\_of}(P,W) \wedge \text{quantity}(P,\text{cardinality},Q2,\text{count\_unit}) \rightarrow \neg\text{property}(Q2,\text{many}))$

(Ax. 26) $\forall N \forall M(\text{relation}(N,\text{most},M) \rightarrow \text{relation}(N,\text{most\_of},M))$

(Ax. 27) $\forall N \forall M(\text{relation}(N,\text{most\_of},M) \rightarrow \text{relation}(N,\text{most},M))$

(Ax. 28) $\forall N \forall M(\text{property}(N,\text{all}) \rightarrow \text{relation}(N,\text{most},N))$

(Ax. 29) $\forall N \forall M(\text{relation}(N,\text{most},M) \rightarrow \text{relation}(N,\text{more\_than\_half\_of},N))$

(Ax. 30) $\forall N \forall M(\text{relation}(N,\text{most},M) \rightarrow \text{relation}(N,\text{many\_of},N))$

(Ax. 31) $\forall N \forall M \forall Z(\neg(\text{relation}(N,\text{few\_of},Z) \wedge \text{relation}(M,\text{leq},N) \wedge \text{relation}(M,\text{many\_of},Z)))$

(Ax. 32) $\forall W \forall Q1 \forall P \forall Q2 \forall Z(\neg(\text{quantity}(W,\text{cardinality},Q1,\text{count\_unit}) \wedge \text{quantity}(P,\text{cardinality},Q2,\text{count\_unit}) \wedge \text{part\_of}(P,W) \wedge \text{property}(Q1,\text{few\_of},Z) \wedge \text{property}(Q2,\text{many\_of},Z)))$

(Ax. 33) $\forall N \forall M(\neg(\text{property}(N,\text{few}) \wedge \text{relation}(M,\text{leq},N) \wedge \text{property}(M,\text{many})))$

(Ax. 34) $\forall W \forall Q1 \forall P \forall Q2(\neg(\text{quantity}(W,\text{cardinality},Q1,\text{count\_unit}) \wedge \text{quantity}(P,\text{cardinality},Q2,\text{count\_unit}) \wedge \text{part\_of}(P,W) \wedge \text{property}(Q1,\text{few}) \wedge \text{property}(Q2,\text{many})))$

Quantity of Parts

(Ax. 35)
1. $\forall W \forall Q1 \forall N \forall P \forall Q2 \forall C \forall U(\text{quantity}(W,C,Q1,U) \wedge \text{value}(Q1,\text{leq},N) \wedge \text{part\_of}(P,W) \wedge \text{quantity}(P,C,Q2,U) \rightarrow \text{value}(Q2,\text{leq},N))$
2. $\forall W \forall Q1 \forall N \forall P \forall Q2 \forall C \forall U(\text{quantity}(W,C,Q1,U) \wedge \text{value}(Q1,\text{less},N) \wedge \text{part\_of}(P,W) \wedge \text{quantity}(P,C,Q2,U) \rightarrow \text{value}(Q2,\text{less},N))$

Modification Types

(Ax. 36)
1. $\forall E \forall P \forall L(\text{modifier}(E,\text{location},P,L) \rightarrow \text{modifier}(E,\text{where},P,L))$
2. $\forall E \forall P \forall L(\text{modifier}(E,\text{time},P,L) \rightarrow \text{modifier}(E,\text{when},P,L))$
3. $\forall E \forall P \forall L(\text{modifier}(E,\text{origin},P,L) \rightarrow \text{modifier}(E,\text{from-where},P,L))$
4. $\forall E \forall P \forall L(\text{modifier}(E,\text{duration},P,L) \rightarrow \text{modifier}(E,\text{how-long},P,L))$
5. …

(Ax. 37) $\forall E \forall H \forall P \forall L(\text{modifier}(E,H,P,L) \rightarrow \text{modifier}(E,\text{how},P,L))$

Figure 12 Auxiliary FOL-Axioms for Reasoning with Plurals 3/3

we want to infer

(14) `A company gets a discount.`

To perform this inference we need to deduce from the second sentence the existence of a company that buys a machine. The extended DRS representation of the second sentence is (13).

(15) Representation of second sentence of (13)

```
paragraph(
drs([A,B],[structure(B,group)-2,
quantity(B,cardinality,A,count_unit)-2,value(A,eq,6)-2,
drs([C],[structure(C,atomic)-2,part_of(C,B)-2])=>
drs([],[object(C,company)-2,property(C,'Swiss')-2]),
drs([D],[structure(D,atomic)-2,part_of(D,B)-2])=>
drs([E,F,G],[structure(F,atomic)-2,
quantity(F,cardinality,E,count_unit)-2,value(E,eq,1)-2,
object(F,machine)-2,structure(G,event)-2,predicate(G,buy,D,F)-2])]),
text(['...','Six Swiss companies each buy a machine.']))
```

As discussed in chapter 4 the representation for plurals assumes a lattice-theoretic structure of the domain of discourse partially ordered by the relation `part_of/2`. Additionally, it is assumed that for any subset `S` of the domain there exists a unique least upper bound (supremum) of the elements of `S` with respect to `part_of/2`. Thus, apart from atomic individuals (atoms) there are complex individuals (groups) formed by the supremum operation which serve as the denotation of plural nouns. Recall that the advantage of this approach is, that both atoms and groups have the same elementary ontological type; both are individuals of the domain of discourse. This avoids recourse to higher-order objects like sets as denotations for plural expressions. In the above representation each object variable is typed according to its position in the lattice. Lines 2, 4 and 5 of the structure express that there is a group `A` the atomic parts of which are Swiss companies, line 3 that the cardinality of `A` equals 6, and lines 6 to 9 express the distributive reading triggered by the cue word *each*.

Since from this representation the existence of a company that buys a machine cannot be directly deduced additional axioms are necessary. In particular an axiom that states that groups consist of atomic parts is needed which is given by (Ax. 10)-1 and internally represented as:

(16) Internal representation of axiom (Ax. 10)

```
fol_axiom(101,
forall([A,B],structure(A,group) & quantity(A,cardinality,B,count_unit)
& value(B,geq,2)=>exists([C,D],structure(C,atomic) &
proper_part_of(C,A)&quantity(C,cardinality,D,count_unit) &
value(D,eq,1))),
'(Ax. 10-1): Every group consists of atomic parts.').
```

The first argument of `fol_axiom/3` serves as an index to identify the FOL axioms used for a proof. The second argument codes the logical axiom in standard FOL syntax, and the last argument contains textual information that is returned during tracking.

Additionally, the lattice-theoretic axiom (Ax. 9) and the number axiom (Ax. 22)-1 are needed which are internally represented as:

(17) Internal representation of axiom (Ax. 9)

```
fol_axiom(9,
forall([A,B],proper_part_of(A,B)=>part_of(A,B) & -(is_equal(A,B))),
'(Ax. 9): Definition of proper_part_of.').
```

(18) Internal representation of axiom (Ax. 22)-1

```
fol_axiom(221,forall([A,B],value(A,eq,B)=>value(A,leq,B) &
value(A,geq,B)),'(Ax. 22-1): Number Axiom.').
```

Note, that the axioms are not domain-specific since they model the meaning of plurals in natural language or describe the order of natural numbers. Hence the axioms have to be available for each proof.

Calling the reasoner with the conjunction of the clauses derived from the natural language text (13) and from the auxiliary axioms (Ax. 10)-1, (Ax. 9) and (Ax. 22)-1 we get the desired deduction and the reasoner will output:

(19) Output

```
The reasoner proved that the sentence(s)
  A company gets a discount.
can be deduced from the sentence(s)
  Every company that buys a machine gets a discount.
  Six Swiss companies each buy a machine.
using the auxiliary axiom(s)
  (Ax. 9): Definition of proper_part_of.
  (Ax. 10-1): Every group consists of atomic parts.
  (Ax. 22-1): Number Axiom.
```

The reasoner includes other lattice-theoretic axioms, e.g. the reflexivity, transitivity and antisymmetry of the part-of relation encoded in (Ax. 6), (Ax. 7) and (Ax. 8), or axiom (Ax. 11) which states that atoms do not have proper parts. The reverse direction of (Ax. 11), viz. if an object does not have proper parts it is an atom, is not implemented since it explodes the size of the search space. Also, the existence of a supremum is not axiomatized. These restrictions are not harmful since I have not (yet) found empirical needs for the introduction of the axioms. Note, that since Satchmo requires domain restriction the axiom (Ax. 6) is actually internally formalized using domain restricted formulae. More concretely the axiom is stated as:

(20) Domain Restricted Implementation of (Ax. 6)
$\forall$X(structure(X,dom) $\rightarrow$ part_of(X,X))

The same holds for the implementation of equality. See chapter 4 for more explanations concerning the individual axioms.

6.3.2.2 Logical Axioms – Equality

Many inferences require the interaction of several auxiliary axioms whereby equality plays an important role. Given the natural language text (13) on page 280 and asking the natural language question

(21) `Who buys machines?`

we expect to retrieve the second sentence of (13), viz.

(22) `Six Swiss companies each buy a machine.`

since the bare plural *machines* in the question is indeterminate as to whether one or more machines are bought. As discussed in section 4.5.7 I model this by representing both the query word *who* and the bare plural *machines* as underspecified with respect to the position in the lattice (`structure(V,dom)`). The query representation is

(23) Representation of (21)

```
paragraph(
drs([A,B,C],[query(A,who)-1,structure(A,dom)-1,
structure(B,dom)-1,
   drs([D],[structure(D,atomic)-1,part_of(D,B)-1])
   =>
   drs([],[object(D,machine)-1]),
predicate(C,event,buy,A,B)-1]),
text(['Who buys machines?'])).
```

Besides the auxiliary axioms (Ax. 2), (Ax. 9), (Ax. 10)-2, (Ax. 11) and (Ax. 22)-1 the deduction additionally requires the equality axiom (Ax. 15)-1 triggered in particular by (Ax. 11). The equality axiom (Ax. 15)-1 is internally represented as

(24) Internal representation of (Ax. 15)-1

```
fol_axiom(151,
forall([X,Y,P],(is_equal(X,Y) & object(X,P) => object(Y,P))),
'(Ax. 15-1): Identical objects have the same properties.')
```

The relation `is_equal/2` models equality and is defined as reflexive, symmetric and transitive via the auxiliary axioms (Ax. 12), (Ax. 13) and (Ax. 14). The equality substitution axioms (Ax. 15) can be formalized directly in first-order logic due to the flat-notation.

Defining equality in this way may seem naïve but there are several reasons for this decision. Although Otter does provide a more sophisticated treatment of equality using paramodulation and demodulation the problem is that paramodulation cannot be effectively controlled in Otter's autonomous mode which is required for the DRoPs reasoner. Using the defined relation `is_equal/2` instead of Otter's built-in equality avoids triggering paramodulation, and turned out to be a viable – though perhaps debatable – approach. What is more, Satchmo has no built-in theory of equality using paramodulation and demodulation which is why – at least for Satchmo – there is no alternative than to use the defined relation `is_equal/2`.

Note again, that in Satchmo the axiom (Ax. 12) is actually implemented as

(25) Internal Representation of (Ax. 12) in Satchmo

1. $\forall$A(structure(A,dom) $\rightarrow$ is_equal(A,A))
2. $\forall$A(structure(A,event) $\rightarrow$ is_equal(A,A))
3. $\forall$A(structure(A,state) $\rightarrow$ is_equal(A,A))

to guarantee domain restriction of the formulae.

6.3.2.3 Evaluable Functions and Predicates

Mathematical Axioms

Assume the slightly modified natural language text

(26) `Every company that buys at least three machines gets a discount.`
`Six Swiss companies each buy one machine.`
`A German company buys four machines.`

Answering the query

(27) `Who gets a discount?`

requires to derive a contradiction from the following two clauses that are submitted to Otter and Satchmo, resp.

(28) Subset of Clauses for the proof (26) |– (27)

```
- value(N,geq,3)
value(sk1,eq,4)
```

To derive a contradiction requires mathematical knowledge about natural numbers, viz. that if there is a number that is equal to 4 this number is also greater or equal than 3. In both, Satchmo and Otter mathematical knowledge is implemented by a combination of first-order mathematical axioms like axiom (Ax. 22) and (Ax. 23) which describe numeric relations, plus the use of evaluable functions and predicates. The concrete realization is different for Otter and Satchmo though.

Otter. In Otter the modelling of mathematical knowledge turned out to be rather tricky. Otter provides evaluable functions and relations like `+`, `>` or `≤` that can be used for integer arithmetic. An evaluable predicate like `>` operates on integers and evaluates to Boolean constants `T` and `F`. Otter's evaluable predicates, however, just trigger the numeric test and *not* the unification – meaning that variables have to be properly instantiated *before* the test is performed. And to control proper instantiation turned out to be a problem. To derive the contradiction for the clauses in (28) the addition of the axiom

(29) $\forall X \forall N \forall M(\text{value}(X,\text{eq},N) \wedge M \leq N \rightarrow \text{value}(X,\text{geq},M))$

will therefore not work since the evaluable predicate ≤ would be called with an uninstantiated variable M. Reformulating the axiom via logical equivalences to

(30) $\forall X \forall N \forall M(\text{value}(X,\text{eq},N) \wedge \neg\text{value}(X,\text{geq},M) \rightarrow M > N)$

does not help either for reasons that correlate to our use of the autonomous mode and additionally to Otter's internal main loop. (McCune 1994, pp. 5). The use of the autonomous mode prevents (i) the splitting of clauses in `usable` and `sos`, since the autonomous mode expects all clauses to be `usable`, (ii) the definition of special inference rules, (iii) the control of demodulation. To get proper instantiation for a number of cases I therefore use a trick combining the following two axioms

(Ax. O-1) $\forall V \forall N \forall C(\neg value(V,C,N) \rightarrow test\_value\_new(V,C,N))$
(Ax. O-4) $\forall V \forall N \forall G((value(V,eq,G) \wedge test\_value\_new(V,geq,N)) \rightarrow N > G)$

The axiom (Ax. O-1) provides the instantiation and (Ax. O-4) derives the contradiction. Other axioms specifically introduced for Otter are listed in Figure 13 on page 289. Using Otter's evaluable relation > and the axioms (Ax. O-1) and (Ax. O-4) then allows the intended deduction and the reasoner outputs:

(31) Output for the reasoner based on Otter

```
The reasoner proved that the query(-ies)
  Who gets a discount?
can be answered on the basis of the sentence(s)
  Every company that buys at least three machines gets a discount.
  A German company buys four machines.
using the auxiliary axiom(s)
  (Ax. O-1): Otter Number Axiom.
  (Ax. O-4): Otter Number Axiom.
```

One has to be careful though. The Otter manual recommends using evaluable functions and relations only if one knows how the inferences are going to occur when the formulae are conceived. This is difficult if you are using Otter's autonomous mode. For example, I have found no *general* solution to *prevent* evaluable conditions to be executed *before* all variables are instantiated. Due to these problems I consider my treatment of evaluable functions and relations in Otter as preliminary. Currently, the axioms work for most examples but scaling up to more complex examples is not guaranteed.

Satchmo. In Satchmo mathematical knowledge about natural numbers can be more directly implemented by triggering the execution of Prolog predicates during the proof. For the current example I use the following user-defined predicate that has the same effect as the above Otter axioms:

(32)

```
value(Cardinality,geq,NewNumber)-Index1:-
     number(NewNumber),
     value(Cardinality,eq,GivenNumber)-Index2,
     number(GivenNumber),
     NewNumber =< GivenNumber.
```

With this predicate it can be proved that an object has a `Cardinality` greater or equal to `NewNumber` (in the example 3) if that object has a `Cardinality` that equals `GivenNumber` (in the example 4) and if `NewNumber` is less or equal than `GivenNumber`. Instantiation problems can be easily avoided by the Prolog predicate `number/1`. Note that the current tracking mechanism does not report the use of evaluable functions or predicates, only the use of auxiliary FOL axioms is presented to the user.

Unless a more stable algorithm to treat evaluable functions in Otter's autonomous mode is found Otter cannot be used to deal with some plural examples. Satchmo has no such restrictions and is therefore superior to Otter in this respect.

Otter Specific Axioms

Mathematical Axioms

(Ax. O-1) $\forall A \forall B \forall C(\neg \mathrm{value}(A,C,B) \rightarrow \mathrm{test\_value\_new}(A,C,B))$
(Ax. O-2) $\forall A \forall B \forall C(\mathrm{value}(A,\mathrm{eq},C) \wedge \mathrm{test\_value\_new}(A,\mathrm{leq},B) \rightarrow B < C)$
(Ax. O-3) $\forall A \forall B \forall C(\mathrm{value}(A,\mathrm{eq},C) \wedge \mathrm{test\_value\_new}(A,\mathrm{less},B) \rightarrow B \leq C)$
(Ax. O-4) $\forall A \forall B \forall C(\mathrm{value}(A,\mathrm{eq},C) \wedge \mathrm{test\_value\_new}(A,\mathrm{geq},B) \rightarrow B > C)$
(Ax. O-5) $\forall A \forall B \forall C(\mathrm{value}(A,\mathrm{eq},C) \wedge \mathrm{test\_value\_new}(A,\mathrm{greater},B) \rightarrow B \geq C)$
(Ax. O-6) $\forall A \forall B \forall C(\mathrm{value}(A,\mathrm{leq},C) \wedge \mathrm{test\_value\_new}(A,\mathrm{leq},B) \rightarrow B < C)$
(Ax. O-7) $\forall A \forall B \forall C(\mathrm{value}(A,\mathrm{leq},C) \wedge \mathrm{test\_value\_new}(A,\mathrm{less},B) \rightarrow B \leq C)$
(Ax. O-8) $\forall A \forall B \forall C(\mathrm{value}(A,\mathrm{less},C) \wedge \mathrm{test\_value\_new}(A,\mathrm{leq},B) \rightarrow B \leq C)$
(Ax. O-9) $\forall A \forall B \forall C(\mathrm{value}(A,\mathrm{less},C) \wedge \mathrm{test\_value\_new}(A,\mathrm{less},B) \rightarrow B < C)$
(Ax. O-10) $\forall A \forall B \forall C(\mathrm{value}(A,\mathrm{geq},C) \wedge \mathrm{test\_value\_new}(A,\mathrm{geq},B) \rightarrow B > C)$
(Ax. O-11) $\forall A \forall B \forall C(\mathrm{value}(A,\mathrm{geq},C) \wedge \mathrm{test\_value\_new}(A,\mathrm{greater},B) \rightarrow B \geq C)$
(Ax. O-12) $\forall A \forall B \forall C(\mathrm{value}(A,\mathrm{greater},C) \wedge \mathrm{test\_value\_new}(A,\mathrm{geq},B) \rightarrow B \geq C)$
(Ax. O-13) $\forall A \forall B \forall C(\mathrm{value}(A,\mathrm{greater},C) \wedge \mathrm{test\_value\_new}(A,\mathrm{greater},B) \rightarrow B > C)$

List Operations

(Ax. O-14) $\forall A \forall B(\mathrm{ins\_a}(A,B) \rightarrow \mathrm{ins\_b}(A,B,[A|B]))$
(Ax. O-15) $\forall A \forall B \forall C(\mathrm{ins\_a}(A,[B|C]) \rightarrow \mathrm{ins\_a}(A,C))$
(Ax. O-16) $\forall A \forall B \forall C \forall D(\mathrm{ins\_a}(A,[B|C]) \wedge \mathrm{ins\_b}(A,C,D) \rightarrow \mathrm{ins\_b}(A,[B|C],[B|D]))$
(Ax. O-17) $\forall A(\mathrm{ins\_b}(A,[],[A]))$
(Ax. O-18) $\forall A(\mathrm{perm\_a}(A) \rightarrow \mathrm{p\_a}(A,A))$
(Ax. O-19) $\forall A \forall B \forall C(\mathrm{p\_a}(A,[B|C]) \rightarrow \mathrm{p\_a}(A,C))$
(Ax. O-20) $\forall A \forall B \forall C \forall D(\mathrm{p\_a}(A,[B|C]) \wedge \mathrm{p\_b}(A,C,D) \rightarrow \mathrm{ins\_a}(B,D))$
(Ax. O-21) $\forall A \forall B \forall C \forall D \forall E(\mathrm{p\_a}(A,[B|C]) \wedge \mathrm{p\_b}(A,C,D) \wedge \mathrm{ins\_b}(B,D,E) \rightarrow \mathrm{p\_b}(A,[B|C],E))$
(Ax. O-22) $\forall A(\mathrm{p\_b}(A,[],[]))$
(Ax. O-23) $\forall A \forall B(\mathrm{p\_b}(A,A,B) \rightarrow \mathrm{perm\_b}(A,B))$
(Ax. O-24) $\forall A \forall B(\neg \mathrm{permutation}(A,B) \rightarrow \mathrm{perm\_a}(A))$
(Ax. O-25) $\forall A \forall B(-\mathrm{permutation}(A,B) \rightarrow \neg \mathrm{perm\_b}(A,B))$
(Ax. O-26) $\forall A \forall B(\neg \mathrm{sum\_of}(A,B)) \rightarrow \mathrm{test\_sum\_of\_new}(A,B))$
(Ax. O-27) $\forall A \forall B \forall C(\mathrm{sum\_of}(A,C) \wedge \mathrm{test\_sum\_of\_new}(A,B) \rightarrow \neg \mathrm{permutation}(C,B))$

Figure 13 Otter Specific Auxiliary FOL-Axioms for Reasoning with Plurals

List Manipulation

In my logic representation I model NP coordinations using lists. Lists basically model the join operation of lattice theory. In Satchmo, commutativity, associativity and idempotence of the lattice-theoretic join operation are not directly implemented via first-order axioms but more efficiently simulated by Prolog's built-in list processing operations like permutation, flattening, removing duplicates etc. These list processing operations are encoded as evaluable functions. In Otter, the relevant list processing operations are simulated by further auxiliary axioms.

To perform the following deduction permutation of lists is necessary:

(33) IF John and Mary together buy a house.
THEN Mary and John together buy a house.

In particular, we need to deduce an inconsistency with respect to the following two clauses derived from the DRS conditions for the antecedent and the consequent in the proof.

```
(34)  sum_of(sk1,['John','Mary'])
      -sum_of(B,['Mary','John'])
```

Otter. For the implementation of list operations an instantiation trick similar to (Ax. O-1) and (Ax. O-4) was necessary. To model e.g. commutativity with respect to the second argument of `sum_of/2` I defined "prolog-like" first-order axioms that simulate the permutation operation. These axioms are listed in Figure 13 as (Ax. O-14) to (Ax. O-27). For comments to the axioms (Ax. O-14) to (Ax. O-23) see also Kalman (2001, pp 358). Axioms (Ax. O-24) and (Ax. O-25) allow to call `permutation/2` directly with two arguments. Axioms (Ax. O-26) and (Ax. O-27) link the permutation operation to the condition `sum_of/2` and test whether two instantiated sets are permutations of each other. If the sets are not instantiated a non-terminating permutation operation could result. Again one has to be careful since similar instantiation problems due to Otter's autonomous mode occur and are not fully solved yet.

Satchmo. Again, in Satchmo list operations are more straightforwardly implemented taking recourse to Prolog's built-in list and sorting operations.

(35)
```
sum_of(A,B)-Index1:-
     ground(B),
     sum_of(A,D)-Index2,
     ground(D),
     list_to_ord_set(B, E),
     list_to_ord_set(D, F),
     ord_seteq(E,F).
```

The list permutation operations allow to perform the deduction in (33):

(36) Output for the reasoner based on Satchmo

```
The reasoner proved that the sentence (-s)
  John and Mary together buy a house.
can be answered on the basis of the sentence(s)
  Mary and John together buy a house.
```

Note that for Satchmo no auxiliary axioms are output since only recourse to user-defined predicates is required for the proof. It is debatable whether the execution of evaluable predicates should also be reported within the tracking process.

Note finally, that lists are the only functions that I use in my first-order logic representation. I use lists directly instead of two-place operators because of computational advantages with respect to Otter and Satchmo.

6.3.2.4 Lexical Knowledge

Meaning Postulates for Single Words

The external knowledge base of the reasoner also contains axioms that relate to our lexical knowledge about the semantics of certain words. For example the axioms (Ax. 24)-(Ax. 34) in Figure 12 on page 283 encode logical relations between different non-standard quantifiers. Given only the logical representations presented in chapter 4 these intended logical relations cannot be derived but need additional knowledge. This lexical knowledge is given in the form of additional axioms. Similarly the axioms (Ax. 36)-(Ax. 37) encode logical relations between query words and modification types.

I am aware that encoding extensive lexical knowledge in the form of auxiliary first-order axioms is difficult and most probably not very efficient. Nevertheless, for certain application domains lexical inferences may be desirable and the auxiliary FOL axioms provide a tool to add this knowledge.

Meaning Postulates for Word Classes

The DRoPs reasoner also contains logical axioms that encode lexical knowledge for certain verb classes, more concretely knowledge about the subentailment properties of certain verb classes. Currently, these inference rules are very coarse and need refinement if a more precise logical analysis is desired. Nevertheless, I will show some of these inference rules here that relate to collective/distributive subentailments of certain argument positions. I will describe the inference rules first verbally and then show how they are preliminarily realized within the DRoPs reasoner.

In the previous chapters I have addressed the following inference rules with respect to the external argument of one-place verbs:

(I1) One-place event verbs
If the subject of an intransitive event verb that is not marked as strictly collective in the subject position is interpreted collectively (one event) one can infer that for every member of the group denoted by the subject there is a subevent of the complex event where that individual has the relation denoted by the verb, i.e. the predicate distributes down to the individual members of the group.

The rule predicts that we can make the following inference:

(37) IF John and Mary left.
THEN John left.

Since the inference rule is very general one might object that one could directly assume a distributive reading instead of first assuming an indeterminate collective reading and then distributing the event down to the individual members of the group. However, if one immediately distributes the event it is e.g. in (37) difficult to represent that there is something that brings together two persons in a complex event. As discussed in section 4.3 it is – for some verb classes – difficult to really distinguish collective from distributive readings. Assuming a collective reading for (37) the leaving events of John and Mary are somehow related, e.g. temporally or spatially, assuming a distributive reading there is no relation whatsoever. A further – technical advantage – of starting with a indeterminate collective reading is that the cumulative inference

(38) IF John left. Mary left.
THEN John and Mary left.

is much more costly to implement than the reverse direction. In fact, in my implementation there are no cumulative inferences (see section 4.3.2). For more detailed inference-rules one would have to make a more fine-grained distinction of verb classes.

The difference between collective and distributive readings for intransitive verbs is more obvious in sentences with additional adverbial modifiers like

(39) a. John and Mary left in a train.
b. John and Mary left quickly.

Do we want to infer from (39)a that John left in a train and that Mary left in the same train, or that Mary left in a possibly different train. This question cannot be decided since the sentence is ambiguous. In my setting I predict a default collective reading for (39)a, which means that John and Mary left in the same train. The inference rules currently do not distribute the modifier down to the individual members of the group. That means, from both sentences in (39) we can conclude that John left and that Mary left, but we can conclude nothing about whether each leaving was quick, or whether each leaving was in a train.

In the discussion in the previous chapter I have also considered the following very general inference rules for state verbs.

(I2) One-place state verbs
If the subject of an intransitive state verb that is not marked as strictly collective in the subject position is interpreted collectively (one state) one can infer that every member of the group denoted by the subject has the property in that same state.

The rule predicts the following inference:

(40) IF John and Mary sleep.
THEN John sleeps. Mary sleeps.

Again, there is a problem if the sentences are further modified.

For direct objects of two-place event verbs I assume similar inference rules.

(I3) Two-place event verbs
If a transitive event verb has a singular or distributive subject and a default collectively interpreted direct object that is not lexically or structurally marked as strictly collective then one can infer for every subgroup of the group denoted by the direct object that there is a subevent of the complex event where the subject bears the respective relation to the subgroup.

The rule predicts, that we can infer

(41) IF John enters five cards.
THEN John enters a card./ John enters two cards.

(I4) Two-place state verbs
If a transitive state verb has a singular or distributive subject and a default collectively interpreted direct object that is not lexically or structurally marked as strictly collective then one can infer that every subgroup of the group denoted by the direct object bears the relation to the subject in that same state.

The rule predicts, that we can infer

(42) IF John sees five birds.
THEN John sees one bird./ John sees three birds.

The suggested inference rules are (too) general in nature, which implies that they make predictions that are not intended or correct. For a small fragment they are easy to control. However, the larger the fragment the more difficult it is to think of all possible constellations and possible inference rules. If one had a more fine-grained verbal semantics one could for each verb-class formulate special inference rules that would allow for a more detailed text understanding. Here I only make first suggestions for the inference rules and develop techniques of how to implement these rules. A further matter of consideration could also be to view inference rules not as axioms but treat them as default rules that can be overridden if other information becomes available. I have not pursued this option here.

In the current implementation I use the same axioms for event and state verbs. This reduces the above inference rules to the following three axioms. In general, the current implementation of the distributive subentailments is not yet fully worked out and needs further refinement since the axioms are not yet restrictive enough. I still give a preliminary sketch of the necessary axioms here:

The distributive subentailments of external arguments is implemented as follows:

(Ax. 38) $\forall A \forall B \forall C \forall D(\text{predicate}(A,B,C) \wedge (\text{structure}(C,\text{group}) \vee \text{structure}(C,\text{mass})) \wedge \text{proper\_part\_of}(D,C) \wedge \text{collective}(C,A,\text{no}) \rightarrow \exists E(\text{predicate}(E,B,D) \wedge \text{e\_part\_of}(E,A)))$

The condition `collective(C,A,no)` states that the group `C` must not be marked as strictly collective with respect to the event `A`. Note that in the discussion of the semantic representation in chapter 4 this condition has not been added to the DRSs. Without the condition the axiom (Ax. 38) would also be triggered if the external argument were marked as collective, e.g. in the sentence

(43) Two men gather.

To prevent the distributive subentailment for sentence (43) the reasoner has to have access to a condition stating that *two men* is marked as strictly collective with respect to *gather*, this can be realized by adding the condition `collective(C,A,yes)`. Due to the additional condition axiom (Ax. 38) is not triggered for (43).

The inference rule for transitive verbs is realized as:

(Ax. 39) $\forall A \forall B \forall C \forall D \forall E(\text{structure(C,atomic)} \wedge \text{predicate(A,B,C,D)} \wedge (\text{structure(D,group)} \vee \text{structure(D,mass)}) \wedge \text{proper\_part\_of(E,D)} \wedge \text{collective(D,A,no)} \rightarrow \text{exists(F,predicate(F,B,C,E)} \wedge \text{e\_part\_of(F,A))}$

Both axioms needs the auxiliary axiom structuring the domain of eventualities.

(Ax. 40) $\forall A \forall B \forall C(\text{structure(A,B)} \wedge \text{e\_part\_of(C,A)} \rightarrow \text{structure(C,B))}$

The axiom states that subevents/substates are again events/states.

Note again, that the distributive subentailments for verb classes can be neatly expressed in first-order logic due to the flat notation where the predicate name occurs as the second argument of the predefined condition `predicate/3` and `predicate/4`, resp.

6.3.2.5 Domain Specific Axioms

The architecture of the DRoPs reasoner also allows to add domain specific knowledge as auxiliary first-order axioms. This domain specific knowledge could consist of domain-specific ontologies, of definitions or equivalences of certain words etc. An interesting application would be to import existing ontologies for a certain domain. Again, it is of advantage that the DRoPs reasoner is based on first-order logic since first-order logic is a widely accepted knowledge representation language.

6.4 Evaluation

6.4.1 Problems and Practical Limitations

The current implementation of the reasoner based on Satchmo and Otter resp. is still preliminary and several practical limitations need to be improved for a large-scale application. Currently, the reasoner works neatly for small examples. An extension to larger problems needs, however, further investigation. Furthermore, more systematic tests have to be performed that check the performance and the robustness of the system.

The implementations of the reasoners based on Otter and on Satchmo are still used independently. Ideally, they should be combined into one integrated inference system – similar to the MathWeb approach – and thus to provide competing theorem provers for a certain problem.

Otter and Satchmo do not yet accept the same set of evaluable functions and predicates. Otter uses its own set of built-in functions and predicates – that lead to instantiation problems – while Satchmo uses Prolog predicates – that are less problematic. A future extension would be to make Prolog predicates accessible to Otter via Otter's interface to foreign evaluable functions.

Due to the flat notation the number of atoms in the logic representation increased. The current "tracking version" of Satchmo does currently not scale up to larger examples formalized in the flat DRS notation. Reducing the number of conditions in the DRS considerably speeds up the time needed for a proof. Since the original version of Satchmo has no problems with the increasing number of conditions due to the flat notation we work on an update of the "tracking version" of Satchmo that solves the problems. For the same reason it can be argued that the flat notation is perhaps "too" flat and should be more compactly formulated. This could be achieved directly within the parser, or it could be based on the current representation by an automated postprocessing that transfers the current flat representation into a more compact flat representation. I am currently experimenting whether such a notation would improve the performance.

In some of the FOL axioms of the current implementation the problem of indirect left-recursion has not been fully eliminated. This leads to non-termination in some examples. This left-recursion can be eliminated with standard techniques but has not yet been realized since it requires a careful (and error-prone) update of many first-order auxiliary axioms.

A link to a demo of the reasoner based on Satchmo can be found at `www.ifi.unizh.ch/~uschwert/index.html`. There the interested reader can also find the set of axioms used for Satchmo and the set of user-defined Prolog predicates. At the time of writing this section the demo version of the reasoner does not yet deliver all solutions in all cases. If unsatisfiability is caused by a Prolog built-in predicate only the first solution is delivered. N. E. Fuchs has programmed an update of the tracking mechanisms that will – after sufficient testing – be made available on-line. The demo of the reasoner based on Otter is only locally available. The main reason is that the instantiation problems that occur with respect to Otter's evaluable functions have not yet been satisfactorily solved for all problems.

6.4.2 Conclusion and Further Research

The DRoPs reasoner that is based on the reasoner RACE (Fuchs and Schwertel 2002, 2003) works on fully specified DRSs created by the DRoPs disambiguation component. The reasoner proves that one natural language text can be deduced from another one and gives a justification for the proof again in natural language. Variations of the basic proof procedure permit question answering and consistency checking. Extending the reasoner by auxiliary first-order axioms and evaluable functions and predicates the reasoner can perform complex deductions on sentences with plurals and numbers. The reasoner is thus a practical tool that demonstrates the proof-theoretic approach to semantics as discussed in section 4.2. The possibility to add general auxiliary first-order axioms shows the usefulness of the flat first-order notation introduced in section 4.5. Also, I showed that the semantic representation that I developed for plurals allows us to reuse different existing standard tools. Furthermore, the approach has the advantage that the auxiliary first-order axioms can be reused in other first-order based applications. Only recourse to built-in evaluables is application specific.

The originally intended practical application of the reasoner was to assist users in logically analysing a natural language text. It turned out that the reasoner can also be used within other useful practical applications. As a special side-effect the application of automated theorem proving techniques can help semanticists to check the correctness of their proposed semantic representations. If the desired inferences can be practically simulated this gives an immediate feedback whether the semantic representation captures the intended interpretation. There are also pedagogical applications of the reasoner: it can e.g. be used for teaching logic, in particular to train and to understand the relation between natural language statements and their first-order representations. The tracking mechanism supports the understanding of logical reasoning processes. For this purpose, it would even be desirable to have a more sophisticated tracking component that is not only based on returning complete sentences used for a proof but that visualizes which individual conditions participated in the proof (see Sukkarieh 2001a, 2001b).

The reasoner performs question answering by logical deduction. I have already discussed that this approach is not suitable to answer *how many*-questions or to answer questions like

(44) Does every company buy a machine?

where a user wants a comprehensive check whether all involved companies buy a machine. In the current setting, the question (44) will only be true if the natural language text explicitly contains a statement like

(45) Every company buys a machine.

An exhaustive list is, however, output when the question is stated as follows:

(46) Which company buys a machine?

It remains to be investigated whether the models that Satchmo generates if a set of clauses is satisfiable can be used to build a more sophisticated question answering tool.

Further interesting research topics would be to check how default-reasoning could be integrated using automated theorem proving techniques. Also, the use of theorem proving techniques for disambiguation itself is an interesting and important field of research.

7 Plural Processing in Attempto Controlled English

7.1 Overview

In the following sections I will show how techniques developed for the DRoPs system in the previous sections are applied to process plurals within the natural-language understanding application Attempto (Schwitter 1998, Fuchs, Schwertel and Schwitter 1999a, 1999b, Fuchs and Schwertel 2002, 2003, Schwertel, Fuchs and Höfler 2003). The core of the Attempto system is the controlled natural language Attempto Controlled English (ACE). Since ACE is not a full, but a *controlled* natural language the Attempto system has partly different requirements than the DRoPs system. In particular, disambiguation in ACE follows different principles than disambiguation of unrestricted natural language in DRoPs. Whereas the DRoPs system generates a hierarchy of plausible readings, the Attempto system generates for each sentence just one reading that is uniquely predictable from a set of construction and interpretation rules. This deterministic approach is a result of the constructive disambiguation strategy chosen for the Attempto system. The strategy will be motivated below.

As for plural and scope ambiguities only minor parameter re-settings were necessary to use the DRoPs implementation of plurals also for the Attempto system. Also, the underlying logical form, i.e. flat first-order DRSs is identical. This is a further evidence for the generality and re-usability of the DRoPs approach to the representation and disambiguation of plurals. Vice versa, I re-used the core reasoning component (RACE) of the Attempto system for the DRoPs reasoning component, showing that the DRoPs approach allows re-use of existing tools. Due to the technical and implementational overlap with the DRoPs system the focus of this chapter will therefore be on the language related part of ACE and there will be only a brief overview of the implementation of the Attempto system itself.

In section 7.2 I will give an overview of the language ACE. I will explain the motivation for developing the language ACE, explain ACE's constructive disambiguation strategy, and give examples for important construction and interpretation rules that concern the singular fragment of ACE. Section 7.3 describes the main ideas of extending ACE with plural constructions. The section shows how syntactic and semantic plural ambiguities can be handled using the constructive disambiguation approach of ACE. A full specification of the language ACE in general and of the plural extensions in particular can be found in the ACE 4 manual (Schwertel, Fuchs and Höfler 2003). A slightly modified excerpt of the plural part of the ACE 4 manual can also be found in Appendix E of this thesis. Section 7.4 will give a brief overview of the architecture of the Attempto system. Since many techniques are similar to the DRoPs system no further details are explained. Finally, in section 7.5 I will conclude, address possible extensions and improvements and give examples for other currently investigated applications of ACE which show the flexibility and the usefulness of the approach pursued in the Attempto project.

7.2 Introduction to Attempto Controlled English

7.2.1 The Philosophy of ACE

The controlled natural language Attempto Controlled English (ACE) was originally designed as a software specification language that bridges the gap between formal and informal specification languages (Schwitter 1998, Fuchs, Schwertel and Schwitter 1999a). Whereas *formal* specification languages have the disadvantage that for domain specialists they are hard to understand and difficult to relate to the application domain, the uncontrolled use of *informal* natural language can lead to ambiguous, imprecise and unclear specifications with possible disastrous consequences for the subsequent software development process. The Attempto project is directly addressing both problems by replacing obviously formal specifications by specifications in Attempto Controlled English (ACE). ACE is a controlled subset of English with a domain-specific vocabulary, a restricted a grammar and a number of interpretation rules. Every ACE sentence is correct English, but not every English sentence is admissible in ACE. ACE allows users to express specifications precisely, yet using the familiar terms of the application domain. ACE specifications are computer-processable and can be unambiguously translated into first-order logic. Though ACE seems informal it is in fact a formal language with the semantics of the underlying logic language. This also means that ACE has to be learned like other formal languages. As a *formal* specification language ACE has the advantage that it supports the automatic analysis of specifications such as consistency verification, and the option to validate specifications through execution (Fuchs 1992b). There is a further incentive in employing ACE: formal tools like theorem provers or model generators become available even to domain specialists who are not familiar with the formal notations that these tools normally require. ACE can replace first-order logic as an interface language to these tools (Fuchs, Schwertel and Torge 2000, Fuchs and Schwertel 2002, 2003).

The development of ACE was inspired by the observation that controlled languages are successfully used for technical documentation (CLAW 1998) and tools are developed to assist technical writers (CLAW 1998, CLAW 2000, CLAW 2003). However, these controlled languages are different from ACE in that they are usually ad hoc defined and rely on rather liberal rules of style and on non-formal conventions that are often not automatically checked. A further difference is that the main goal of these controlled languages is to make text easier for people to write and read whereas ACE has a goal beyond that: texts should also be easier for a computer to understand. There have been projects with similar aims as Attempto, however, the subsets of English were not systematically and clearly defined. For example Macias and Pulman (1995) developed a system which resembles ACE with the important difference that their system restricts only the form of composite sentences, but leaves the form of the constituent sentences completely free. As a consequence the problem of ambiguity remains and has to be resolved by the users after the system has translated the specification into a formal representation. A more recent approach using controlled natural language was pursued within the project PROSPER (Proof and Specification Assisted Design Environments) (Grover et. al. 2000). One aim of the PROSPER project is to make formal specification and verification techniques more user-friendly by developing novel interfaces for requirements capture. One such interface is given by a controlled natural language used for specifying hardware properties. The idea is to allow the expression of temporal logic constraints on circuits through sentences of a restricted subset of English. The project makes also use of model checking techniques on the basis of these English specifications. Since the application is developed for a very restricted domain direct comparisons with the Attempto project are difficult.

7.2.2 Ambiguity Processing in ACE: The Constructive Strategy

Ambiguity occurs on all levels of natural language processing (lexical, syntactic, semantic, pragmatic). ACE is fundamentally concerned with syntactic and semantic ambiguities while many other controlled languages aim at controlling lexical ambiguity, e.g. by standardizing the vocabulary. In section 2.5 I have discussed several disambiguation strategies of natural language processing systems dealing with *uncontrolled* natural language, e.g. *Generate and Test* and *Underspecified Representations*. Also in section 5.4 I presented a semantic disambiguation algorithm for plurals that makes use of both strategies. However, all of these strategies are not directly suitable to resolve syntactic and semantic ambiguities in ACE.

Since ACE is a formal language every sentence has to get just one and only one interpretation. This interpretation should be systematic, reliable and predictable, i.e. the same syntactic construction must lead to the same interpretation in all cases. This requirement precludes context, world-knowledge or even extended lexical knowledge as a driving disambiguation force since then the same syntactic constructs could lead to different disambiguations depending for example on context or on the concrete application domain. A further prerequisite of the ACE disambiguation rules is that they are easy to use and to remember for the user.

For these reasons ACE chooses a different approach to handle both syntactic and semantic ambiguities. We call our approach *constructive*. The *Constructive Approach* consists of three simple means:

(1) Constructive Disambiguation in ACE

1. The *construction rules* of ACE avoid many ambiguous sentences and phrases; unambiguous alternatives are available in their place.
2. All remaining ambiguous constructions are interpreted deterministically on the

basis of a number of *interpretation rules* that use syntactic information only; the interpretations are reflected to the user in a *paraphrase* which is again ACE.
3. Users can either accept the assigned interpretation, or they must *rephrase* the input to obtain another one. A number of non-formal ACE *style rules* support the user to find alternative formulations.

In section 7.2.3 below I will introduce some of ACE's interpretation rules. For example in full natural language the sentence

(2) A manager orders a program with a code.

is syntactically ambiguous. In ACE there is an attachment interpretation rule stating that a prepositional phrase always modifies the verb. ACE will therefore generate the following paraphrase where the brackets indicate the attachment.

(3) A manager {orders a program with a code}.

This interpretation is probably not the one intended by the user. To obtain the other interpretation the user can reformulate the sentence using a complementary interpretation rule stating that relative sentences always modify the immediately preceding noun phrase:

(4) A manager orders a program that has a code.

yielding the paraphrase

(5) A manager orders {a program that has a code}.

The constructive disambiguation strategy of ACE allows for a systematic, automatic, predictable and efficient disambiguation of ACE texts. It requires no recourse to contextual knowledge and thus guarantees an expected and reproducible behaviour of the Attempto system. A problem is of course that the rules do not in all cases lead to the natural or preferred interpretation and both, writers and readers of ACE texts, have to learn the disambiguation rules. Furthermore, the higher the expressive power of ACE the more difficult it is to control all ambiguities this way. Some of the possible ambiguities may even remain undetected and no rule may be given. In section 7.3 I will extend ACE with plurals and this extension shows that the ACE disambiguation rules have to be carefully designed with respect to already existing language constructs so as not create conflicts, or new ambiguities. Despite these problems, we are convinced that for ACE and its intended applications there is no other choice than a constructive disambiguation strategy.

7.2.3 The Language ACE (Singular) in a Nutshell

In this section I will briefly present the components of the singular fragment of ACE before I show in section 7.3 how ACE has been extended with plurals. Here I can only show some examples of the structure of ACE. For a full definition of the language ACE see the ACE 4 language manual (Schwertel, Fuchs and Höfler 2003). Also for current information on the status of the Attempto project consult the Attempto homepage (www.ifi.unizh.ch/attempto).

7.2.3.1 Lexicon

The vocabulary of ACE comprises

- predefined function words (e.g. determiners, conjunctions, prepositions, query words)
- user-defined, domain-specific content words (nouns, verbs, adjectives, adverbs)

Users can define content words with the help of a lexical editor that presupposes only basic grammatical knowledge (Dörflinger 2003, Hein 2003). Alternatively, we are working on a tool to import existing lexica (Bünzli, to appear). ACE allows the user to define so called aliases which resemble synonyms in full natural language (for example *insert* as an alias for *enter*) and abbreviations (for example *SM* as an abbreviation for *SimpleMat*).

Note that in ACE the content vocabulary is completely in the responsibility of the user. This in particular implies that lexical ambiguities have to be controlled by the user. This is in contrast to many other controlled languages which aim at controlling lexical ambiguity, e.g. by standardizing the vocabulary.

7.2.3.2 Construction Rules

Construction rules define the form of ACE sentences and texts, and state restrictions intended to limit imprecision and to remove ambiguities of full natural language. Following are some important construction rules.

ACE Specifications. ACE specifications are sequences of anaphorically interrelated simple and composite sentences.

Simple Sentences. Simple sentences have the form *subject* + *verb* + *complements* + *adjuncts* where complements (noun phrases, prepositional phrases) are required for transitive and ditransitive verbs, and adjuncts (adverbs, prepositional phrases) are optional. An example is:

(6) A manager of a German company orders a machine in Switzerland.

Composite Sentences. Composite sentences are built from other sentences through sentence or VP coordination (*and*, *or*), subordination by *if … then …*, subordination by relative sentences (*who*, *which*, *that*), verb phrase negation (*does not*, *is not*), sentence negation (*it is not the case that*), global quantification (*for every*, *there is a*). Examples are

(7) a. Every company that buys a machine of a Swiss manufacturer gets a discount.
b. If a company buys a machine of a manufacturer then the manufacturer delivers a regular software update.
c. For every machine there is a software update of the machine.

Anaphora. ACE sentences can be interrelated by anaphora, i.e. personal pronouns or definite noun phrases that refer to previously mentioned noun phrases. Examples can be found in (7)b/c or in

(8) Hardware Corporation produces a special machine. A Swiss customer orders it.

The personal pronoun *it* of the second sentence is an anaphoric reference to the noun phrase *a special machine* of the first sentence. Similarly, *the manufacturer* in (7)b is an anaphoric reference to *a manufacturer*.

Verbs. Verbs are only used in the simple present tense, the active voice, the indicative mood, and the third person.

We are currently considering extensions to also include the imperative mood.

Modality. Modal verbs (*can*, *must* etc.), intensional verbs (*hope*, *know* etc.), and modal adverbs (*possibly*, *probably* etc.) are not supported in ACE.

Questions. ACE allows for *yes/no*-questions and *wh*-questions derived from ACE sentences.

(9) a. Who buys a Swiss machine?
b. Does a customer buy a Swiss machine?

'Of'-Constructions. The only allowed postnominal prepositional phrases are *of*-constructions (e.g. *a manager of a company*).

There are further constructions rules, e.g. rules that the control conjunction and disjunction of sentences and phrases. A complete list can be found in the ACE 4 manual.

7.2.3.3 Interpretation Rules

Interpretation rules control the semantic analysis of grammatically correct ACE sentence. They, for example, resolve syntactic and semantic ambiguities that cannot be avoided by the construction rules alone. The result of this analysis is reflected to the user in a paraphrase. The paraphrase contains two types of brackets. Square brackets show substitutions, and (optional) curly brackets show how syntactic elements are grouped. Important interpretation rules are:

PP Attachment Ambiguities. Prepositional phrases in adjunct position always modify the verb. For example in

(10) a. A manager orders a machine in Switzerland.
b. A manager {orders a machine in Switzerland}.

the adjunct *in Switzerland* gives additional information about the location of the ordering event, not about the location of the machine. This rule is (optionally) shown to the user by the paraphrase (10)b.

Anaphora Resolution. Anaphoric reference is possible via definite noun phrases or pronouns (personal, reflexive or possessive pronouns). Definite noun phrases refer to the most recent accessible suitable noun phrase that agrees in number and gender. For personal pronouns and reflexive pronouns there are complementary interpretation rules. A personal pronoun always refers to the most recent accessible noun phrase that has the same number and gender and that *is not* the subject, an object or an adjunct of the same verb, whereas a reflexive pronoun always refers to the most recent accessible noun phrase that has the same number and gender and that *is* the subject, an object or an adjunct of the same verb. For example in

(11) A technician installs a machine. A project manager observes him.

the personal pronoun *him* refers to *a technician* not to *a project manager*, whereas in

(12) A technician installs a machine. A project manager observers himself.

the reflexive pronoun *himself* relates to *a project manager*. For possessive pronouns I refer to the rules in the ACE 4 manual.

Quantifier Scope. The relative scope of a quantifier corresponds to its surface position. A textually preceding quantifier has wide scope with respect to a succeeding quantifier that has narrow scope. The range of the scope is determined by extra principles for local (*every*, *a*) and global quantifiers (*for every*, *there is a*).

Binding Hierarchy. Sentences can contain several different connectors. To avoid ambiguities ACE uses a binding hierarchy similar to formal languages. The following binding hierarchies are realized (where ">" stand for "binds stronger than"). Note that for conjunctions and disjunctions commas are used to override the default binding:

(13) Binding Hierarchy for Connectors
Negation > Conjunction > Disjunction > Implication

(14) Binding Hierarchy for Conjunctions
'*and*' > '*and then*' > '*, and*' > '*, and then*'

In the ACE 4 manual the interested reader will find a complete set of the ACE interpretation rules with many examples. Also the manual contains a style guide with hints how problems can be avoided and how sentences can be reformulated to express the intended interpretation.

7.2.4 Design Principles

Defining ACE

In designing the language ACE our emphasis was to find a good trade-off between a number of – partly competing – requirements. ACE should have sufficient expressive power for the needs of a typical user. The construction and interpretation rules should support the constructive disambiguation strategy of ACE. They should not require user interaction to choose a particular interpretation. Furthermore, the rules should be simple, systematic and general so that ACE is easy to learn and to use. Exceptions to the rules should be kept to a minimum. Still, the rules should predict the natural interpretation of a sentence in a majority of cases. A further important criterion was that the construction rules should not allow for ACE sentences that are obviously not acceptable in full English. We could not avoid that valid ACE sentences sometimes sound "stilted" but we tried to avoid obviously unacceptable constructions. Besides syntactic requirements there were also logical requirements, e.g. the predicted interpretation should support intuitive inferences. Also, the semantic representation of ACE sentences should support the use of automatic inferencing tools. We noted that adding a new language construct needs careful consideration so as not to violate these requirement. As I will show below the introduction of plurals to ACE is a particularly tricky issue in this respect. As a result of our design principles ACE sentences may sometimes look over-simplistic but we consider this a valuable feature rather than a disadvantage. After all, defining ACE as a simple and systematic language conforming to the requirements is by no means trivial.

Learning ACE

We think that – in contrast to the rules of formal languages – the construction and interpretation rules of ACE are easy to use and to remember since they are similar to English grammar rules and only presuppose basic grammatical knowledge. One has to bear in mind, however, that in spite of its natural appearance ACE is in fact a formal language that – like other formal languages – must be learned. Positive experience with teaching ACE to a group of students support our view that ACE can be learned with reasonable effort. Further experiments are necessary though. The view that ACE can be realistically learned is also supported by the fact that companies like Boeing or Caterpillar report that their controlled languages can be taught to users in a few days and users get competent within a few weeks. Thus we claim that domain specialists need less effort to learn and to apply the rules of ACE than to cope with an unfamiliar formal language. Thus the Attempto approach also increases the acceptability of formal methods.

Still, there is room for improvement. It would, for example, be desirable to develop automatic tools that assist the user with the learning process. For example, if a sentence is not accepted the user could be given feedback which ACE rule was violated. One could also think of tools proposed by Schwitter and Ljungberg (2003) who developed ECOLE, a look-ahead editor for the controlled language PENG that is similar to ACE. The editor provides syntactic hints after each word form entered and indicates how the author can continue the text. Schwitter and Ljungberg claim that this way the author does not need to learn or to remember the restrictions of the controlled language. While this approach may support learning the construction rules it still misses support for teaching the interpretation rules. Therefore, it would also be desirable to develop tools – beyond the ACE paraphrase – that show the users how a sentence is interpreted by the Attempto system or how they can reformulate a sentence.

7.3 Plurals in ACE

7.3.1 How to Control Plurals in ACE?

Not all plural sentences can be equivalently expressed by reformulations in the singular. In particular collective readings are irreducible and therefore need to be added to ACE to guarantee sufficient expressive power. In full natural language, plurals do not only introduce semantic but also syntactic ambiguities, e.g. in connection with noun phrase coordination. In ACE, both types of ambiguities have to be controlled by construction and interpretation rules. A main problem with controlling plural ambiguities in ACE was to find a good trade-off between the different guiding principles of ACE that I have introduced in section 7.2.4 above. The following sections will give an overview how plurals are used and interpreted in ACE.

Simplicity or Naturalness?

To control semantic ambiguities of plurals in ACE I considered several options. One option is to base plural disambiguation in ACE on the DRoPs disambiguation algorithm (see section 5.4) by simply choosing the best interpretation predicted by DRoPs as the unique interpretation of ACE. This treatment would, however, violate some of the design principles of ACE. First, the DRoPs algorithm is too complicated to be easily learned, remembered and actively used. Second, the DRoPs algorithm would in some cases violate basic assumptions of ACE, e.g. the interpretation rule stating that surface order determines quantifier scope. Third, although the DRoPs algorithm relies mostly on structurally available information there is some recourse to complex lexical knowledge that is difficult to manage within ACE because users define part of the lexicon themselves and therefore should not be burdened with giving too much detailed lexical information.

As a second option for controlling semantic plural ambiguities in ACE I therefore considered a simpler alternative (Schwertel 2000) that requires less complicated calculations than the DRoPs algorithm. This approach was using a simplified distributivity hierarchy and information about the grammatical function of an NP (see section 5.3.2). The simplified distributivity hierarchy assumed a distinction between quantificational and individual denoting noun phrases where the latter are further subdivided into NPs that prefer a collective interpretation (*two machines*, *some customers* etc.) and NPs that prefer a distributive interpretation (*at least two companies*, *exactly three machines* etc.). The reading of an NP was then calculated on the basis of this simplified distributivity hierarchy and additionally dependent on the grammatical function of the NP. The approach predicted for example that in

(15) a. Two companies order a machine.
b. At least two companies order a machine.
c. A company orders at least two machines.

the plural NP *two companies* in (15)a was assigned a collective reading, in (15)b the plural NP *at least two companies* in subject position was assigned a distributive reading, whereas in (15)c the same NP *at least two machines* in object position received a collective reading. Although this approach attempted to offer a good compromise between using manageable rules, yet making intuitive semantic predictions, discussions with potential users of ACE revealed that even this simplified approach is too complicated to be learned and remembered. Also, the approach has the problem that the same syntactic construction does not always lead to same semantic interpretation, e.g. *at least two companies* is interpreted differently depending on its grammatical function. This constitutes a violation of the constructive disambiguation strategy of ACE.

How to Trigger Disambiguation?

Therefore I considered a radical simplification of plural disambiguation in ACE which is based only on the distinction between individual denoting and quantificational noun phrases ignoring the grammatical function of the NPs. The approach uses the following two basic assumptions:

(16) Basic Assumption of Constructive Plural Disambiguation in ACE
1. All individual denoting plural NPs get a default collective reading.
2. Distributive readings are triggered using partitive constructions with *each of*.

This radical simplification predicts for example that each of the three sentences in (15) gets a default collective reading. This result shows that in ACE the conflict between simplicity and generality of rules on the one hand, and the naturalness of the interpretation on the other hand cannot always be perfectly reconciled. In this case simplicity, generality and learnabilty of the rules was assessed more important than the naturalness of the interpretation, since in full natural language sentence (15)b tends to prefer a distributive reading. The second rule of (16) states that the distributive readings of the sentences in (15) have to be expressed by using constructions with *each of*. Note that NPs introduced by *each of* in subject position require singular agreement with the verb.

(17) a. Each of two companies orders a machine.
b. Each of at least two companies orders a machine.
c. A company orders each of at least two machines.

Equivalent, more natural ACE reformulations would be

(18) a. There are two companies. Each of them/the companies orders a machine.
b. There are at least two companies. Each of them/the companies orders a machine.
c. There are at least two machines. A company buys each of them.

Again, triggering the distributive reading only by using *each of* and not by the disambiguation marker *each* as proposed in Schwertel (2000) is a compromise between different ACE requirements: on the one hand ACE sentence should sound as natural as possible, on the other hand the rules should not allow to generate grammatically wrong sentences, or sentences the interpretation of which is obviously in conflict with the natural interpretation. And this latter requirement should be guaranteed by rules that are general and avoid exceptions. In the earlier approach the distributive reading of sentences (15)a/b can be expressed by using the floated quantifier *each*:

(19) a. Two companies each order a machine.
b. At least two companies each order a machine.

Although these sentences sound more natural than the constructions in (17)a/b the unrestricted use of *each* can lead to the following problems. First, using *each* as a general disambiguation marker that relates to the immediately preceding noun phrases would also approve the following sentence which would express the distributive reading of (15)c.

(20) *A company orders at least two machines each.

which is unacceptable in full natural language. Furthermore, using *each* in constructions with a possible licensing plural NP in subject position as in

(21) Two companies buy three machines each.

would – in full natural language – trigger distribution over the subject NP and not – as predicted by ACE – over the object NP. A further problem is that to express the distributive reading of an extracted direct object a partitive construction with *each of* is necessary anyway since the marker *each* cannot be put after the trace:

(22) John buys two books each of which Mary sells.

Therefore, in a previous version of the extension of ACE with plurals both the marker *each* and the partitive *each of* were allowed and the use of *each* in sentence final position was forbidden. This, however, conflicts with the requirement that ACE rules should be general and avoid exceptions. For these reasons it was decided to drop the disambiguation marker *each* from the language ACE. Accepting some stilted constructions with the partitive *each of* seemed less problematic than allowing an unrestricted use of the marker *each*. Further experiments with users of ACE must decide whether this decision is indeed the best compromise or whether re-introducing the marker *each* (with corresponding restrictions) would rather increase the acceptance of ACE as a controlled *natural* language.

Which Paraphrase?

The constructive disambiguation strategy of ACE states that the interpretations are reflected to the user in a paraphrase. The paraphrase is – without brackets – again valid ACE text and leads to exactly the same interpretation as the original text. Currently, there is no solution for paraphrasing the interpretation of plurals that fulfils these requirements. In particular, adding an optional *together* after a noun phrase to show to the user that the noun phrase is interpreted collectively is not a satisfactory option. The reason is that the elaboration marker *together* has more semantic content than just triggering an indeterminate collective reading (see section 4.5.4.4). Since ACE contains proper elaboration markers – including the marker *together* – the following two sentences are admissible in ACE but they are not logically equivalent:

(23) a. Two companies buy three machines.
b. Two companies together buy three machines.

Sentence (23)b implies sentence (23)a but not vice versa. Therefore, a paraphrase using *together* to visualize the (non-strict) collective reading would not be equivalent to the original ACE sentence. Also adding together after every collectively interpreted plural NP would lead to an inflation of markers. For example the sentence

(24) Five climbers who carry two tents use an oxygen bottle.

would have to be paraphrased as

(25) {Five climbers who [together] carry {two tents} [together]} [together] use {an oxy-

gen bottle}.

which shows a potential inflation of markers that make ACE paraphrases unnatural or even ungrammatical.

In the simple paraphrase provided by the DRoPs disambiguator I added meta-language information after the noun phrase to indicate how the system interpreted the sentence, e.g.

(26) a. two companies-[coll]
b. at least three machines-[distr]

To adopt this paraphrase option for ACE would violate the requirement that the paraphrase should again be valid ACE text. Also, this format of paraphrases would give ACE a touch of non-naturalness and formality that I wanted to avoid.

For these reasons I decided to give no paraphrase that indicates the collective reading but to expect the user to learn the respective interpretation rules. A paraphrase for the distributive reading is not necessary anyway since the reading has to be made visible by using *each of*.

7.3.2 Plural Constructions in ACE

The following sections describe some important construction and interpretation rules for plurals in ACE. Complete details can again be found in the ACE 4 manual, or in Appendix E of this thesis.

7.3.2.1 Construction Rules

Individual Denoting and Quantificational Plural NPs. ACE distinguishes individual denoting and quantificational singular and plural NPs. Individual denoting NPs are introduced by non-quantificational determiners, or can be realized as proper nouns or pronouns. Following are examples for both singular and plural NPs:

(27) Individual denoting NPs
a. NPs introduced by
indefinite determiners (*some*, *several*, *a few*, empty determiner, *a*, …)
definite determiners (*the*)
cardinality determiners (*two*, *at least 3*, *at most 7*, *all*, *many*, *few*, *one*, …)
b. proper nouns (*the Smiths*, *John*, …)
c. pronouns (personal, reflexive, possessive and relative pronouns)
(*they*, *themselves*, *their*, *their own*, *he*, *himself*, *who*, *which*, *that*, *each of which*…)

Quantificational plural NPs are introduced by quantificational determiners

(28) Quantificational NPs
a. quantificational plural determiners (*every*, *each*, *no*)

Simple, Complex and Modified Plural NPs. Individual denoting and quantificational plural noun phrases can be realized as simple, complex or modified NPs.

(29) Syntactic Types of Plural NPs
- a. simple plural noun phrases (*two men*, *no students*, *the children, they, the Smiths*)
- b. partitives (*each of the men*, *at most two of the children*)
- c. measurement constructions (*2 ounces of gold*, *3 pounds of apples*)
- d. coordination of individual denoting NPs (*some men and some women*, *John and Mary*)
- e. modifications of NPs of types a. – d. (*two expensive cars that John owns*, *a card of John and Mary*)

Plural noun phrases can occur in the same positions as singular noun phrases. Like singular noun phrases plural noun phrases can be modified by adjectives (*two young students*), relative clauses (*two men who lifted a table*), *of*-PPs (*the books of John and Mary*), possessive nouns (*the students' books*), appositions (*the customers John and Mary*). The detailed construction rules can be found in the Appendix E or in the ACE 4 manual.

Elaboration Markers. In ACE predefined elaboration markers can be put directly after a non-distributive plural noun phrase to detail the indeterminate collective interpretation of the NP with respect to the verb phrase. ACE distinguishes three elaboration dimensions that are expressed by predefined elaboration markers:

(30) Elaboration Markers
- a. temporal
  at the same time, *simultaneously*, *one by one*
- b. group-structures
  as a whole, *as a group*, *collectively*, *jointly*, *together*
- c. spatial
  at the same place

For detailed construction rules see the Appendix E.

Conjunction of NPs. Only individual denoting plural NPs can be coordinated. The resulting NP is again an individual denoting NP.

7.3.2.2 Interpretation Rules

Following are some examples for interpretation rules. The interpretation rules concern on the one hand the collective/distributive interpretation of NPs, on the other hand the resolution of syntactic ambiguities triggered e.g. by NP coordination. Again a complete list can be found in the Appendix E.

Collective and Distributive Readings. In full natural language individual denoting plural NPs are ambiguous between a collective and a distributive reading. Quantificational NPs never get a collective reading. The ambiguity in ACE is resolved by the following interpretation rule:

(31) Collective/Distributive Interpretation of NPs

a. Individual denoting plural NPs get a default collective reading.

b. Distributive readings have to be explicitly triggered by using partitive constructions with *each of*.

c. Quantificational NPs never get a collective reading.

d. Measurement NPs are always interpreted collectively.

For examples see sentences in (15) on page 305 above.

Elaboration Markers. Elaboration markers detail the indeterminacy of the collective reading of a plural NP with respect to a predicate along temporal, spatial or group-structure dimensions. The following interpretation rules hold:

(32) Elaboration Markers

a. The elaboration marker relates the maximal preceding NP to the predicate of which the maximal NP is a complement or an adjunct. A maximal NP includes all modifiers of the NP and all elements of a conjunction.

b. If two or more elaboration markers follow each other the markers are resolved from outside to inside.

In the example

(33) Five climbers who carry two tents together use an oxygen bottle.

the elaboration marker *together* relates the complex noun phrase *five climbers who carry two tents* with the event introduced by the predicate *use an oxygen bottle*. In the paraphrase this rule is expressed by adding (optional) brackets around noun phrases:

(34) {Five climbers who carry {two tents}} together use {an oxygen bottle}.

The elaboration marker relates to the noun phrase that is enclosed in the outer brackets. For more examples see the Appendix E.

Attachment Ambiguities with NP Coordinations. If NP coordinations occur together with other ACE constructions like relative sentences, *of*-PPs or possessives syntactic attachment ambiguities can occur. Here I give an example for the resolution of syntactic ambiguities triggered by coordination and relative sentences. In the Appendix E a complete list can be found.

If a relative sentence occurs after a noun phrase conjunction ambiguities are possible as to whether the relative pronoun refers to the whole coordinated noun phrase or only to the immediately preceding simple noun phrase. ACE resolves the ambiguity with the following interpretation rules:

(35) Relative Sentence after NP Coordination

a. A relative sentence after a noun phrase coordination refers to the immediately preceding noun phrase if this noun phrases agrees in number and object type with the relative pronoun.

b. If the immediately preceding noun phrase that is part of an NP coordination does *not* agree in number the relative pronoun refers to the whole coordinated noun phrase that agrees in number.

The sentence

(36) A customer has two Visacards and three Mastercards that are expired.

will be interpreted as

(37) A customer has two Visacards and {three Mastercards that are expired}.

In the example, the relative pronoun cannot refer to the whole coordinated noun phrase. To express this meaning users have to reformulate the sentence for example as

(38) A customer has two Visacards and three Mastercards. The Visacards and the Mastercards are expired.

Since in

(39) A customer has a Visacard and a Mastercard that are expired.

the relative sentence *that are expired* requires plural agreement it cannot refer to *a Mastercard*. The relative sentence therefore relates to the whole coordinated noun phrase which is shown in the paraphrase as

(40) A customer has {{a Visacard and a Mastercard} that are expired}.

It is also possible that the noun phrase conjunction occurs *within* a relative sentence.

(41) NP Conjunction within a Relative Sentence
If a conjoined noun phrase occurs *within* a relative clause the whole conjoined noun phrase belongs to the relative clause.

Therefore in ACE the sentence

(42) John sees a man who enters a card and a code.

is interpreted as

(43) John sees {a man who enters {a card and a code}}.

To express that John sees a man who enters a card and that John sees a code users have to reformulate the sentence for example by one of the following sentences.

(44) a. John sees a man who enters a card and sees a code.
b. John sees a code and a man who enters a card.

For the interpretation of (44)a ACE provides similar rules for VP coordination. Also, in Appendix E similar rules are formulated for further interactions of NP coordination with other constructions.

In general, the user is advised to avoid complex NP conjunctions in ACE.

7.4 The Attempto System in a Nutshell

7.4.1 System Architecture

The architecture of the Attempto system is summarized in Figure 14 on page 313. The overall structure of the Attempto system is similar the DRoPs system explained in section 5.4. The user inputs ACE text and possibly an ACE query. The Attempto Parsing Engine (APE) analyses and processes ACE text and ACE query deterministically by a unification based phrase structure grammar enhanced by feature structures. The output of this analysis is a paraphrase that is presented to the user to give feedback how ACE interpreted the sentence. Furthermore, ACE sentences are translated into flat DRSs that have the same structure as the DRSs used within the DRoPs system. The paragraph is then automatically translated into standard first-order logic. Like the DRoPs system the Attempto system contains an external knowledge-base consisting of domain-independent axioms, evaluable functions and predicates, and optionally a set of domain-specific axioms. The FOL formulae generated from the user input plus the FOL-axioms derived from the knowledge base are then input to the ACE reasoner RACE which is an extension of Otter and Satchmo, resp. As described in chapter 6 RACE finds all proofs and additionally gives a justification of the proof again in ACE. Thus the user interacts with the system only on the level of ACE. More details about the reasoner can be found in chapter 6 of this thesis.

7.4.2 Implementation of Plurals in the Attempto System

In section 5.3 and 5.4 I gave a description of the implementation of plural disambiguation in the DRoPs system. The current implementation of plurals in the Attempto system is using this implementation with minor parameter re-settings for the disambiguation algorithm. More concretely, the output of the ACE parser corresponds basically to the underspecified logical form output by the DRoPs parser. The constructive disambiguation of ACE is then applied to this output. In contrast to the DRoPs disambiguation no scope-reorderings take place, i.e. the order of the NPs in the list determines the scoping of the NPs. Furthermore, calculations about collective/distributive readings are simplified as follows. The reading value of individual denoting plural noun phrases that are not occurring within an *each of* construction is set as collective. If the noun phrase is used in an *each of* construction the reading value is set as distributive. The result of the constructive ACE disambiguation component is a fully specified NP list as described in section 5.4.3. This specified NP list is then input to the same DRS construction component that was used within the DRoPs system.

The previous implementation of the Attempto parser (Schwitter 1998) used an approach where parsing and DRS construction were performed concurrently within the parser. I have already addressed the disadvantages of this approach in general in section 5.4.3, pp. 268. Although, for ACE the splitting of parsing and DRS construction would not be necessary I still consider my approach to be more flexible and re-usable, in particular as far as extensions and modifications

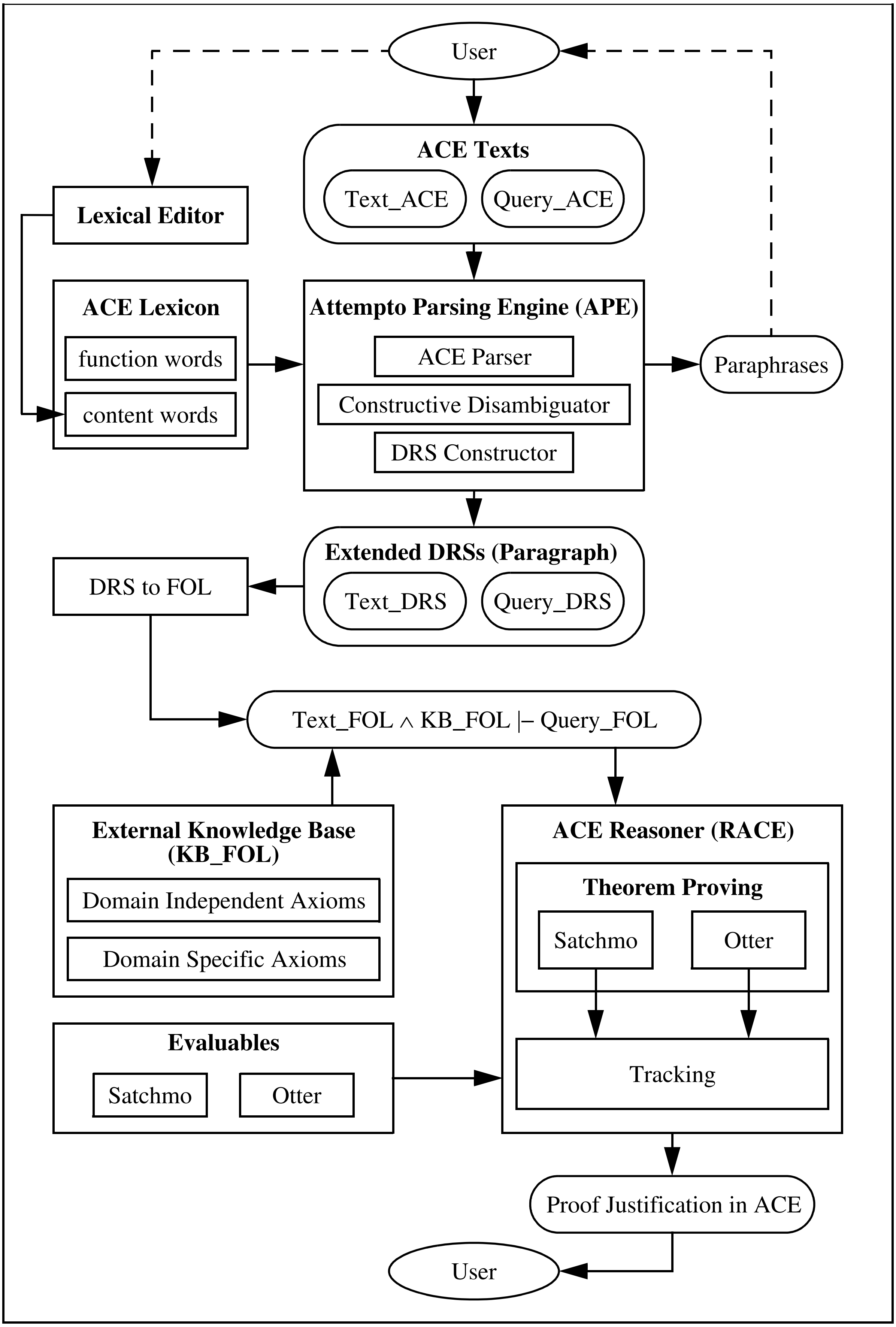


Figure 14 Architecture of the Attempto System

of the ACE language are concerned.

7.4.3 Evaluation

The prototypical implementation of the Attempto system (see also www.ifi.unizh.ch/attempto) is very promising and attracts many researches interested in using ACE for their projects. Although the core of the Attempto system has been implemented there are several extensions that still need to be developed to make the Attempto system a “tool”. These extensions concern for example a user-friendly web interface including a web-based lexical editor or possibilities to add additional domain axioms. Furthermore, the efficiency of the Attempto reasoner has to be improved and additional techniques for question answering have to be developed.

As far as plurals are concerned the prototypical ACE implementation shows that plurals can be managed in a practical natural language understanding system. The techniques presented in previous parts of this thesis, e.g. the flat first-order DRS representation, the definition of first-order axioms needed for theorem proving and the assumption of an indeterminate collective reading constitute an important foundation for the integration of plurals in ACE. Furthermore, I have shown that only minor parameter re-settings were necessary to use the DRoPs implementation also for the Attempto system. This is a further evidence for the generality and re-usability of my DRoPs approach to the representation and disambiguation of plurals. Vice versa, I have re-used the core reasoning component of the Attempto system (RACE) for the DRoPs system, showing that the DRoPs approach allows re-use of existing tools.

7.5 Conclusions and Further Research

The previous sections gave an overview of Attempto Controlled English (ACE) with a particular focus on the integration of plurals into ACE. ACE is originally intended as a software specification language that combines the familiarity of natural language with the rigour of formal specification languages. The use of formal specification languages allows to make use of formal specification methods to compose, query and execute formal specifications. Furthermore, I gave a brief overview about the prototypical Attempto implementation.

Several applications of ACE have been completed: a specifications of an automated teller machine, Kemmerer’s library data base (Schwitter 1998), Schubert’s Steamroller and Kowalski’s subway example. Furthermore, ACE was used to formulate data base integrity constraints and it was used as an interface to the model generator EP Tableaux (Fuchs, Schwertel and Torge 2000).

Originally designed as a software specification language ACE has by now received a great deal of attention from other fields of research. The main advantage of ACE is that it can serve as a natural language interface to applications that require formal input: many researches have realized that ACE is a much more convenient interface language than first-order logic or other visibly formal languages. Further motivation to use the Attempto system arose from the development of the Attempto reasoner RACE. RACE performs automatic reasoning on ACE sentences and outputs the results again on the level of ACE.

Current or planned applications of the Attempto system concern the acquisition of requirements (University of Erlangen), the control of agents/robots by Fluent Calculus (University of Dresden, Daw Elbait 2003, to appear), planning (University of Uppsala, Swedish National Defense College), knowledge assimilation (University of Munich), and – most importantly – the use of ACE as a user-friendly interface to Semantic Web reasoning languages (cf. EU Network of Excellence REWERSE – Reasoning on the Web with Rules and Semantics, http://rewerse.net).

Suggested or partially investigated applications have been the use of ACE for the description of medical diseases and their courses (University of Uppsala). To investigate the courses of diseases and to compare different diseases theorem proving techniques are required. A further suggested application of ACE concerns the use of ACE for the synthesis of web sites and for knowledge sharing between web sites (University of Barcelona, University of São Paulo, University of Edinburgh). Furthermore, the use of ACE as an interface to a synthesizer of constraint logic programs (University of Uppsala) has been suggested. For this purpose further extensions of ACE would be required to represent mathematical structures and operations on these. These extensions could eventually lead to a dialect of ACE that can be used for mathematical text books. A further potential application is to use ACE for legal texts, to formalize standards or ontologies (Sowa 2000). Other researches have shown interest in using ACE for teaching logic. The original application of ACE as a software specification language is currently considered by Siemens Germany.

The variety of possible applications and the increasing interest of other researches show that Attempto provides a promising approach to problems that require precise, computer-processable yet natural and readable input the semantic representation of which allows for logical post-processing. The intended applications require, however, further development of ACE and its tools. For example, user-feedback should be improved. Also the structuring of large specifications is to be further investigated and implemented. Further directions of research concern the development of tools to execute specifications. Executing a specification means to demonstrate its logical and temporal structure. Execution is important to validate specifications written in ACE and to give users immediate feedback of the behaviour of the future software. Executing a specification can be considered as prototyping the future software. Two execution modes are planned – the batch mode and the discourse mode. In the batch mode a user submits a predefined ACE test case to the specification that is then executed, while in the discourse mode the user interactively enters ACE commands to control, for instance, the behaviour of an agent or robot. The discourse mode requires an extension of ACE by the imperative mood. The execution of ACE specifications also requires that temporal reasoning is included in the system. Above that, the language has to be extended by variables, and by relations and functions using these variables.

8 Conclusion and Further Research

8.1 Summary and Main Achievements

This dissertation develops a computational proof-theoretic approach to plural semantics that can be used in practical applications performing logic-oriented natural language understanding. This computational approach includes the following main parts: the definition of a computationally suitable flat first-order semantic representation for many difficult plural constructions, the development of a computationally tractable disambiguation algorithm, and the realization of automated reasoning with plurals in natural language. Furthermore, the main ideas of the thesis are implemented within two logic-oriented text understanding systems: DRoPs (Disambiguating and Reasoning with Plurals) that is designed to disambiguate full natural language, and Attempto that processes controlled English.

Whereas many existing formal theories to plural semantics rely on a higher-order model-theoretic semantic representation I chose a first-order proof-theoretic approach to plural semantics thus showing that a trade-off between expressive power, computational tractability and reusability can be achieved. Computational plural semantics generally faces the problem of a combinatorial explosion of semantic ambiguities triggered by plurals. The development of my disambiguation component is motivated by the observation that there is – according to my knowledge – no satisfactorily worked out computationally suitable disambiguation algorithm for plurals. The algorithm proposed in this thesis relies on structurally available disambiguation information and takes no recourse to computationally not manageable contextual knowledge. The algorithm extrapolates factors developed by other researchers for scope disambiguation and combines these with my own observations concerning plural disambiguation. A further important theoretical aspect of my disambiguation approach is my conviction that plurals can often be analysed as being indeterminate instead of ambiguous thus reducing the number of possible readings. Finally, the thesis shows how automated reasoning with plurals can be implemented by modifying existing theorem provers and extending them with auxiliary first-order axioms defining the properties of plural objects. A main achievement of my approach with respect to reasoning is to practically implement general and re-usable first-order axioms for plurals and showing how they support the desired inferences.

Furthermore, in this thesis I integrate my proposals for representation, disambiguation and reasoning within practical applications. This practical integration is neatly possible due to my choice of a first-order proof-theoretic approach. The practical applications can for example be used to support the logical analysis of natural language texts or to provide a user-friendly interface to applications that require a natural yet precise and computer-understandable input.

The wide range of the tasks did not allow me to investigate all constructions and problems related to plurals in adequate detail. Yet, my proposal is designed to be flexible enough to smoothly integrate new phenomena. Although much remains to be done a main achievement of this thesis lies in combining methods from formal semantics, computational linguistics and automated reasoning to propose a computationally oriented approach to plural semantics that can be used in practical applications. I am not aware of other approaches that integrate a corresponding wide range of requirements and that are worked out and practically tested in similar detail.

8.2 Evaluation and Further Research

For each of the main parts of my approach, viz. representation, disambiguation, reasoning and applications, I want to give a brief evaluation, address problems and issues for further research.

8.2.1 Representation

My thesis defines a first-order, thus computationally tractable, solution for many difficult plural phenomena that are often taken as an argument to use higher-order representations. For example, the thesis describes the semantics of non-monotone increasing quantifiers like *at most n*, *exactly n* using explicit maximality conditions, represents vague and context dependent determiners like *few*, *many* or proportional quantifiers like *most*. Furthermore, computationally suitable representations for measurements, partitives, coordination, collective and distributive adjectives and other phenomena are given. The representations are designed to support the desired and block the undesired inferences. Thereby, the representations try to avoid over-precision that would make automated reasoning practically very inefficient. Yet, the representations allow to integrate explicit additional elaborations like part-structure modifiers (e.g. *as a whole*, *simultaneously*). Furthermore, if certain application domains require more detailed inferences the representation can be extended by additional meaning postulates formulated as auxiliary first-order axioms for certain words or word classes, e.g distributive subentailments of certain verb classes. In general, however, the coverage of the representation has to be extended. A topic of future research would for example be to find more precise lexical regularities triggered for example by certain verb classes, by certain adjectives, or types of noun phrases. In particular the representation could be extended by a more precise verb and event semantics also including reasoning about eventuality parts. I consider it particularly interesting how to *automatically* extract and integrate more information from existing lexica that could – if desired – make the representation more precise.

Technically, the flat first-order representation allows us to integrate first-order auxiliary axioms for plurals, identity, mathematical axioms and others. Also, the first-order approach allows us to use off-the shelf first-order theorem provers that by now have reached a high level of maturity. This shows the generality, flexibility and re-usability of my approach. However, choosing a flat first-order notation also has the consequence that the discourse representation structures contain many conditions which may reduce the efficiency of the associated theorem-provers. It therefore needs to be investigated whether the notation can be formulated more compactly. Also the status of the maximality conditions is still a matter of consideration. A further interesting line of investigation would be to relate the currently defined representation to other important knowledge representation language and thus to have access to a wider range of existing applications.

The currently intended applications assume a disambiguated text that logical reasoning is performed on. This involves defining fully disambiguated logical representations. My approach does not attempt a solution for incomplete parsing, partial disambiguation or partial semantic representation. This is argued to be necessary to make a system robust and flexible with respect to processing input not covered by the rules of the system. The underspecified representations introduced in chapter 5 could, however, be used as a starting point for representing partially specified logical representations. However, as soon as partially specified representations are allowed one also has to define other reasoning methods that do not rely on fully specified logical forms.

8.2.2 Disambiguation

The thesis develops a plural and a simple scope disambiguation algorithm that uses only information that can be automatically extracted from the lexicon or the structure of a sentence. The algorithm performs a radical reduction of possible to a small number of plausible readings and ranks the plausible readings so that – if desired – a best reading can be automatically selected. This strategy is important if – as in this thesis – an application requires disambiguated semantic representations for further logical processing. I chose a global strategy for disambiguation meaning that the ambiguity – in general – is not triggered by lexical items alone but can be a result of complex factors that interact when elements of a sentence or phrase are combined. The global strategy is practically realized by a complex system of features structures. The disambiguation algorithm starts with an underspecified representation, viz. an underspecified store, and gradually instantiates the relevant features when more information becomes accessible. Only after the features are fully instantiated does the DRS construction algorithm generate discourse representations from the stores. This is of practical advantage since it allows to use just one grammar and yet be able to generate different readings of a semantically ambiguous sentence.

Nevertheless, the current algorithm requires several extensions and improvements. First of all, the coverage has to be extended to include e.g. vertical scoping within complex noun phrases, to resolve (plural) anaphora, to include other scope bearing elements like adverbs. Furthermore, the currently defined parameters need to be fine-tuned with respect to a larger set of empirical data. This involves possible adaptations of the numeric parameter values. An interesting area for further research would be to define semantically tagged corpora where collective/distributive and scope ambiguities are resolved and to test and improve the proposed algorithm with respect to these test suites. In creating these test suites methods should be developed to distinguish two disambiguation sources: structural factors that hold for neutral contexts and factors that relate to human knowledge about the world, the context or about particular lexical items. Only the first factors are currently accessible to my algorithm, however, the latter factors are – for human disambiguation – much more important and further research to automatically access these factors should be pursued.

A more theoretical line of research would consist of trying to find explanations for the influence of the disambiguation parameters proposed in this thesis. Currently the parameters may seem arbitrary and ad hoc and it would be nice to have a theory of these parameters that goes beyond mere trial and error for certain applications domains.

For robust processing of full natural language texts it would again be important to investigate the role of underspecified or partially specified representations as opposed to fully specified representations that are the result of my disambiguation algorithm. This involves the investigation whether it is always necessary to fully disambiguate or whether certain semantic ambiguities need no resolution for further processing. The distinction of real semantic ambiguity from mere indeterminacy already points into this direction. However, certain ambiguities remain and for the applications intended in this thesis, viz. interfaces to logic oriented applications, these real ambiguities have to be resolved for further processing. I have followed a traditional approach to disambiguation that combines methods from both the *Generate and Test* approach and the *Underspecification* approach. If sufficiently disambiguated test suites existed one could also work on other disambiguation techniques using statistical and other methods. I have not pursued these directions within my thesis.

8.2.3 Reasoning

Natural language understanding requires methods to reconstruct intuitive reasoning processes associated with natural language texts. This thesis shows how *automated* reasoning with natural language plurals can be implemented using a proof-theoretic first-order approach. A further advantage of the suggested approach is that the results of the reasoning process are again reported on the level of natural language. This approach makes formal methods accessible to untrained users. For example, the reasoning component supports the automatic analysis of technical texts, shows that natural language can function as a user-friendly interface to formal applications, or, in general, demonstrates that natural language can be used as a suitable knowledge representation language for formal applications.

Currently, the reasoner is only tested for relatively small examples and its efficiency has to be increased to scale up to larger texts. It may turn out that the proposed approach has to be modified to be applicable to large scale applications. For example, the reasoning may turn out to be too deep for certain applications. However, whenever strict logical reasoning for a number of sentences is required the proposed deep and logically correct reasoning technique is required. A further concrete problem lies in the fact that some of the auxiliary plural axioms have to be reformulated to increase efficiency of the reasoning process. Also, the instantiation problems that we encountered when using the reasoner Otter require further investigation.

Topics of further research could also be to include further machine-processable knowledge, e.g. existing ontologies, via auxiliary axioms into the reasoning component. A further important task would be to include axioms for temporal and spatial reasoning. The reasoning method proposed in this thesis requires fully disambiguated texts. To increase robustness other complementing reasoning techniques, e.g. reasoning with underspecified or partially specified representations, would have to be investigated. Also, probabilistic methods or default reasoning could be investigated.

In some current computational semantic applications logical reasoning is not only applied *after* the text has been disambiguated but is itself used to support disambiguation. I have not considered this option for plural disambiguation.

As discussed in chapter 6 question answering is currently simulated by logical reasoning. This approach is not suitable for certain types of questions, for example 'How many …?' questions or questions involving universal quantifiers ('Does every …?'). Further research is required to test whether question answering can be improved based on the models that are generated by the Satchmo core of the reasoner if a set of clauses is satisfiable.

8.2.4 Applications

I have used my plural approach within two practical implementations performing text understanding. The DRoPs system currently parses only a limited fragment related to plural problems. In the long run the coverage has to be increased and techniques to deal with non-parsable input have to be included. As discussed above this relates to the theoretical question of what role underspecified or partially specified input plays for further semantic processing and reasoning. The Attempto system is easier in this respect since Attempto Controlled English is a *controlled* natural language and anything that cannot be parsed according to the construction and interpretation rules is – as intended – not part of the language ACE. However, to make Attempto more user friendly a number of extensions are still necessary. For example, we need to provide techniques for structuring large documents and we have to improve explanatory feedback to the user if a sentence cannot be parsed. Also the integration of complementary notations like graphical or algebraic notations that certain applications require have to be investigated. Technically, in Attempto problems of temporal reasoning are only partially solved. Also, further reasoning techniques have to be investigated to better support the analysis of ACE specifications, for example hypothetical reasoning ('What happens if …?'), abductive reasoning ('Under which conditions does … occur?'), and the execution of ACE specifications using ACE scenarios.

To test the adequacy and acceptance of ACE the language has to be practically tested in different applications which may require a further modification of the current coverage. Currently, different applications of ACE are investigated, for example to use the Attempto system for knowledge representation and reasoning within the Semantic Web, to control robots/agents with ACE, to employ ACE as an interface to planning applications, or to describe software and hardware capabilities in ACE. The variety of possible applications and the increasing interest of other researches show that Attempto provides a promising approach to problems that require precise, computer-processable yet natural and readable input the semantic representation of which allows for logical post-processing.

Appendix

A Predefined DRS Conditions

This part of the Appendix gives a complete list of predefined relation symbols that the DRS language uses. Table 15 on page 324 lists conditions used in the singular fragment. The conditions introduced for the representation of plural constructions will be listed in Table 16 on page 325. The first row gives the syntax of the predefined relations using the following convention for variables: variables like X, Y, Z (with possible subscripts) stand for discourse referents introduced by objects, variables starting with E or S stand for eventualities, variables starting with K must be instantiated to constants taken from a closed set of predefined constants (as specified in the second row), variables starting with L are instantiated to constants derived from the lexical entries.

Predefined Relation r	Predefined Constants K	Example
object(X,L)		object(X,man)
object_type(X,K)	person | time | object	object_type(X,person)
named(X,L)		named(X,'John')
relation(X,L,K,Y)	of	relation(X,husband,of,Y)
property(X,L)		property(X,red)
property(X,L,Y)		property(X,bigger_than,Y)
property(X,L,Y,Z)		property(X,fonder_of_than,Y,Z)
predicate(E,L,X)		predicate(E,sleep,X)
predicate(E,L,X,Y)		predicate(E,lift,X,Y)
predicate(E,L,X,Y,Z)		predicate(E,give_to,X,Y,Z)
modifier(E,K,L,X)	location | origin | direction | time | start | end | duration | instrument | comitative | manner | …	modifier(E, direction, into,X)
modifier(E,K,none,upwards)	location | direction | time | frequency | duration | manner | …	modifier(B, direction, none,upwards)
relation(X,L,of,Y)		relation(X,friend,of,Y)
tense(E,K)	present | past	tense(E,present)
temporal_order(E_1,K,E_2)	after | before | overlap | part_of | include | same | different | any	temporal_order(E_1,after,E_2)
query(X,K)	who | what | how_many	query(X,who)
query(E,K)	when | where | …	query(E,when)

Table 15 Predefined Relation Symbols used in the Singular Fragment

Predefined Relation r	Predefined Constants K	Example
structure(X,K)	dom | atomic | group | mass | e_dom | event | state	structure(A,group) structure(E,event)
part_of(X,Y)		part_of(A,B)
e_part_of(E_1,E_2)		e_part_of(A,B)
proper_part_of(X,Y)		proper_part_of(A,B)
sum_of(X,[X_1, …, X_n])		sum_of(A,[B,C])
quantity(X,K_1,Y,K_2)	K_1:cardinality | weight | length | volume | … K_2: count_unit | kg | cm | liter | …	quantity(A,cardinality,B,count_unit) quantity(X,weight,Y,kg)
value(Y,K, I)	eq | leq | geq | greater | less	value(A,eq,3)
property(Y,K)	many | few | …	property(A, many)
is_equal(X,Y)		is_equal(A,B)
relation(X,K,Y)	eq | leq | geq | greater | less | most | few | many | all | …	relation(X,most,Y)
maximal(X,context,Y)		maximal(X,context,drs(U1,C1) => drs(U2,C2))
elaboration(E,X,K_1,K_2)	K_1: group_structure | time | space K_2: same | different | coll | distr | …	elaboration(E,A,group_structure, coll)

Table 16 Predefined Relation Symbols added for the Plural Fragment

B Meaning Postulates for 'Same' and 'Different'

The following formulae show how exact meaning postulates for elaboration markers could be realized (see also Moltmann 1997, pp. 138)

(1) *same*

a. John and Mary used the same tents.

b. Meaning Postulate

$$\begin{array}{l}\forall E \forall P \forall Y \forall T(\text{predicate}(E,P,Y,T) \wedge \text{elaboration}(E,T,\text{same},Y) \rightarrow \\ \forall Y_1 \forall Y_2 \forall E_1 \forall E_2((\text{proper\_part\_of}(Y_1,Y) \wedge \text{proper\_part\_of}(Y_2,Y) \wedge \\ \quad \text{proper\_e\_part\_of}(E_1,E) \wedge \text{proper\_e\_part\_of}(E_2,E) \wedge \\ \quad \neg(\text{is\_equal}(E_1,E_2)) \wedge \\ \quad \exists T_1 \exists T_2(\text{part\_of}(T_1,T) \wedge \text{part\_of}(T_2,T) \ \& \\ \qquad \text{predicate}(E_1,P,Y_1,T_1) \wedge \text{predicate}(E_2,P,Y_2,T_2))) \\ \qquad \rightarrow \text{is\_equal}(T_1,T_2)))\end{array}$$

c. Verbal Description

There is an event (E) of using tents (T) by John and Mary (Y) such that for any two distinct proper parts E_1 and E_2 of E, the part of the tents used in E_1 (by a proper part of John and Mary) is the same as the part of the tents used in E_2 (by a proper part of John and Mary).

(2) *different*

a. John and Mary used different tents.

b. Meaning Postulate

$$\begin{array}{l}\forall E \forall P \forall Y \forall T(\text{predicate}(E,P,Y,T) \wedge \text{elaboration}(E,T,\text{different},Y) \rightarrow \\ \forall Y_1 \forall Y_2 \forall E_1 \forall E_2((\text{proper\_part\_of}(Y_1,Y) \wedge \text{proper\_part\_of}(Y_2,Y) \wedge \\ \quad \text{proper\_e\_part\_of}(E_1,E) \wedge \text{proper\_e\_part\_of}(E_2,E) \wedge \\ \quad \neg(\text{is\_equal}(Y_1,Y_2)) \wedge \neg(\text{is\_equal}(E_1,E_2)) \wedge \\ \quad \exists T_1 \exists T_2(\text{part\_of}(T_1,T) \wedge \text{part\_of}(T_2,T) \wedge \\ \qquad \text{predicate}(E_1,P,Y_1,T_1) \wedge \text{predicate}(E_2,P,Y_2,T_2))) \\ \qquad\quad \rightarrow \neg(\text{is\_equal}(T_1,T_2))))\end{array}$$

c. Verbal Description

There is an event (E) of using tents (T) by John and Mary (Y) such that for any two distinct proper parts E_1 and E_2 of E, the part of the tents used in E_1 (by a proper part of John and Mary) is different from the part of the tents used in E_2 (by a different part of John and Mary).

C English Technical Texts

The following texts have been used for empirical testing of the approach.

C.1 Sources

- Parts of an IBM AS/400 Documentation (Format SGML) (ca. 30 MB)
  (kindly provided by IBM Deutschland Informationssysteme GmbH, SW NLS 1, Dept. 2076, Bldg. 71034-91, Stuttgart, 1998)
- Scientific abstracts provided by the U.S. Department of Energy (ca. 40 MB)
- International Telecommunications Union CCITT Handbook "The Blue Book" (ca. 10 MB)
- ScanWorX User's Guide (Optical Character Reader) (260 KB)
- not electronically prepared (software-)specification texts:
  - "Steam Boiler Control" Specification
  - Specification of a "Bug Log" Program
  - …
- Google-Queries

C.2 Electronic Preparation and Investigation

After appropriate preprocessing of the texts I used the statistical part-of-speech tagger "TnT" (*Trigrams'n'Tags*) developed by Thorsten Brants (see `http://www.coli.uni-sb.de/~thorsten/tnt/`) to tag the data with the BNC tagset. This tagging was only approximate since I didn't train the tagger on the corpora. Since the results were sufficiently good for my purposes I did not optimize the tagging results. After some post-processing of the texts I could then investigate the texts using the "IMS Corpus Workbench (CWB)" (see `http://www.ims.uni-stuttgart.de/projekte/CorpusWorkbench/`). The CWB is a very elegant and useful workbench for full-text retrieval from large textual resources. It provides a powerful query language that allows to extract different types of collocations. For example it allows the following (simple) query

(1) Example Query within IMS Corpus Workbench

```
[pos="MC"][]{0,1}[pos="NN2"];
```

This query extracts all occurrences where a cardinal number ("MC") is followed by 0 or 1 arbitrary words which is followed by a plural common noun ("NN2"). You can choose the size of the context that is printed out as a result. Furthermore the number of occurrences is counted. A typical output to this query is given in Table 15.

Since the results of the query need human interpretation the tool did not help to find out preferences for collective/distributive readings automatically. The tool was mainly used to look for typical phenomena, to estimate their frequency and to check theoretical results against a number of examples.

```
#----------------------------------------------------------------
# User:    Uta Schwertel
# Date:    Wed Jan 20 17:49:45 1999
# Corpus:  IBM AS/400
# Name:    IBM:Last
# Size:    10208 intervals/matches
# Context: 60 characters left, 60 characters right
#
# Query: IBM; [pos="MC"][]{0,1}[pos="NN2"];
#----------------------------------------------------------------


ocess refreshes the target file . This parameter consists of <three
elements> . Element 1 : SQL statement SQL-statement The SQL state-
ment

ontrol tables and two indices for the new registration . The <three
tables> are the Change Data table , the Pruning Control table , and

Pruning Control table , and the Critical Section table . All <three
tables> are created in the library specified in the CTLLIB paramete

ternate name for the protocol . You can specify a maximum of <4
aliases> . No checking is done to ensure that an alias is unique .

s JOHN and RCHAS100. Try to limit your domain name labels to <12
characters> . Shorter labels are easier to remember . It is a com-
mon pr

estrict . Enter an option 4 (ENDTCPCNN) for each . There are <two
independent sets> of ports . One set is for TCP processing and the
other is f

 Control Protocol/Internet Protocol (TCP/IP) configuration . <Five
parameter values> uniquely define a route . These values are the
route destin

) This command indicates that the next time FTP is started , <5 FTP
servers> will start automatically . Example 4 : Changing the FTP
Att
```

Figure 15 Sample Output IMS Corpus Workbench

D Lexica

The following lexica have been used to search for computationally available lexical disambiguation information.

D.1 COMLEX

COMLEX Syntax (see `http://nlp.cs.nyu.edu/comlex/`) is a monolingual English Dictionary consisting of 38,000 head words intended for use in natural language processing. This dictionary was developed by the Proteus Project at New York University under the auspices of the Linguistic Data Consortium (LDC). It contains exceptionally detailed syntactic information and is now a widely-used lexical resource. For more information consult Wolff et. al. (1998).

D.2 WordNet

WordNet® (see `http://www.cogsci.princeton.edu/~wn/`) is an on-line lexical reference system whose design is inspired by current psycho-linguistic theories of human lexical memory. English nouns, verbs, adjectives and adverbs are organized into synonym sets, each representing one underlying lexical concept. Different relations link the synonym sets. WordNet was developed by the Cognitive Science Laboratory at Princeton University under the direction of Professor George A. Miller (Principal Investigator).

D.3 Verb Classification by Beth Levin

In her rich reference work, Beth Levin (1993) (see also `http://www-personal.umich.edu/~jlawler/levin.html`) classifies over 3,000 English verbs according to shared meaning and behaviour. Levin starts with the hypothesis that a verb's meaning influences its syntactic behaviour and develops it into a powerful tool for studying the English verb lexicon. She shows how identifying verbs with similar syntactic behaviour provides an effective means of distinguishing semantically coherent verb classes, and isolates these classes by examining verb behaviour with respect to a wide range of syntactic alternations that reflect verb meaning.

E Plurals in ACE – Reference Manual

The following sections describe the construction and interpretation rules for plurals in ACE in some detail. They build upon the construction and interpretation rules for singulars in ACE. The sections are – with minor modifications – taken from the ACE 4 manual (Schwertel, Fuchs and Höfler 2003) in which a complete description of the current status of the language ACE is given. Here only the parts relevant for plurals are given. For complete reference, the interested reader should consult the ACE 4 manual the most recent version of which is available at `http://www.ifi.unizh.ch/attempto/`.

E.1 Overview

Construction

ACE is extended with plural noun phrases. There are several types of plural noun phrases:

- simple plural noun phrases (*two men*, *no students*, *the children*)
- coordinated noun phrases (*some men and some women*, *John and Mary*)
- partitives (*each of the men*, *at most two of the children*)
- measurement constructions (*2 ounces of gold*, *3 pounds of apples*)
- plural pronouns (*they*, *them*, *themselves*, *their*, *their own*)

Plural noun phrases can occur in the same positions as singular noun phrases in ACE.

Like singular noun phrases plural noun phrases can be modified by the following additional elements:

- adjectives (*two young students*)
- relative clauses (*the men who each lifted a table*)
- *of*-PPs (*two customers of John*, *the books of John and of Mary*)
- possessive nouns (*the students' books*)
- appositions (*the customers John and Mary*)

Plural noun phrases can additionally be modified by so-called elaboration markers.

- elaboration markers (*one by one*, *simultaneously, together, as a whole*)

Elaboration markers give additional information about the interpretation of the plural noun phrase with respect to the verb phrase or to other elements of the sentence. Note that in ACE 4 floating quantifiers like *each* are not allowed for use as disambiguation markers.

Interpretation

In full natural language sentences with plural noun phrases can be ambiguous between collective, distributive and possibly other (e.g. mixed) readings. In ACE only collective and distributive readings are distinguished. Furthermore, not all plural noun phrases can get a collective reading. ACE distinguishes between individual denoting plural noun plural phrases (indefinite, definite and cardinality NPs) that can in principle be interpreted collectively, and quantificational noun phrases that always get a distributive interpretation.

In full natural language, multiple occurrences of plural noun phrases in a sentence can lead to an explosion of interpretations since each of the noun phrases can be read collectively and distributively. Furthermore, distributive readings add scope ambiguities due to their inherent universal force. In ACE, the disambiguation of plural noun phrases is based on the following assumptions:

- ACE only distinguishes between collective and distributive readings. Mixed interpretations are treated like collective interpretations expressing a complex event.
- Surface order determines quantifier scope.
- Individual denoting plural noun phrases are assigned a default collective reading if there are no overt syntactic markers triggering the distributive reading.
- The distributive reading is expressed by adding *each of* in front of the noun phrases.
- The default collective reading is *not* indicated to the user in the paraphrase. The user has to learn and remember the respective interpretation rule.
- If users wish to express a different interpretation they can reformulate the sentence. A main strategy to override the default collective interpretation is to use constructions with *each of*.

The ACE style guide furthermore recommends to avoid the use of plurals as much as possible since plurals may lead to ambiguities that are not recognized by the untrained users of ACE. Users should instead look for equivalent re-formulations in the singular. Ideally, these re-formulations should describe the situation as explicit as possible. Since these re-formulations are not always possible users should formulate sentences with plurals as clearly as possible, e.g. by avoiding conjunctions of plural noun phrases or avoiding (nested) relative sentences.

E.2 Plural Noun Phrases in ACE

E.2.1 Classification

There are two main types of plural noun phrases in ACE:

- individual denoting plural noun phrases:
  introduce objects into the domain of discourse and can – in principle – get a collective readings. Individual denoting noun phrases are further subclassified into four classes:
  - indefinite noun phrases
  - definite noun phrases
  - cardinality noun phrases
  - pronouns
- quantificational plural noun phrases:

are always interpreted distributively

Each type of plural noun phrase can be realized as a

- simple plural noun phrase, or as a
- complex plural noun phrase.

Simple plural noun phrases are introduced in section E.2.2, complex plural noun phrases in sections E.2.3 to E.2.6.

E.2.2 Simple Plural Noun Phrases

Construction

Simple plural noun phrases are proper nouns, pronouns or noun phrases that consist of a determiner and a plural noun. More concretely, the following types of simple plural noun phrases are distinguished:

Indefinite Plural Noun Phrases. Noun phrases that are introduced by one of the following determiners are called indefinite plural noun phrases in ACE. These noun phrases introduce a new object into the domain of discourse.

(1) Indefinite determiners in ACE
some (plural)
a few
several
the "empty" determiner needed for bare plurals (e.g. *men*, *tables*)

Definite Noun Phrases. Definite plural noun phrases are introduced by definite determiners

(2) Definite determiners in ACE
the (plural)
the two, *the three*

or are realized as

(3) Plural proper nouns in ACE
the Alps

Note, that plural proper nouns often begin with a definite determiner.

Cardinality Noun Phrases. Cardinality noun phrases can be introduced by one of the following types of determiners:

(4) absolute non-context-dependent determiners
two, *three*, …*twelve*, 13,14,15, … (note that *one* is also allowed)
at least n
more than n
exactly n
at most n
less than n
not more than n

(5) absolute context-dependent determiners

many

few

(6) proportional context-dependent determiners

most

most of the

many of the

few of the

more than half of the

(7) *all*

Note, that in this classification the terminology "cardinality determiner" is also applied to determiners like *many* or *most* which do not contain explicit numerals.

Quantificational Noun Phrases. In addition to the following singular quantificational noun phrases

(8) Singular Quantificational Noun Phrases:

every, *each*

not every

no (sg)

in ACE 4 it is additionally possible to use

(9) Plural quantificational Noun Phrases

no (pl)

to form a plural quantificational noun phrase.

Pronouns. In ACE 4 the following plural pronouns are admissible:

(10) Plural Pronouns

personal pronouns (*they*, *them*)

reflexive pronouns (*themselves*)

possessive pronouns (*their*, *their own*)

Pronouns are interpreted anaphorically as is described in section E.7. Note that the reciprocal pronoun *each other* is not allowed in ACE since it creates many syntactic and semantic ambiguities.

Negation of Quantifiers. ACE also contains some negated form of simple quantifiers, e.g. *not every*, *not all*, *not many*, *not more than n* etc.

Interpretation

General. Individual denoting plural noun phrases get a default collective reading. The distributive reading has to be expressed by *each of* in front of the respective noun phrase and – if the noun phrase occurs as subject – using a singular morphology for the verb.

Quantificational noun phrases never get a collective reading and do not introduce objects into the domain of discourse. Note that in ACE the sentence

(11) No men enter a card.

is inconsistent with both a collective and a distributive reading of

(12) Some men enter a card.

Following are some specific interpretation notes for the different types of NPs.

Indefinite Plural NPs. Indefinite noun phrases introduce a new object into the domain of discourse. If the determiners *some*, *several*, *a few* are used the cardinality of the object is greater or equal than two. The determiners induce no further restriction on the cardinality of the object. A bare plural like *cards* has the same meaning as *at least one card*.

Definite Plural NPs. Plural definite NPs have the same two uses like singular NPs in ACE:

- anaphoric use
- independently referring use

The interpretation of anaphorically used definite plural noun phrases is treated below in E.8. When the definite plural noun phrase is *not* used anaphorically it introduces a new group object into the specification. The collective/distributive interpretation rules for these noun phrases are analogous to those for indefinite plural noun phrases.

Plural proper nouns are always interpreted collectively.

Absolute Non-Context Dependent Cardinality NPs. In *full* natural language cardinality NPs often get a default distributive reading if they occur as subjects of transitive or ditransitive main verbs. For ease of learnability, for simplicity and uniformity of the ACE interpretation principles in ACE cardinality noun phrases *always* get a default collective reading. Thus they are treated on a par with other individual denoting noun phrases. A distributive reading is triggered by using partitive constructions with *each of*:

Note that the interpretation of the cardinality determiner *three* is different from the interpretation of *exactly three*. The use of the determiner *exactly three* excludes that there are more than three objects of a particular kind, whereas *three* does not exclude this option.

Note furthermore, that in ACE *at most n* is treated as an individual denoting noun phrase, which means it carries an existential import that excludes 0. The interpretation thus corresponds to full natural language *some but at most n*. To avoid this existential import users can use the negated determiner *not more than n*.

Absolute vs. Proportional Cardinality NPs. In full natural language determiners like *many* or *few* are ambiguous between an absolute and a proportional reading. Furthermore, the interpretation of these determiners is context-dependent. The absolute cardinality reading of the determiners *many* and *few* expresses that the number of objects that have a certain property is many or few with respect to the size of the domain of objects. In an absolute cardinality reading the sentence

Many automatic tellers are defect.

therefore means that the number of automatic tellers that are defect is many. Which number counts as many is contextually determined.

The determiners *many* and *few* can also have a proportional reading in full natural language. In a proportional reading they express a vaguely specified proportion between two quantities. The proportional reading of

Many automatic tellers are defect.

says that the number of automatic tellers that are defect is a large proportion of the number of all (contextually relevant) automatic tellers. The size of the proportion is again contextually specified.

To resolve the ambiguity between a cardinality and a proportional reading ACE uses the following interpretation principles.

(13) **Absolute Context-Dependent Cardinality Determiners**
In ACE context-dependent NPs starting with the determiners *many* or *few* always get an absolute cardinality reading.
To express a proportional reading users have to choose the partitive constructions *many of the* and *few of the*, resp. Note that indefinite and definite NPs always get an absolute reading.

(14) **Proportional Context-Dependent Cardinality Determiners**
The context-dependent determiners *many of the*, *few of the*, *most*, *most of the* and *more than half of the* get a proportional reading in ACE.

Thus, in ACE the sentence

(15) Many automatic tellers are defect.

means that the absolute number of defect automatic tellers is many. Which cardinality counts as many is not further specified. If users want to make this cardinality explicit they have to use non-context dependent absolute cardinality determiners like *at least n* or *more than n*.

(16) At least 25 automatic tellers are defect.

In ACE it is also possible to express proportional readings. The sentence

(17) Many of the automatic tellers are defect.

expresses that the cardinality of the set of automatic tellers that is defect is many with respect to the cardinality of the set of all (contextually relevant) automatic tellers. Which proportion counts as many is not further specified. If users want to make the proportion explicit they have to use non-context-dependent proportional constructions like partitives:

(18) At least 25 of the automatic tellers are defect.

Note that all partitive constructions express a proportion between two cardinalities.

In general, users are advised to avoid context-dependent determiners. More precise alternatives using numerals or cardinality quantifiers should be used instead.

E.2.3 Coordinated Noun Phrases

Construction

In ACE 4 coordination of noun phrases is possible but only between individual denoting noun phrases.

(19) Valid NP coordinations in ACE

a. a man and two children

b. John and the man

c. several women and a student

d. at least two men and several women

Coordinations containing quantificational noun phrases are *not* allowed.

(20) Not approved in ACE

a. every man and every woman

b. a man and every woman

c. no man and every woman

The examples in (20) have to reformulated, for example by explicitly repeating the verb for each of the noun phrases or by independently introducing the referents of the noun phrases.

Interpretation

A plural noun phrase that is a coordination of individual denoting noun phrases is again an individual denoting noun phrase. It introduces a complex group object into the domain of discourse. As an individual denoting plural noun phrase the coordinated noun phrase can show collective/distributive ambiguities. The ACE 4 rules to resolve these ambiguities correspond to those for individual denoting noun phrases in general. That means the sentence

(21) Two men and several women enter a card.

gets a fully collective reading where a group consisting of two men and several women together enters a card. An optional paraphrase indicates the grouping of the noun phrases as:

(22) {{Two men} and {several women}} enter {a card}.

Note that if non-monotone increasing individual denoting plural NPs (e.g. *at most two men*, *exactly three men*) are coordinated in ACE the monotonicity properties of the NPs get lost. The coordinated NP will be monotone increasing. This is a simplification with respect to full natural language where coordinations of these NPs often get a distributive interpretation that then maintains the monotonicity properties.

Distributive readings of coordinations are expressed by putting *each of* in front of the coordinated NP. More details will be added below.

If an elaboration marker like *together* is added after the coordinated noun phrase the marker refers to the whole coordinated noun phrase. This is expressed by the following ACE 4 rule.

(23) **Elaboration Markers after Coordinated NPs**
Elaboration markers always refer to the whole coordinated noun phrase that precedes the marker. In the paraphrase this rule is indicated to the user through brackets around the coordinated NP that immediately precedes the marker.

For example, in

(24) Two men and several women together enter a card.

the elaboration marker *together* relates to the whole coordinated noun phrase. This is reflected in the paraphrase

(25) {{Two men} and {several women}} together enter {a card}.

Note that markers cannot be added within the coordinated noun phrase.

(26) *Not* approved in ACE:
Two men together and several women enter a card.

Note, furthermore, that a partial distribution where the predicate is distributed to each of the conjuncts is not possible in ACE 4. For example,

(27) Two men and several women enter a card.

cannot mean that two men together enter a card and several women together enter a card. This meaning has to be reformulated by explicitly distributing the verb to the conjuncts, for example:

(28) Two men together enter a card and several women together enter a card.

Furthermore, it is not possible in ACE 4 to coordinate only plural nouns

(29) *Not* allowed in ACE:
The men and women enter a card.

In general, the use of coordinated noun phrases is not recommended in ACE due to the increase of possible ambiguities.

See also the following sections for interactions of NP coordination with other constructions.

Interaction of NP Coordination with Other Constructions

The interpretation of coordination interacts with other elements of the ACE language, for example with relative sentences, negation or other coordinators. In the ACE 4 manual the interpretation principles governing these interactions will be introduced. Here I only summarize the interaction of NP coordination with other ACE constructions.

NP Conjunction and Relative Sentences

If a relative sentence occurs after a noun phrase conjunction ambiguities are possible as to whether the relative pronoun refers to the whole coordinated noun phrase or only to the immediately preceding simple noun phrase. Therefore, in full natural language the sentence

(30) A customer has two Visacards and three Mastercards that are expired.

can mean that the relative pronoun refers to the whole conjoined noun phrase *two Visacards and three Mastercards* or only to the immediately preceding noun phrase *three Mastercards*. ACE employs the following principle to disambiguate these sentences:

(31) **Relative Sentences after Noun Phrase Conjunction**
A relative sentence after a noun phrase conjunction refers to the immediately preceding noun phrase if it agrees in number and object type with the relative pronoun.
If the immediately preceding noun phrase that is part of an NP conjunction does *not* agree in number the relative pronoun refers to the whole conjoined noun phrase that agrees in number.

The sentence

(32) A customer has two Visacards and three Mastercards that are expired.

will be interpreted as

(33) A customer has two Visacards and {three Mastercards that are expired}.

In the example, the relative pronoun cannot refer to the whole coordinated noun phrase. To express this meaning users have to reformulate the sentence for example as

(34) a. A customer has two Visacards and three Mastercards.
b. The Visacards and the Mastercards are expired.

Since in

(35) A customer has a Visacard and a Mastercard that are expired.

the relative sentence *that are expired* requires plural agreement *it* cannot refer to *a Mastercard*. The relative sentence therefore relates to the whole coordinated noun phrase which is shown in the paraphrase as

(36) A customer has {a Visacard and a Mastercard that are expired}.

It is also possible that the noun phrase conjunction occurs *within* a relative sentence.

(37) **Noun Phrase Conjunction within a Relative Sentence**
If a conjoined noun phrase occurs *after* a relative pronoun the whole conjoined noun phrase belongs to the relative sentence.

Therefore in ACE the sentence

(38) John sees a man who enters a card and a code.

is interpreted as

(39) John sees {a man who enters {a card and a code}}.

To express that John sees a man who enters a card and that John sees a code users have to reformulate the sentence for example as

(40) John sees a man who enters a card and sees a code.

or as

(41) John sees a code and a man who enters a card.

The interaction of the principles for VP coordination and NP conjunction predict that the following sentence

(42) Two customers see a man and two women who enter a card and a code and type a password.

is interpreted as

(43) Two customers {see {a man and {two women who enter {a card and a code}}}} and {type a password}.

It is strongly recommended to avoid multiple conjunctions in a sentence. The same situation can be equivalently expressed as.

(44) Two women enter a card and a code.
Two customers see a man and the two women.
The two customers type a password.

In general, the user is advised to make complex situations explicit.

NP Conjunction and 'Of'-PPs

In full natural language the sentence

(45) John sees a customer of Mary and Bill.

is ambiguous. It can mean that the complex phrase *of Mary and Bill* modifies the noun phrase *a customer*:

(46) John sees {a customer of {Mary and Bill}}.

In this case the sentence means that John sees one object, and this object is the common customer of Mary and Bill. The sentence can also mean that *of Mary* modifies *a customer*, and the conjunction *and* conjoins the complex noun phrase *a customer of Mary* and the noun phrase *Bill*.

(47) John sees {{a customer of Mary} and Bill}.

In this case the sentence means that John sees two objects, viz. a customer of Mary and the object Bill.

To resolve this ambiguity ACE uses a principle that states that *of* only relates the immediately preceding noun phrase and the complete following noun phrase.

(48) **Noun Phrase Conjunction and *Of*-PPs**
The special preposition *of* only relates the immediately preceding and the complete (possibly coordinated) following noun phrase. More concretely we distinguish noun phrase conjunction *before* and noun phrase conjunction *after* the preposition *of*:

a. Noun Phrase Conjunction before *of*-PPs
If a coordinated NP occurs before the *of*-PP the preposition *of* only relates the last conjunct with the following noun phrase. That means in ACE the sentence
John sees a clerk and a customer of Mary.
will be interpreted as
John sees a clerk and {a customer of Mary}.
To express that both the clerk and the customer are related to Mary users can write
John sees a clerk of Mary and sees a customer of Mary.
John sees Mary's clerk and her customer.

b. Noun Phrase Conjunction after *of*-PPs
If a coordinated NP occurs after the *of*-PP the preposition *of* relates the preceding NP with the whole coordinated noun phrase. That means in ACE the sentence
John sees a customer of Mary and Bill.
will be interpreted as
John sees {a customer of {Mary and Bill}}.
which means that John sees the common customer of Mary and Bill. To express the interpretation that John sees Bill and that John sees a customer of Mary users can write
John sees Bill and a customer of Mary.
which is paraphrased as
John sees {Bill} and {a customer of Mary}.

Note, that in

(49) John sees a customer of Mary and Bill.

the noun phrase *a customer* has wide scope, that means there is just one customer involved. To express that for each of Mary and Bill there is a customer users can write one of the following sentences:

(50) a. John sees a customer of Mary and sees a customer of Bill.
b. John sees for each of John and Mary his/her customer.
c. John sees Mary's customer and Bill's customer.

Note that the sentence

(51) John sees a customer of Mary and a customer of Bill.

would be interpreted as

(52) John sees a customer of {Mary and {a customer of Bill}}.

which is probably not intended by the user.

Furthermore note that the difference between

(53) John sees a customer of Mary and sees a customer of Bill.

and

(54) John sees Mary's customer and Bill's customer.

is that in the first sentence there are two seeing events, one for each customer. In contrast, in the second sentence there is just one seeing event of (possibly) two customers.

A further example is added to explain the relation between *of* and *coordination*. The above principle predicts that the sentence

(55) John enters a card and a code of a customer.

is paraphrased as

(56) John enters {a card} and {a code of a customer}.

To express a different meaning re-formulations are necessary, for example:

(57) a. John enters {a card of a customer} and enters {a code of the customer}.
b. John enters {a customer's card} and {the customer's code}.

Again, users are advised to make situations as explicit as possible.

Noun Phrase Conjunction and Possessives

Similar to *of*-PPs the combination of possessives with conjunction creates possible ambiguities. In full natural language the sentence

(58) John sees Mary and Bill's card.

can express joint ownership, or a coordination between a simple NP and a possessive NP. To resolve the ambiguities ACE employs the following principle.

(59) Noun Phrase Conjunction and Possessive Nouns

a. If the preceding noun phrase is not an NP coordination the possessive marker belongs to the immediately preceding simple noun phrase (the noun phrase cannot be modified by relative clauses or other modifiers).
b. If the preceding noun phrase is a coordination of simple noun phrases the possessive marker belongs to the whole coordinated NP. The sentence

John sees Mary and Bill's card.

will be interpreted as expressing joint ownership, i.e.

John sees {Mary and Bill}'s card.

To express a different interpretation users can write:

John sees Bill's card and Mary.
which is paraphrased as
John sees {{Bill's card} and Mary}.

Note, that in the sentence

(60) John sees Mary and Bill's card.

there is just one common card that belongs to Mary and Bill. If one wants to express that John and Mary each have their own card one has to write

(61) John sees Mary's card and Bill's card.

Note furthermore that in ACE the following sentence is not possible

(62) Not admissible in ACE
John enters a customer's card and code.

The intended meaning can be expressed in ACE by one of the following sentences:

(63) a. John enters a customer's card and his code.
b. John enters a customer's card and the customer's code.

Again, the user is advised to avoid coordination of complex noun phrases if possible.

NP Conjunction and Appositions

ACE allows for coordinated noun phrases in appositive position. The apposition and the noun phrase have to agree in number.

(64) a. John enters two cards A and B.
b. John sees three customers Bill and Mary and Sue.

The following sentence with the singular NP *a card*

(65) John sees a card A and B.

will, however, be interpreted as

(66) John sees {a card A} and {B}.

since *a card* is singular and the noun phrase *A and B* is plural.

The noun phrase

(67) two customers John and Mary

introduces a group object into the domain of discourse and the apposition enumerates the parts of this group.

NP Conjunction and Adjuncts

Adjuncts (adverbs or prepositional phrases) can occur after a coordinated noun phrase. The sentences

(68) a. John enters a card and a code carefully.
b. John enters a card and a code in the morning.

are paraphrased and interpreted as

(69) a. John {enters {a card and a code} carefully}.
b. John {enters {a card and a code} in the morning}.

That means there is a complex entering event where John enters the card and the code as a whole, and this event is said to be carefully or in the morning, resp.

To express that the adjunct is distributed to both parts of the conjunct you can repeat the verb and the adjunct for each conjunct.

(70) a. John enters a card carefully and enters a code carefully.
b. John enters a card in the morning and enters a code in the morning.

Alternatively, you can say

(71) John enters each of a card and a code carefully.

to express distribution.

Analogously, in full natural language, the sentence

(72) John puts a Visacard and a Mastercard into a purse

is ambiguous between a collective reading where there is just one purse for the Visacard and the Mastercard together, or a distributive reading where there is a purse for the Visacard and a (possibly different) purse for the Mastercard. For the resolution of this type of ambiguity see above.

Interpretation of Coordinated NPs as Arguments of Prepositions

In ACE the sentence

(73) John enters a key with the left hand and the right hand.

is paraphrased as

(74) John {enters a key with {{the left hand} and {the right hand}}}.

and means that there is one entering event, and the key is entered with the left hand and the right hand together. The NP *the left hand and the right hand* introduces a group object into the domain of discourse and this group object is the instrument of entering the key. From the sentence we cannot conclude that John enters a key with the left hand.

Note, in contrast, that the above sentence is *not* equivalent to the sentence

(75) John enters a key with the left hand and enters the key with the right hand.

The sentence expresses that for each of the two hands there is a separate entering event, one with the left hand and one with the right hand, but not necessarily that there is an event with both hands together. From this sentence we can conclude that John enters a key with the left hand.

It is furthermore possible to express a distributive reading by using *each of* constructions:

(76) John enters a key with each of the left hand and the right hand.

though these sentences sound somewhat stilted in full English. Note also, that in this sentences a key has wide scope meaning that there is just one key for the left hand and the right hand. Scope reversal is treated in the full ACE 4 manual.

E.2.4 Partitive Noun Phrases

Construction

Partitives in the strict sense have the following form in ACE:

(77) Standard Partitives in ACE
determiner + *of* + definite plural NP

where the definite plural NP can be a simple definite NP (*the men*) or a coordinated definite NP (*the men and the women*) and where the determiner can be

(78) Determiners

a. *each*
Each of the students passes the exam.

b. indefinite plural determiner (except ø)
Some/two of the students pass the exam.

c. cardinality determiner (singular or plural)
One of the students fails the exam.
At most two of the five students pass the exam.
At least three of the students receive a distinction.

Partitives with *each* and partitives with singular numerals as determiners require singular agreement with the verb.

ACE 4 also allows for "non-standard" partitives that do not use the definite determiner *the*. More concretely, the following constructions are admissible.

(79) Non-standard partitives in ACE
each + of + individual denoting plural NP

An example is:

(80) John reads each of two books.

Note, that in full natural language these "non-standard" constructions are not so frequent but in

ACE 4 they are important to trigger the distributive interpretation of plural noun phrases.

(81) John reads each of two books that Mary likes.

Note, that in ACE 4 partitive constructions can also be used with personal plural pronouns (*each of them*) or with relative pronouns (*each of which*), or even with coordinated NPs (*each of John and Mary*).

Interpretation

Partitives with *each* always get a distributive reading.

The interpretation rules for partitives with indefinite and cardinality determiners correspond to those for individual denoting NPs.

The definite noun phrases within the partitive construction is interpreted like a simple definite noun phrases, i.e. it can be used anaphorically or independently referring.

If the partitive contains coordinated noun phrases, in full natural language ambiguities are possible as to the intended grouping of the NPs. In full natural language the sentence

(82) Two of the men and the women lift a table.

can express a coordination of the NPs *two of the men* + *the women*, or coordination with the structure *two* + *of* + *the men and the women*. In ACE there is a construction rules that assumes the second analysis.

(83) **Grouping of Partitive**
The partitive always relates to the maximal succeeding noun phrase. A maximal noun phrase includes all modifiers of the noun phrases and all elements of a conjunction.
The grouping of the noun phrase is optionally displayed in a paraphrase

The rule predicts that in

(84) Each of the men and the women lifts a table.

distribution is triggered over the whole coordinated NP, and in

(85) Two of the men and the women lift a table.

the partitive picks two objects of the whole group consisting of the men and the women. This interpretation is displayed to the user by adding optional brackets.

(86) Paraphrase of (85)
{Two of {the men and the women}} lift a table.

Users can reformulate the sentence to get a different grouping:

(87) a. The women and two of the men lift a table.
b. {{The women} and {two of the men}} lift a table.

Note that in full natural language individual denoting partitives often prefer a distributive reading. Again, for simplicity, ease of learnability and uniformity in ACE partitive noun phrases with non-quantificational determiners nevertheless get a default collective reading parallel to the non-partitive counterparts.

E.2.5 Measurement Noun Phrases

Construction

ACE 4 allows for a restricted set of measurement constructions, e.g. *two kg of apples*. Measurement noun phrases have the following syntactic structure:

(88) Measurement Noun Phrase:
number + unit + of + mass noun or plural count noun

The constituent *number + unit* is also called the "measure phrase".

Examples for measurement noun phrases are

(89) a. 2 kg of red apples
b. 2 l of water
c. at most 2 ounces of gold

Standard measurement units (*m*, *cm*, *kg*, *seconds*) will be predefined in the ACE lexicon. Note that the abbreviations and the full version of the units, e.g. *s* and *seconds*, have to be marked as aliases in the lexicon. Application specific measurement units can be added by the user. Each measurement unit is associated with a dimension (e.g. size, distance, area, volume, weight, speed, temperature).

Interpretation

Measurement constructions are always interpreted collectively.

Note that users are strongly discouraged from adding non-standard measurement units since ambiguities could occur between measurement constructions and normal *of*-PP modification. For example adding *glass* as a measurement unit would lead to the following ambiguity. In full natural language the noun phrase *two glasses of wine* can be interpreted as a measurement construction, e.g. in the sentence

(90) John drinks two glasses of wine.

But, the noun phrase *two classes of wine* can also be used in a normal *of*-PP modification:

(91) John spills two glasses of wine.

In ACE this ambiguity is resolved by the following rule.

(92) **Normal Nouns and Measurement Nouns**
If a noun occurs both as a normal noun and as a measurement noun in the ACE lexicon then ACE interprets the noun as a measurement noun in all constructions where

this is possible. If an interpretation as a measurement noun is not possible ACE interprets the noun as a normal countable noun. The chosen interpretation is optionally indicated to the user in a paraphrase showing the grouping of the NPs.

For example, if *glass* is defined both as a measurement noun and as a countable noun in the lexicon the sentences

(93) a. John drinks two glasses of wine.
b. John spills two glasses of wine.

will be interpreted and paraphrased as measurement constructions:

(94) a. John drinks two {glasses of wine}.
b. John spills two {glasses of wine}.

If *glass* is only defined as a normal countable noun in the ACE lexicon the sentences will be interpreted as normal *of*-PP modification:

(95) a. John drinks {two glasses} of wine.
b. John spills {two glasses} of wine.

Users are discouraged from defining the same noun both as a countable and as a measurement noun in the lexicon.

E.2.6 Other Possibilities to Express Measurements and Amounts

Measure Phrases in Isolation

The occurrence of measure phrases (*2 kg*) in isolation, i.e. without noun, is restricted to the following constructions. The dimension of measurement (weight, size, volume) has to be made explicit or the measurement copula has to be predefined in the lexicon together with its dimension:

- *the* + dimension + *of* + NP + *be* + measure phrase
  The weight of John is 70 kg.
- NP + *have* + *a* + dimension + *of* + measure phrase
  John has a weight of 70 kg.
- NP + measurement copula + measure phrase
  John weighs 2 ounces.

It is in the responsibility of the user to classify dimension nouns and measurement copulae in the lexicon.

Fractions

Fractions can currently only be expressed via percentages using the following constructions:

- n *percent of the* + plural noun

5 percent of the patients suffer from headaches.
60% of the students pass the exam.

- n *percent of the* + mass noun
  10% of the water is polluted.
  5 percent of the data is corrupt.

Ranges

Ranges of measurement/cardinality can be specified by:

- *between* number1 *and* number2 …
  The weight of John is between 70 and 80 kg.
  John eats between two and five apples.

E.3 Modification of Plural Noun Phrases

E.3.1 Adjectives

Construction

Adjectives can be used to modify nouns, that is to give more information about the object. The admissible positions correspond to the positions described for ACE 3, for example

(96) a. John buys two yellow cars.
b. John buys two expensive cars.
c. John sees two identical cars.

Interpretation

In full natural language there are adjectives that can only be true of individuals (*yellow*), adjectives that can be true of both individuals and group objects (*expensive*) and adjectives that can only be true of groups (*identical*). Currently, in ACE this distinction is not adopted since it requires a subtle distinction of adjectives in the lexicon, and, what is more, the distinction is not clear for all adjectives. ACE uses the following interpretation rules to interpret the adjectives.

(97) **Interpretation of Adjectives Modifying Plural Nouns**
All adjectives modifying a noun get a default distributive interpretation.

The rule reflects that most adjectives in natural language are intrinsically distributive. Therefore in the sentences (96) all adjectives distribute to individual members of the group although this interpretation is not perfectly natural for sentence (96)c. There is no natural way to show the distributive interpretation of the adjective to the user in a paraphrase. The user therefore has to know the interpretation rules.

E.3.2 Relative Clauses

Construction

Relative pronouns like

(98) who, which, that

can relate to plural nouns in the same way as was specified for the singular fragment. More concretely *who* can only relate to single persons or groups of persons, *which* can relate to single non-human objects or groups of such objects, and *that* can relate to either human and non-human objects or mixtures thereof. ACE 4 additionally allows for the "distributive relative pronouns"

(99) each of which
each of whom

The distributive relative pronouns are used to trigger distributive readings over the group object to which the relative pronoun refers. More concretely, *each of whom* is used for a group of persons and *each of which* is used for a group of things or for a mixed group.

Interpretation

In full natural language, sentences with relative pronouns relating to individual denoting plural noun phrases are in the same way ambiguous as the noun phrase itself. For example, the relative pronoun *who* in the relative clause of

(100) Five climbers <u>who carry two tents</u> use an oxygen bottle.

can be interpreted distributively or collectively with respect to *carry* since it relates to the individual denoting plural noun phrase *five climbers*. ACE resolves this ambiguity in the following way.

(101) **Collective/Distributive Interpretation of Relative Pronouns**
If the relative pronoun relates to an individual denoting plural noun phrase it gets a default collective reading. To override this default interpretation constructions with *each of which/each of whom* have to be used.

For example in the sentence (100) both *five climbers* and *who* get a default collective reading. To override the default reading users can reformulate the sentence as

(102) Five climbers each of which carry two tents use an oxygen bottle.

In this example *five climbers* is interpreted collectively with respect to *use an oxygen bottle*, and – via the distributive relative pronoun – distributively with respect to *carry two tents*.

It is also possible that noun phrases functioning for example as a direct or indirect objects within the relative clause are modified. In this case we cannot add a the *each of* construction directly before the noun phrase since the noun phrase is extracted from its original position which is now empty. In the example

(103) Five programs which two students use require a password.

the noun phrase *five programs* is extracted from its object position that follows after the verb *use*. In ACE 4 a relative pronoun that relates to an extracted individual denoting plural noun phrase gets a default collective reading. To trigger a distributive reading of *five programs* with respect to *use* one has to use the “distributive” relative pronoun *each of which*:

(104) Five programs each of which two students use require a password.

Note however, that in this case *each of which* has wide scope over *two students* meaning for each of the five programs there can be two different students. Techniques for scope reversal are discussed in the full ACE 4 manual.

If one additionally wants to express that *five programs* is interpreted distributively with respect to the verb phrase *require a password* one can use the following reformulation:

(105) Each of five programs each of which two students use require a password.

Note that the grouping of the noun phrases follows the rule that *each of* relates to the maximal succeeding noun phrase yielding the following optional paraphrase:

(106) {Each of {five programs each of which two students use}} require a password.

Note, that the examples show that is not recommended to use many plural noun phrases in one sentence, in particular nesting within relative clauses should be avoided. We recommend users to split sentences into several parts. For example, depending on the intended scopings, users can for example formulate the following sentences:

(107) a. Two students use each of five programs. Each of the five programs requires a password.
b. There are five programs each of which two students use. Each of the five programs requires a password.

E.3.3 *Of*-PPs and Plurals

Construction

ACE allows for plural NPs in *of*-PP constructions. The admissible positions correspond to the positions of singular NPs.

Interpretation

In full natural language constructions with plural NPs and *of*-PPs show basically the same range of collective/distributive ambiguities and scope ambiguities as noun phrases that are arguments of verbs. For example in

(108) the weight of two automatic tellers

the noun phrase *two automatic tellers* can have a narrow scope collective interpretation with respect to *the weight* in which case the above complex noun phrase refers to the weight of the two automatic tellers together. The complex noun phrase can also be interpreted with a wide scope distributive interpretation of *two automatic tellers* in which case for each of the two automatic tellers its weight is considered. In some constructions it is also possible that the first NP of an *of*-PP construction gets a wide scope distributive reading, for example in

(109) four books of 450 pages

which can mean that the books together have 450 pages or that each book has 450 pages. Note however, that this is a different use of the preposition *of* than the one intended in ACE.

ACE 4 deals with these ambiguities using the same principles as for other collective/distributive and scope ambiguities. That means, surface order determines scope and individual denoting noun phrases get a default collective reading. Other readings have to be expressed by reformulating the sentence. The noun phrase

(110) the weight of two automatic tellers

therefore expresses the collective weight of two automatic tellers. To express the distributive wide scope interpretation of *two automatic tellers* the sentence can for example be expressed using a construction with *for each of*:

(111) a. for each of two automatic tellers the weight of the automatic teller
b. for each of two automatic tellers its weight

The default interpretation for

(112) four books of 450 pages

expresses that the four books together have altogether 450 pages. The wide scope distributive interpretation of *four books* has to be expressed by a reformulation in ACE, for example

(113) four books each of which consists of 450 pages

Note that the interaction *of*-PPs with coordinations is described in E.2.3.

E.3.4 Possessive Nouns and Plurals

Construction

It is possible to use plural nouns in possessive constructions. Possessives have the following syntactic structure

(114) Possessive NP:
embedded NP + *s* + noun

Both the embedded NP and the noun can be in the plural. The embedded noun phrase can be simple or coordinated. The noun can be modified by an adjective.

More concretely, the embedded NP can be realized as

- an individual denoting simple singular NP
  the bank's automatic tellers
  a man's credit card
- an individual denoting simple plural NP
  the customers' invalid cards
  two customers' invalid cards
- a coordinated plural NP
  the man and the woman's cards
- a quantifying NP
  every customers's valid cards
  no man's card

As a result there are several types of possessive NPs. If the noun is in the singular and the embedded NP is individual denoting then the possessive NP is an

- individual denoting singular possessive NP
  John's book, *a man's card*, *the men's house*, *John and Mary's book*

If the noun is in the plural and the embedded NP is individual denoting the possessive NP is an

- individual denoting plural possessive NP
  John's books, a man's cards, the men's houses, John and Mary's books

If the noun is in the singular or the plural and the embedded NP is quantifying the possessive NP is a

- quantifying possessive NP
  every man's book, every man's books, no men's books

Note that in ACE it is not possible to use numerals together with the noun, e.g. *the bank's five automatic tellers* is not admissible in ACE. Also it is not possible to only coordinate the modified nouns as in *the customer's cards and codes*. The treatment of possessive *s* with coordinated NPs is explained in section E.2.3. It is also not possible to use constructions like *a customer's card of Mary* whereas constructions like *a card of Mary's customer* are admissible.

Interpretation

Since in possessive constructions two nouns are put into relation to each other scope ambiguities and collective/distributive ambiguities can occur in full natural language. The ambiguities can occur within the possessive noun phrase and ambiguities can occur if the whole possessive noun phrase is combined with other noun phrases in a sentence. In full natural language the noun phrase

(115) the customers' accounts

can (i) denote for each of the customers his or her accounts, in which case *the customers* is interpreted distributively with respect to *accounts*. The noun phrase can also denote (ii) the whole set of accounts that the customers have, in which case *the customers* is interpreted collectively with respect to *accounts*. As a consequence, the sentence

(116) The clerks discuss the customers' accounts.

can mean that (i) the clerks check for each of the customers his or her accounts. The sentence can also mean that (ii) there are some customers that have some accounts and that the clerks check the all these accounts. What is more, the whole possessive noun phrase *the customers' accounts* is an individual denoting plural noun phrase and as such can be interpreted distributively or collectively with respect to the verb *discuss*. Assuming a distributive interpretation of the embedded NP *the customers* and a distributive interpretation of the whole possessive noun phrase the sentence means that each of the customers' individual accounts is discussed in a separate event. Assuming a collective interpretation of the whole possessive NP and a distributive interpretation of the embedded NP the sentence means that for each of the customers there is a set of account that is discussed in one collective event. Other combinations of collective/ distributive readings follow accordingly.

It is not always evident what exactly the difference between the possible readings is. Still, the different readings allow for different logical inferences which is why in ACE the ambiguities have to be resolved. Again ACE chooses its standard algorithm to resolve the ambiguities. First, surface order of noun phrases determines scope. Second, individual denoting plural NPs get a default collective reading. That means in

(117) the customers' accounts

the noun phrase *the customers* is interpreted collectively with respect to *accounts*. This reading is not displayed to the user in the paraphrase since there is no natural way to do so. To express a distributive reading of *the customers* with respect to *accounts* users have to reformulate the phrase e.g. as

(118) for each of the customers his accounts

The possessive NP *the customers's account* is individual denoting and thus also gets a default collective reading with respect to the verb. In the sentence

(119) The clerks discuss the customers' accounts.

all NPs are interpreted collectively. A distributive reading of t*he customers' accounts* can be triggered e.g. by

(120) The clerks discuss each of the customers' accounts.

Note that the interpretation of the above sentence is different from

(121) For each of the customers the clerks discuss his accounts.

Again, we recommend the users to avoid complex NPs in a sentence.

E.3.5 Appositions and Plurals

Construction

ACE allows for plural noun phrases in appositive positions. More concretely the first NP has to be a simple individual denoting plural NP and the second NP has to be a coordination of proper nouns, or a coordination of dynamic names:

(122) a. two customers John and Mary
b. the customers C1 and C2 and C3

Interpretation

In an appositive construction with two individual denoting plural NPs the two noun phrases refer to the same group, the apposition explicitly lists the parts of the group introduced by the first NP. Within the apposition there are no collective or distributive ambiguities. However, the whole noun phrase can get a distributive or a collective reading according to the rules of ACE.

(123) The customers John and Bill and Mary enter a card.

gets a default collective reading. The distributive reading has to be expressed by

(124) Each of the customers John and Bill and Mary enters a card.

Note that the whole coordination belongs to the appositive position. To express a different grouping the sentence has to be reformulated, e.g. as

(125) Mary and the customers John and Bill enter a card.

E.4 Elaboration Markers

General Remarks

In general, a collective reading means that a situation is viewed as expressing one possibly complex event. However, the collective reading is often indeterminate as to how the individuals making up a group participate in a certain relation or event. If we have a collective reading for the object noun phrase *three chairs* in

(126) A man carries three chairs upstairs.

it is not clear how the chairs are actually carried upstairs. The sentence is true as long as – finally – three chairs are carried upstairs. There could be a complex event of lifting first one chair alone, then the other two chairs together, or an event where the three chairs are lifted as a whole. The sentence is indeterminate in this respect, it does not spell out the possible constellations, perhaps because this precision is not necessary for what the user wants to express. In natural language it is not always easy to verbalize the intended constellations. Sometimes tables or diagrams are used for this purpose. The ACE language does not include these graphical elements. Nevertheless there are several means to elaborate on possible constellations. Users can split the complex event into several sub-events that describe the situation more precisely. It is also possible that the user introduces elaboration markers to reduce the indeterminacy.

ACE assumes an indeterminate collective reading for the above sentence and offers the users additional predefined elaboration markers that can be used to reduce the indeterminacy of the collective reading if desired. These markers can elaborate on the collective reading along different dimensions:

- temporal dimension (*simultaneously*)
- spatial dimension (*at the same place*)
- group-structure dimension (*collectively*, *as a whole*)

Note, that elaboration markers not only elaborate on the collective reading but – indirectly – also indicate that a noun phrase is read collectively.

Construction

Elaboration markers are always put directly after the noun phrase that is elaborated on. Corresponding to the dimensions of elaboration ACE distinguishes three types of predefined elaboration markers

- temporal elaboration markers
  at the same time, *simultaneously*, *one by one*
- spatial elaboration markers
  at the same place
- group-structure elaboration markers
  as a whole, as a group, collectively, jointly

In contrast to disambiguation markers, elaboration markers cannot be used within *of*-constructions:

(127) Not allowed in ACE
two customers as a whole of a Swiss bank

The example also shows that elaboration markers can only be used to elaborate on the interpretation of NPs with respect to verbal predicates, therefore elaboration markers are similar to adverbials.

Interpretation

As stated above elaboration markers detail the indeterminacy of the collective reading. More concretely, they add information about the internal structure of the event with respect to a group participant. That means, elaboration markers relate to both a noun phrase and an eventuality. The is controlled by the following rule.

(128) **Elaboration Markers**

a. The elaboration marker always refers to the immediately preceding maximal noun phrase. A maximal noun phrase includes all modifiers of the noun phrases and all elements of a conjunction.

b. If two or more markers follow each other the markers are resolved from the outside to the inside.

c. Elaboration markers relate the preceding maximal noun phrase to the predicate of which the maximal noun phrase is a complement or an adjunct.

In the example

(129) Five climbers who carry two tents together use an oxygen bottle.

the elaboration marker *together* relates to the complex noun phrase

(130) five climbers who carry two tents

In the paraphrase this rule is expressed by adding (optional) brackets around noun phrases

(131) {Five climbers who carry {two tents}} together use {an oxygen bottle}.

The elaboration marker relates to the noun phrase that is enclosed in the outer brackets. In principle it is possible that the same elaboration markers follow each other in complex sentences. Although the following sentence is not recommended

(132) Three men which enter two cards simultaneously simultaneously lift a table.

it will be accepted by ACE 4 and paraphrased as

(133) {{Three men} which enter {two cards} simultaneously} simultaneously lift a table.

The example shows that markers are resolved from outside to inside.

Note, that elaboration markers are not always sufficient to detail the intended constellation. In this case ACE users have to reformulate the sentence to express more precisely what they mean. For example, they can use the sentence

(134) A man carries three chairs upstairs.
He carries two of the chairs upstairs and then he carries one of the chairs upstairs.

Note however, that the logical form for (134) does not guarantee that the second chair is different from the first two chairs. The exact meaning of the elaboration markers that is needed for example for temporal reasoning has to be spelled out in the inference component. This has not yet been fully completed in ACE 4.

Note furthermore, that elaboration markers cannot be put after distributively interpreted NPs.

E.5 Copula and Plurals

Construction

Plural noun phrases can occur as arguments of the copula *be*. In ACE 4 there are the following main constructions with plural noun phrases and the copula *be*:

- plural NP + copula + intransitive adjective
  Two cards are valid.
  Five cards are expensive.
- plural NP + copula + transitive adjective or comparative
  Many customers are older than the clerks.
  Some customers are interested in Visacards.
- plural NP + copula + bare plural
  Some customers are children.
- plural NP + copula + PP
  Some customers are in a bank.
- definite plural NP + copula + definite plural NP
  The customers are the children.
  John and Mary are the customers.
  The customers are John and Mary.

Note that constructions with *there are* are treated in E.6.

Interpretation

Copula plus adjective or PP complement behaves analogously to normal VPs. Copula plus bare plurals is a special case.

Plural NP + copula + intransitive adjective

In ACE 4 individual denoting plural noun phrases in subject position of a copula with adjective complement get a default collective interpretation independent of the meaning of the adjective. The distributive reading has to be triggered by *each of*. In ACE 4 the sentence

(135) Two men are hungry.

means that there are two men and that in a complex state these two men as a group are hungry. That means the adjective is not automatically distributed to every member of the group. To get a distributive interpretation the sentence

(136) Each of two men is hungry.

has to be used.

Note, that in full natural language adjectives used with the copula are often distributed to the individual members of a group. Also adjectives used as noun modifiers in ACE get a distributive reading, but the verb phrase generated from the combination copula + adjective is treated as a “normal” verb phrase and as such gets a collective reading. Therefore in ACE from the sentence *Two men are hungry.* it is not possible to deduce *There are two hungry men*. If it turns out that this deduction is necessary additional first-order axioms for “distributive” adjectives could be introduced within the inference component.

Plural NP + copula + transitive adjective

The readings of noun phrases as arguments of transitive adjectives and comparatives are analogous to those of transitive full verbs. That means in the sentence

(137) Two customers are interested in a credit card.

the subject noun phrase *two customers* gets a default collective reading. To express the distributive reading users have to write

(138) Each of two customers is interested in a credit card.

Plural NP + copula + PP

If the complement is a prepositional phrase the default interpretation of individual denoting plural subjects is collective. The sentence

(139) Two men are in a car.

means that there are two men and one car and the two men are together in that car. To get a distributive reading with possibly different cars per man users can write:

(140) Each of two men is in a car.

Plural NP + copula + bare plural

The sentence

(141) Five climbers are women.

means that there is a complex state where there is a group consisting of five climbers each of which is a woman. The bare plural *woman* is only needed for agreement reasons, it is therefore treated like other plural nouns in ACE in that the property of being a woman is distributed to the single members of the group. However, the state itself is still "collective". To get a "distributive" state users have to write

(142) Each of five climbers is a woman.

Again, additional axioms have to be formulated to model respective collective-distributive inferences if necessary.

Note that ACE 4 does currently not allow to express that the sentence

(143) Five climbers are friends.

expresses a reciprocal relation, that means that the five climbers are friends of each other. In ACE the sentence simply states that each of the five climbers is a friend. ACE also has no account of collective nouns in constructions like *Five climbers are a team.*

Definite plural NP + copula + definite plural NP

If both arguments of the copula are definite plural NPs the two NPs refer to the same group object. Both noun phrases are interpreted collectively.

E.6 *'There'*-Constructions

Construction

In ACE 4 it is possible to use *there are* with non-definite individual denoting plural noun phrases or with negated noun phrases. The result functions as a sentence. The constructions have the following form:

(144) *'There'*-constructions
there + copula + non-definite individual denoting noun phrase

Simple plural examples are

(145) a. There are a man and a woman.
b. There are two customers.
c. There are exactly three cards.

The noun phrase can be further modified, e.g.

(146) a. There are two men John and Mary.
b. There are two cards of John.
c. There are three cars each of which John buys.

Note that *there*-constructions can also be used with NPs starting with *no*:

(147) There are no men.

which is equivalent to

(148) It is not the case that there are men.

The combination with definite noun phrases is not possible.

There-constructions can be used as parts of complex sentences.

(149) a. There is a card and there is a code.
b. If there is a card that every automatic teller rejects then the card is invalid.

Interpretation

The construction *there is*/*there are* does not introduce a state. It simply introduces new objects into the domain of discourse or negates their existence. There are no collective/distributive ambiguities. Only if the construction is modified by relative clauses eventualities are introduced and collective/distributive ambiguities possible.

Note that *there*-constructions can be used to extend the scope of a noun phrase. In

(150) Every man sees a customer.

the noun phrase *a customer* has narrow scope with respect to *every customer*. To change the scope you can instead write

(151) There is a customer such that every man sees the customer.

where *a customer* has wide scope with respect to *every man*. Note that ACE requires to add *such that* after the *there*-construction if it is to be used as a global quantifier. For more information see the full ACE 4 manual.

E.7 Plural Pronouns as Anaphors

E.7.1 Personal Plural Pronouns

Construction

Personal plural pronouns (*they*, *them*) function as whole noun phrases. They are used as substitutes or replacements for previously mentioned individual denoting plural noun phrases. In the sentence

(152) A customer enters two cards. They are valid.

the personal pronoun *they* is an anaphor that substitutes the noun phrase *two cards*. *They* and *two cards* refer to the same object.

Interpretation

Reference Resolution

Like a singular pronoun, a personal plural pronoun can only refer to an explicitly mentioned noun phrase that has the same number and that *is not* the subject, an object or an adjunct of the same verb. In the sentence

(153) A customer has three credit cards. Five automatic tellers accept them.

the personal pronoun *them* cannot refer to *five automatic tellers* because this noun phrase is the subject of the sentence. Neither can it refer to *a customer* because this noun phrase has a different number. The only noun phrase it can refer to is *three credit cards*. The sentence will be paraphrased as

(154) A customer has three credit cards. Five automatic tellers accept [the three credit cards].

If the preceding accessible noun phrase is a coordinated plural noun phrase the pronoun refers to the whole coordinated noun phrase that functions as an argument of a verb or a preposition. In the example

(155) A customer has a Visacard and a Mastercard. SimpleMat accepts them.

the pronoun *them* refers to the noun phrase *a Visacard and a Mastercard*. The sentence will be paraphrase as

(156) The automatic teller accepts [the Visacard and the Mastercard].

This principle is also valid if the coordination consists of two plural noun phrases as in

(157) A customer has one Visacard and two Mastercards.
SimpleMat accepts them.

The second sentence is paraphrased as:

(158) SimpleMat accepts [the Visacard and the two Mastercards].

Note that plural anaphors to discontinuous noun phrases are *not* possible in ACE.

(159) Not admissible in ACE:
A customer meets a clerk. They are in a bank.
Use instead:
A customer meets a clerk. The customer and the clerk are in a bank.

Collective/Distributive Ambiguities

In full natural language plural personal pronouns can exhibit collective/distributive ambiguities. The sentence

(160) A customer sees two cars. They have a rich owner.

can mean that each of the cars has a rich owner or that the cars together have a rich owner. ACE first determines to which noun phrase the pronoun relates to then ACE assigns a default collective/distributive reading to the pronoun. This default reading is independent of the reading of the antecedent. For interpreting the collective/distributive reading of a personal plural pronoun ACE uses the same principles as for other individual denoting plural NPs. The default reading can be overridden by *each of*-constructions.

In the example above the personal pronoun gets a default collective reading. To override the default reading the users can write for example the following sentences

(161) a. A customer sees two cars. Each of them has a rich owner.
b. A customer sees two cars each of which have a rich owner.

Note that in ACE the singular pronoun *it* in the sentence

(162) Three men each insert a card. It is invalid.

cannot be resolved to *a card* since *a card* is in the scope of a distributively interpreted (i.e. universally quantified) plural noun phrase.

E.7.2 Reflexive Plural Pronouns

Construction

Reflexive plural pronouns (*themselves*) function as substitutes or replacements for previously mentioned individual denoting plural noun phrases. In the sentence

(163) Five robots destroy themselves.

the reflexive pronoun *themselves* is an anaphor that substitutes the noun phrase *two robots*.

Interpretation

Reference Resolution

Like singular reflexive pronouns, in ACE a plural reflexive pronoun can only relate to a preceding plural noun phrase that agrees in number and that *is* the subject, an object or an adjunct of the same verb. In

(164) The company has five computers. The robots destroy themselves.

the reflexive pronoun *themselves* can only refer to *the robots*, not to *five computers*.

Collective/Distributive Ambiguities

In full natural language the collective/distributive distinction for reflexive plural pronouns is less clear than for plural personal pronouns. The sentence

(165) Five robots destroy themselves.

most likely means that each of the five robots destroys itself, but collective readings are also possible as in

(166) Five robots place a huge block behind themselves.

The sentence can mean that there is one huge block and the robots act together and put this block behind themselves as a group. Collective readings are also possible in a sentence like

(167) The five companies call themselves ‘EFE’.

which most likely means that the group consisting of the five companies together has the name *‘EFE’*.

To avoid these ambiguities ACE again assigns a default interpretation that is determined by the same principles as the default interpretation of personal pronouns. ACE requires additional syntactic cues to override this default interpretation. This cues do not always sound perfectly felicitous in full English but they help to make the texts more explicit.

The reflexive pronoun *themselves* in

(168) Five robots destroy themselves.

gets a default collective reading. The sentence means that a group of five robots destroys the whole group of robots.

In ACE, the following sentence with a distributive marker

(169) Each of five robots destroys themselves.

will get the (unlikely) reading that each of the robots destroys the whole group of robots. In contrast

(170) Five robots destroy each of themselves.

means that the group of robots destroys each single part of the group.

If you want to express in ACE that each of the robots destroys only itself you have to use a singular construction such as

(171) Each of five robots destroys itself.

Note, that if you use *each of*-constructions in subject position grammatical congruency not only requires a singular verb but also a singular reflexive pronoun if you want to relate to the individual members of the group.

In general, it is recommended that users make the interpretation of the sentence as explicit as possible ideally by using singular constructions.

The above remarks showed that in ACE it is possible that the antecedent and the reflexive pronoun are interpreted differently. In full natural language intuitions vary as to how acceptable these readings are.

(172) a. The robots together place a cube behind each of themselves.
b. Each of the robots places a cube behind themselves as a whole.

ACE generally recommends to avoid these constructions, allows them however for systematic reasons.

Note also that in ACE the reflexive pronoun can be replaced by an anaphoric definite description:

(173) Each of the robots places a cube behind <u>the robots</u> as a whole.

Note finally that in ACE the reciprocal pronoun *each other* is not allowed. It has to be replaced by definite descriptions or explicit descriptions of a certain situation.

E.7.3 Possessive Plural Pronouns

Construction

Plural possessive pronouns (*their*, *their own*) can occur in the same positions as singular possessive pronouns.

Interpretation

Reference Resolution

Like singular possessive pronouns plural possessive pronouns can be used together with the marker *own* to indicate that the possessive pronoun behaves analogue to reflexive pronouns in ACE.

(174) **Plural Possessive Pronouns with the Marker *own***
A plural possessive pronoun that carries the disambiguation marker *own* (*their own*) always refers to the most recent accessible plural noun phrase that has the same number and that is the subject, an object or an adjunct of the same verb.

In the sentence

(175) A customer enters two cards. The automated tellers read their own code.

the disambiguation marker *own* makes the possessive pronoun *their* unambiguously refer to *the automated tellers*.

(176) **Plural Possessive Pronouns without Marker**
If no marker is provided, the possessive pronoun behaves analogue to personal pronouns in ACE. That means, a possessive plural pronoun (*their*) without disambiguation marker always refers to the most recent accessible noun phrase that has the same number that is *not* the subject, an object or an adjunct of the same verb.

In ACE the possessive pronoun *their* in

(177) A customer enters two cards. The automated tellers read their code.

therefore unambiguously refers to *two cards*.

Collective/Distributive Ambiguities

Like with other possessive constructions it is possible that collective/distributive ambiguities occur within the possessive noun phrase.

For example, in full natural language the sentence

(178) The students discuss their own papers.

can mean that the students discuss the papers that they wrote together (collective interpretation of *their own* with respect to *papers*), or the sentence can mean that the students discuss for each of the students his or her own papers (distributive interpretation of *their own* with respect to *papers*).

In ACE the plural possessive pronoun always gets a default collective reading. The distributive reading has to be expressed e.g. by using *for each of* and a singular possessive pronoun.

(179) a. The students discuss for each of the students his/her own papers.

b. The students discuss for each of themselves his/her own papers.

Although theoretically possible it is difficult to additionally express a distributive interpretation of the possessive NP with respect to the verb. Still, the following construction can be used for this purpose:

(180) The students discuss for each of themselves each of his/her own papers.

Other rules concerning possessive NPs are discussed in the singular part of the ACE 4 manual.

E.8 Plural Definite Descriptions as Anaphors

Construction

Definite plural noun phrases can also be used as anaphors.

(181) A customer shows a defect card to two employees. The employees check it.

The definite noun phrase *the employees* refers back to the indefinite noun phrase *two employees*. Both noun phrases denote the same object.

Principle — Definite Noun Phrases as Anaphors

The interpretation of plural definite noun phrases used as anaphors is guided by the following interpretation principle:

(182) **Definite Plural Noun Phrases as Anaphors**
Definite plural noun phrases used anaphorically always refer to the most recent accessible noun phrase that is suitable.
A suitable noun phrase is a noun phrase that has the same plural noun and at least the same numeral, adjectives, possessive noun, *of*-prepositional phrase, and apposition as the anaphoric definite noun phrase.
The presence of relative sentences in the antecedent does currently not play a role in the determination of suitable antecedents.
If no reference can be found, a definite plural noun phrase introduces a new object that is maximal with respect to the restrictor of the definite article.

Each of the following definite noun phrases

(183) a. the three valid cards
b. the two cards
c. the cards of the customers
d. the cards of the Swiss customers

could refer for example refer to the noun phrase

(184) the two valid cards of the Swiss customers

In contrast, the following noun phrases cannot refer back to the noun phrases in (184) because they do not share the same restrictions:

(185) a. the invalid cards
b. the four cards
c. the cards of the German customers

If the antecedent is an NP-conjunction and both conjuncts are suitable then the definite noun phrase refers only to the closest preceding conjunct. In the discourse

(186) The Swiss customers and the German customers have a valid card.
Each of the customers enters a code.

the noun phrase *the customers* only refers to *the German customers*. To refer back to the whole conjunction the conjuncts have to be repeated:

(187) Each of the Swiss customers and the German customers has a valid card.

Also it is not possible to have a plural definite description refer back to a conjunction of singular noun phrases. In the discourse

(188) A Swiss customer and a German customer have a valid card.
The customers have a valid card.

an anaphoric reference from *the customers* to the coordinated NP *a Swiss customer and a German customer* is not possible in ACE.

The correct reference can be achieved by a personal plural pronoun or by a conjunction of definite NPs:

(189) Admissible anaphoric references to *a Swiss customer and a German customer* in (188):
a. The Swiss customer and the German customer have a valid card.
b. They have a valid card.

Notes

Definite noun phrases can be used to establish anaphoric references that are impossible with personal pronouns.

(190) John enters two cards into three machine. They are damaged.

According to the above principle the Attempto system will generate the paraphrase

(191) John enters two cards into three machine. [The three machines] are damaged.

while the user wanted to express that the cards are damaged. Anaphoric reference with the help of a definite noun phrase

(192) John enters two cards into three machine. The cards are damaged.

solves the problem.

E.9 Questions in ACE

ACE 4 allows to use plural nouns in *wh*-questions:

(193) Which customers enter a card?

The question asks for both singular and plural objects that either enter a card together or enter a card individually. To only ask for individual card owners users have to write:

(194) Each of which customers enters a card?

For further comments about questions in ACE users should refer to the complete ACE 4 manual.

E.10 Style Recommendations

Though the ACE parser can readily process complex plural situations that are expressed as syntactically correct ACE sentences, you may not be able to fully understand the meaning of these sentences. For your support we formulated the following rules of style.

- Avoid plurals where possible, use singular instead.
- Carefully use disambiguation and elaboration markers to express the intended meaning.
- Try to avoid putting several plural noun phrases into one sentence.
- Try to avoid relative clauses, or other NP modifiers, use separate sentences instead. Instead of
  Two men who together repair a computer each receive a present.
  write
  Two men together repair a computer. They each receive a present.
- Make distributions explicit. Instead of
  Each of John and Mary lifts a computer.
  write
  John lifts a computer and Mary lifts a computer.
- Try to avoid coordination of (plural) noun phrases.

References

Akmajan, A. and Lehrer, A. (1976). NP-like quantifiers and the problem of determining the head of an NP. *Linguistic Analysis, 2*(4).

Allen, J. (1995). *Natural Language Understanding* (2nd ed.). Redwood City, CA: Benjamin/ Cummings.

Allgayer, J. and Redding-Siekmann, C. (1990). What KL-ONE lookalikes need to cope with natural language. In K. H. Bläsius, U. Hedtstück and C. R. Rollinger (eds.), *Sorts and Types in Artificial Intelligence* (Lecture Notes in Artificial Intelligence, Vol. 418, pp. 240–285). Berlin: Springer.

Alshawi, H. (ed.) (1992). *The Core Language Engine*. Cambridge, Mass.: MIT Press.

Aone, C. (1991). *Treatment of Plurals and Collective Distributive Ambiguity in Natural Language Understanding* (Ph.D. Thesis and MCT Technical Repot ACT-NL-155-91). Austin: University of Texas.

Bach, E. (1986). The algebra of events. *Linguistics and Philosophy, 9*, 5–16.

Balzer, R. M. (1985). A 15 Year Perspective on Automatic Programming. *IEEE Transactions Software Engineering, 11*(11).

Barwise, J. and Cooper, R. (1981). Generalized Quantifiers and Natural Language. *Linguistics and Philosophy, 4*, 159–219.

Baumgartner, P., Fröhlich, P., Furbach, U. and Nejdl, W. (1997). Tableaux for Diagnosis Applications. In *6th Workshop on Theorem Proving with Tableaux and Related Methods* (LNAI, pp. 76–90): Springer-Verlag.

Beckert, B. and Posegga, J. (1995). leanTAP: Lean, Tableau-based Deduction. *Journal of Automated Reasoning, 15*(3), 339–358.

Blackburn, P. and Bos, J. (2000a). *Representation and Inference for Natural Language: A First Course in Computational Semantics.* (Draft available at http://www.comsem.org).

Blackburn, P. and Bos, J. (2000b). *Working with Discourse Representation Theory: An Advanced Course in Computational Semantics.* (Draft available at http://www.comsem.org).

Blackburn, P. and Bos, J. (2003). Computational Semantics. *Theoria, 18*(1), 27–45.

Blackburn, P., Bos, J., Kohlhase, M. and de Nivelle, H. (1999). *Inference and Computational Semantics.* Proc. Third International Workshop on Computational Semantics (IWCS-3), Tilburg, The Netherlands.

Blackburn, P., Bos, J., Kohlhase, M. and de Nivelle, H. (2001). Inference and Computational Semantics. In H. Bunt, R. Muskens and E. Thijsse (eds.), *Computing Meaning* (Vol. 2, pp. 11–28): Kluwer Academic Publishers.

Bläsius, K. H., Hedtstück, U. and Rollinger, C. R. (1990). *Sorts and Types in Artificial Intelligence* (Lecture Notes in Artificial Intelligence, Vol. 418). Berlin.

Bonnema, R., Bod, R. and Scha, R. (1997). *A DOP Model for Semantic Interpretation.* Proc. 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics, Madrid, Spain, July 7–12.

Bos, J. (2001). DORIS 2001: Underspecification, Resolution and Inference for Discourse Representation Structures. In P. Blackburn and M. Kohlhase (eds.), *ICoS-3, Inference in Computational Semantics* (pp. 117–124).

Bos, J. and Gabsdil, M. (2000). First-Order Inference and the Interpretation of Questions and Answers. In M. Poesio and D. Traum (eds.), *Goetalog 2000. Fourth Workshop on the Semantics and Pragmatics of Dialogue* (Gothenburg Papers in Computational Linguistics, Vol. 00-5, pp. 43–50).

Bos, J., Gambäck, B., Lieske, C., Mori, Y., Pinkal, M. and Worm, K. (1996). *Compositional Semantics in Verbmobil.* Proc. 16th International Conference on Computational Linguistics, Copenhagen, Denmark.

Bowen, J. P. and Hinchey, M. G. (eds.) (1999). *High-Integrity System Specification and Design*. London: Springer.

Brusconi, V., Console, L. and Terenziani, P. (1997). Later: Managing Temporal Information Efficiently. *IEEE Expert: Intelligent Systems and Their Applications, 12*(4), 56–64.

Bry, F., Eisinger, N., Schütz, H. and Torge, S. (1998). *SIC: Satisfiability Checking for Integrity Constraints.* Proc. 6th International Workshop on Deductive Databases and Logic Programming (DDLP'98), Workshop at the Joint International Conference and Symposium on Logic Programming (JICSLP'98), Manchester, UK, June 20.

Bry, F. and Torge, S. (1998). A Deduction Method Complete for Refutation and Finite Satisfiability. In *Proc. 6th European Workshop on Logics in Artificial Intelligence* (Lecture Notes in Artificial Intelligence, Vol. 1489): Springer.

Bry, F. and Torge, S. (1999). *Solving Database Satisfiability Problems (Extended Abstract).* Proc. 11. Workshop Grundlagen von Datenbanken, Luisenthal (Thüringen), Germany, May 25–28.

Bry, F. and Yahya, A. (1996). *Minimal Model Generation with Positive Unit Hyper-Resolution Tableaux* (Research Report PMS-FB-1996-1): Institut für Informatik, Ludwig-Maximilians Universität München.

Bünzli, A. (to appear). *ACELex.* Master Thesis, University of Zurich, Zurich.

Carlson, G. N. (1987). *Same* and *Different*: Some Consequences for Syntax and Semantics. *Linguistics and Philosophy, 10*, 531–565.

Carpenter, B. (1995). *Distribution, Collection and Quantification: A Type-Logical Account of Plurality.* Proc. First International Conference on Formal Grammar, Barcelona.

Carpenter, B. (1998). *Type-Logical Semantics*. Cambridge, Mass.: MIT Press.

CLAW. (1996). Proc. 1st International Workshop on Controlled Language Applications (CLAW), University of Leuven, Belgium, March 26–27.

CLAW. (1998). Proc. 2nd International Workshop on Controlled Language Applications (CLAW), Pittsburgh, Pennsylvania, May 21–22.

CLAW. (2000). Proc. 3rd International Workshop on Controlled Language Applications (CLAW), Seattle, Washington, April 29–30.

CLAW. (2003). Proc. EAMT/CLAW 2003, The Joint Conference of the 7th International Workshop of the European Association for Machine Translation and the 4th Controlled Language Application Workshop, Dublin City University, Ireland, May 15–17.

Cooper, R. (1983). *Quantification and Syntactic Theory*. Dordrecht: Reidel.

Covington, M. A. (1994). *Natural language processing for Prolog programmers*. Englewood Cliffs, New Jersey: Prentice Hall.

Covington, M. A. (1996). *Natural Language Plurals in Logic Programming Queries* (Research Report AI–1996–01). Athens, Georgia: Artificial Intelligence Center, University of Georgia.

Dale, R. (1992). *Generating Referring Expression* (ACL-MIT Press Series in Natural Language Processing). Cambridge, Massachusetts, London, England: MIT Press.

Davidson, D. (1967). The Logical Form of Action Sentences. In N. Rescher (ed.), *The Logic of Decision and Action*. Pittsburgh, Pennsylvania: University of Pittsburgh Press.

Daw Elbait, G. E. (2003, to appear). *Attempto Controlled English as a Communication Language for a Flux Agent.* Master Thesis, Faculty of Computer Science, Artificial Intelligence Institute, Dresden University of Technology, Dresden.

Dörflinger, M. (2003). *Webschnittstelle für den lexikalischen Editor des Projektes Attempto.* Semesterarbeit, University of Zurich, Zurich.

Dowty, D. (1986). A Note On Collective Predicates, Distributive Predicates and *All*. In F. Marshall (ed.), *Third Eastern States Conference on Linguistics (ESCOL 86)* (pp. 97–115). Ohio State University.

Eberle, K. (1998). The Influence of Plural NPs on Aktionsarten in DRT. In F. Hamm and E.

Hinrichs (eds.), *Plurality and Quantification* (pp. 55–111). Dordrecht: Kluwer Academic Publishers.

Erbach, G. (1994). *Multi-Dimensional Inheritance.* Proc. KONVENS '94, Wien.

Erbach, G. (1995). *ProFIT 1.54 user's guide* (Research Report). Saarbrücken: Saarland University.

Erbach, G. (1995). *ProFIT: Prolog with Features, Inheritance and Templates.* Proc. EACL 95, Dublin.

Fellbaum, C. (ed.) (1998). *WordNet: An Electronic Lexical Database* (Language, Speech, and Communication). Cambridge, MA: MIT Press.

Fitting, M. (1996). *First Order Logic and Automated Theorem Proving* (2nd ed.). New York: Springer.

FRACAS Consortium. (1994a). *Harmonizing the Approaches* (FRACAS Deliverable D7). Edinburgh: Centre for Cognitive Science.

FRACAS Consortium. (1994b). *Describing the Approaches* (FRACAS Deliverable D8). Edinburgh: Centre for Cognitive Science.

FRACAS Consortium. (1994c). *The State of the Art in Computational Semantics: Evaluating the Descriptive Capabilities of Semantic Theories* (FRACAS Deliverable D9). Edinburgh: Centre for Cognitive Science.

FRACAS Consortium. (1995). *Evaluating The State of the Art* (FRACAS Deliverable D10). Edinburgh: Centre for Cognitive Science.

Franconi, E. (1993). A Treatment of Plurals and Plural Quantifications based on a Theory of Collections. *Minds and Machines, 3*, 453–474.

Fuchs, N. E. (1992). Specifications Are (Preferably) Executable. *Software Engineering Journal, 7*(5), 323–334. Reprinted in Bowen, J. P. and Hinchey, M. G. (eds.) (1999).

Fuchs, N. E. and Schwertel, U. (2002). *Reasoning in Attempto Controlled English* (Technical Report). Zurich: IFI, University of Zurich.

Fuchs, N. E. and Schwertel, U. (2003). *Reasoning in Attempto Controlled English.* Proc. Workshop on Principles and Practice of Semantic Web Reasoning (PPSWR 2003), Mumbai, India, December 13.

Fuchs, N. E., Schwertel, U. and Schwitter, R. (1999a). Attempto Controlled English – Not Just Another Logic Specification Language. In P. Flener (ed.), *Proc. Logic Programming Synthesis and Transformation, 8th International Workshop, LOPSTR '98* (Lecture Notes in Computer Science, Vol. 1559, pp. 1–20). Manchester, UK: Springer-Verlag.

Fuchs, N. E., Schwertel, U. and Schwitter, R. (1999b). *Attempto Controlled English (ACE), Language Manual, Version 3.0* (Technical Report ifi-99.03). Zurich: IFI, University of

Zurich.

Fuchs, N. E., Schwertel, U. and Torge, S. (1999a). *Controlled Natural Language Can Replace First-Order Logic*. Proc. 14th IEEE International Conference on Automated Software Engineering, Cocoa Beach, Florida, October 12–15.

Fuchs, N. E., Schwertel, U. and Torge, S. (1999b). *A Natural Language Front-End to Automatic Verification and Validation of Specifications* (Technical Report PMS-FB-1999-5). München: LMU.

Fuchs, N. E., Schwertel, U. and Torge, S. (2000). A Natural Language Front-End to Model Generation. *Journal of Language and Computation, 1*(2), 199–214.

Gärdenfors, P. (ed.) (1987). *Generalized Quantifiers. Linguistic and Logical Approaches*. Dordrecht: Reidel.

Gardent, C. and Webber, B. (2001). Towards the use of automated reasoning in discourse disambiguation. *Journal of Logic, Language and Information, 10*.

Gazdar, G. (1979). *Pragmatics: Implicature, presupposition and logical form*. New York: Academic Press.

Geurts, B. (1997). Book review of Linda M. Moxey and Anthony J. Sanford: Communicating Quantities. Lawrence Erlbaum, Hove (UK)/Hilldale (US). *Journal of Semantics, 14*, 87–94.

Gil, D. (1982). Quantifier Scope, Linguistic Variation, And Natural Language Semantics. *Linguistics and Philosophy, 5*, 421–472.

Gillon, B. (1987). The Readings of Plural Noun Phrases in English. *Linguistics and Philosophy, 10*, 199–219.

Gillon, B. (1990). Ambiguity, generality, and indeterminacy: Tests and definitions. *Synthese, 85*, 391–416.

Gillon, B. S. (1996). Collectivity and Distributivity Internal To English Noun Phrases. *Language Sciences, 18*(1-2), 443–468.

Givan, R., McAllester, D. and Shalaby, S. (1991). *Natural Language Based Inference Procedures applied to Schubert's Steamroller*. Proc. AAAI-91.

Graham, L. K. (1994). *An Implementation of Plurality in Discourse Representation Theory* (Master Thesis and Report AI-1994-04). Athens, Georgia: Artificial Intelligence Center.

Green, C. (1969). Theorem proving by resolution as a basis for question-answering systems. *Machine Intelligence, 4*, 183–205.

Grice, H. P. (1975). Logic and conversation. In P. Cole and J. L. Morgan (eds.), *Syntax and Semantics* (Vol. 3, pp. 41–58). New York: Academic Press.

Groenendijk, J. and Stokhof, M. (1991). Dynamic Predicate Logic. *Linguistics and Philosophy, 14*, 39–100.

Grover, C., Holt, A., Klein, E. and Moens, M. (2000). *Designing a Controlled Language for Interactive Model Checking.* Proc. Third International Workshop on Controlled Language Applications (CLAW 2000), Seattle, April 29–30.

Guarino, N. and Poli, R. (eds.) (1996). *Formal Ontology in Conceptual Analysis and Knowledge Representation. Special issue of the International Journal of Human and Computer Studies* (Vol. 43): Academic Press.

Hall, A. (1990). Seven Myths of Formal Methods. *IEEE Software, 7*(5), 11–19.

Hamm, F. and Hinrichs, E. (eds.) (1998). *Plurality and Quantification* (Studies in Linguistics and Philosophy, Vol. 69). Dordrecht: Kluwer.

Heim, I. (1982). *On the Semantics of Definite and Indefinite Noun Phrases.* Ph.D. Thesis, University of Massachussetts, Amherst.

Heim, I. (1983). File Change Semantics and the Familiarity Theory of Definites. In R. Bäuerle, C. Schwarze and A. von Stechow (eds.), *Meaning, Use and Interpretation of Language* (pp. 164–189). Berlin: de Gruyter.

Heim, I., Lasnik, H. and May, R. (1991). Reciprocity and Plurality. *Linguistic Inquiry, 22*, 63–101.

Hein, R. (2003). Command line interface of the lexical editor for ACE. Program and Documentation. Zurich: IFI, University of Zurich.

Herzog, O. and Rollinger, C.-R. (eds.) (1991). *Text Understanding in LILOG: Integrating Computational Linguistics and Artificial Intelligence. Final Report on the IBM Germany LILOG-Project* (Lecture Notes in Computer Science, Vol. 546). Berlin - Heidelberg - New York: Springer.

Hess, M. (1989). *Reference and Quantification in Discourse.* Habilitation Thesis, University of Zurich.

Hindle, D. and Roth, M. (1993). Structural Ambiguity and Lexical Relations. *Computational Linguistics, 19*(1), 103–120.

Hirst, G. (1987). *Semantic Interpretation and the Resolution of Ambiguity*. Cambridge: Cambridge University Press.

Hirst, G. (1997). *Context as a Spurious Concept.* Proc. AAAI Fall Symposium on Context in Knowledge Representation and Natural Language, Cambridge, MA, November 1997 (Superseded by February 2000 version).

Hirst, G. (2000). *Context as a Spurious Concept.* Proc. Conference on Intelligent Processing and Computational Linguistics, Mexico City, February.

Hobbs, J. R. (1983). *An improper treatment of quantification in ordinary English.* Proc. ACL-83, Cambridge, MA, June.

Hobbs, J. R. (1985). *Ontological Promiscuity.* Proc. 23rd Annual Meeting of the ACL, University of Chicago, Illinois, July 8-12.

Hobbs, J. R. (1996). Monotone decreasing quantifiers in a scope-free logical form. In K. van Deemter and S. Peters (eds.), *Semantic Ambiguity and Underspecification* (pp. 55–76). Stanford, CA: CSLI publications.

Hobbs, J. R. (n.a.). *The Logical Notation: Ontological Promiscuity*.Unpublished manuscript.

Hobbs, J. R. and Shieber, S. M. (1987). An algorithm for generating quantifier scopings. *Computational Linguistics, 13*(1–2), 47–63.

Hoeksema, J. (1983). Plurality and conjunction. In A. ter Meulen (ed.), *Studies in Model-Theoretic Semantics* (pp. 63–84). Dordrecht: Foris.

Holt, A., Klein, E. and Grover, C. (1999). *Natrual Language For Hardware Verification: Semantic Interpretation and Model Checking.* Proc. First Workshop on Inference in Computational Semantics (ICoS-1), Institute for Logic, Language and Computation (ILLC), Amsterdam, August 15.

Horn, L. (1972). *On the Semantic Properties of Logical Operators in English.* Ph.D. Thesis, UCLA, Distributed by Indiana University Linguistics Club, 1976.

ICoS-1. (1999). Proc. First Workshop on Inference in Computational Semantics (ICoS-1), Institute for Logic, Language and Computation (ILLC), Amsterdam, August 15.

ICoS-2. (2000). Proc. Second Workshop on Inference in Computational Semantics (ICoS-2), International Conference and Research Center for Computer Science, Schloss Dagstuhl, July 29-30.

ICoS-3. (2001). Proc. Third Workshop on Inference in Computational Semantics (ICoS-3), Siena, Italy.

Ioup, G. (1975). Some Universals for Quantifier Scope. In J. P. Kimball (ed.), *Syntax and Semantics* (Vol. 4, pp. 37–58). New York: Academic Press.

Izzo, G. J. (1993). *Incorporating Defeasible Reasoning into an Implementation of Discourse Representation Theory* (Research Report AI-1993-06). Athens, Georgia: The University of Georgia.

Johnson, M. and Klein, E. (1986). *Discourse, anaphora and parsing.* Proc. 11th International Conference on Computational Linguistics (COLING), Bonn, Germany.

Kalman, J. A. (2001). *Automated Reasoning with Otter*. Princeton: Rinton Press.

Kamp, H. (1981). A Theory of Truth and Semantic Representation. In J. Groenendijk, T. Janssen and M. Stokhof (eds.), *Formal Methods in the Study of Language* (pp. 277–322).

Amsterdam: Mathematical Centre Tract. Reprinted in Kamp (1984).

Kamp, H. (1984). A Theory of Truth and Semantic Representation. In J. Groenendijk, T. Janssen and M. Stokhof (eds.), *Truth, Representation and Information* (GRASS Series, pp. 277–322). Dordrecht: Foris.

Kamp, H. and Reyle, U. (1993). *From Discourse to Logic. Introduction to Modeltheoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. Dordrecht: Kluwer.

Keenan, E. (1987). A Semantic Definition of "Indefinite NP". In E. J. Reuland and A. ter Meulen (eds.), *The Representation of (In)definiteness*. Cambridge: MIT Press.

Keller, W. (1988). Nested cooper storage: The proper treatment of quantication in ordinary noun phrases. In U. Reyle and C. Rohrer (eds.), *Natural Language Parsing and Linguistic Theories* (pp. 432–447). Dordrecht: Reidel.

Kempson, R. M. and Cormack, A. (1981). Ambiguity and quantification. *Linguistics and Philosophy, 4*(2), 259–309.

Konrad, K. (1999). *Model Generation for Natural-Language Semantic Analysis*: Department of Computer Science, Saarland University.

Kowalski, R. (1994). Logic without Model Theory. In D. M. Gabbay (ed.), *What is a Logical System?* (pp. 35–71): Oxford University Press.

Krifka, M. (1989a). *Nominalreferenz und Zeitkonstitution: Zur Semantik von Massentermen, Pluraltermen und Aspektklassen* (Studien zur Theoretischen Linguistik). München: Wilhelm Fink Verlag.

Krifka, M. (1989b). Nominal Reference, Temporal Constitution and Quantification in Event Semantics. In R. Bartsch, J. van Benthem and P. van Emde Boas (eds.), *Semantics and Contextual Expression* (pp. 75–115). Dordrecht: Foris Publications.

Krifka, M. (1991a). How to Get Rid of Groups, Using DRT: A Case for Discourse Oriented Semantics. *Texas Linguistic Forum, 32*, 71–110.

Krifka, M. (1991b). Massenausdrücke. In A. von Stechow and D. Wunderlich (eds.), *Semantik/Semantics*. Berlin, New York: de Gruyter.

Krifka, M. (1996). Parametrized Sum Individuals for Plural Anaphora. *Linguistics and Philosophy, 19*, 555–598.

Krifka, M. (1999). At least some determiners aren't determiners. In K. Turner (ed.), *The semantics/pragmatics interface from different points of view* (Current Research in the Semantics/Pragmatics Interface, Vol. 1, pp. 257–291): Elsevier Science B.V.

Kurtzman, H. S. and MacDonald, M. C. (1993). Resolution of quantifier scope ambiguities. *Cognition, 48*, 243–279.

Lakoff, G. (1971). Semantic interpretation in generative grammar. In D. A. Steinberg and L. A. Jakobovits (eds.), *Semantics: An interdisciplinary reader in philosophy, linguistics, anthropology, and psychology*: Cambridge University Press.

Landman, F. (1989). Groups, Part I, II. *Linguistics and Philosophy, 12*, 559–605; 723–744.

Landman, F. (1998). Plurals and Maximalization. In S. Rothstein (ed.), *Events and Grammar* (Studies in Linguistics and Philosophy, Vol. 70, pp. 237–271). Dordrecht, Netherlands: Kluwer Academic Publishers.

Langendoen, T. (1978). The Logic of Reciprocity. *Linguistic Inquiry, 9*, 177–197.

Lappin, S. (ed.) (1996). *The Handbook of Contemporary Semantic Theory*. Oxford: Blackwell.

Lappin, S. and Leass, H. J. (1994). An algorithm for pronominal anaphora resolution. *Computational Linguistics, 20*(4), 535–561.

Lasersohn, P. (1989). On the readings of plural noun phrases. *Linguistic Inquiry, 20*, 179–206.

Lasersohn, P. (1990). Group action and spatio-temporal proximity. *Linguistics and Philosophy, 13*, 179–206.

Lasersohn, P. (1995). *Plurality, Conjunction, and Events*. Dordrecht: Kluwer.

Lasersohn, P. (1998). Events in the Semantics of Collectivizing Adverbials. In S. Rothstein (ed.), *Events and Grammar* (pp. 273–292). Dordrecht: Kluwer Academic Publishers.

Lehrer, A. (1986). English classifier constructions. *Lingua, 68*(2–3), 109–148.

Levin, B. (1993). *English Verb Classes and Alternations: A Preliminary Investigation*. Chicago, IL: University of Chicago Press.

Levinson, S. C. (1984). *Pragmatics*. Cambridge: Cambridge University Press.

Link, G. (1983). The Logical Analysis of Plurals and Mass Terms: A Lattice-Theoretical Approach. In R. Bäuerle, C. Schwarze and A. von Stechow (eds.), *Meaning, Use, and Interpretation of Language* (pp. 302–323). Berlin: de Gruyter. Reprinted as Chapter 1 of Link (1998b).

Link, G. (1984). Hydras. On the logic of relative constructions with multiple heads. In F. Landman and F. Veltman (eds.), *Varieties of Formal Semantics* (pp. 245–257). Dordrecht: Foris. Reprinted as Chapter 3 of Link (1998b).

Link, G. (1987). Generalized Quantifiers and Plurals. In P. Gärdenfors (ed.), *Generalized Quantifiers. Linguistic and Logical Approaches* (pp. 151–180). Dordrecht: Reidel. Reprinted as Chapter 4 of Link (1998b).

Link, G. (1991). Plural. In A. von Stechow and D. Wunderlich (eds.), *Semantik/Semantics* (pp. 418–440). Berlin, New York: de Gruyter. Translated and Reprinted as Chapter 2 of Link (1998b).

Link, G. (1998a). Ten Years of Research on Plurals – Where Do We Stand? In F. Hamm and E. Hinrichs (eds.), *Pluraltiy and Quantification* (Studies in Linguistics and Philosphy, Vol. 69, pp. 19–54). Dordrecht: Kluwer. Reprinted as Chapter 7 of Link (1998b).

Link, G. (1998b). *Algebraic Semantics in Language and Philosophy*. Stanford: CSLI Publications.

Link, G. and Schütze, H. (1991). The Treatment of Plurality in L-LILOG. In O. Herzog and C. R. Rollinger (eds.), *Text Understanding in LILOG: Integrating Computational Linguistics and Artificial Intelligence.* (Lecture Notes in Computer Science, Vol. 546, pp. 342–352). Berlin, Heidelberg, New York: Springer.

Löbner, S. (1985). Definites. *Journal of Semantics, 4*, 279–326.

Löbner, S. (1987). Natural Language and Generalized Quantifier Theory. In P. Gärdenfors (ed.), *Generalized Quantifiers. Linguistic and Logical Approaches.* (pp. 181–201). Dordrecht: Reidel.

Lønning, J. T. (1987). Collective Readings of Definite and Indefinite Noun Phrases. In P. Gärdenfors (ed.), *Generalized Quantifiers. Linguistic and Logical Approaches.* (pp. 203–235). Dordrecht: Reidel.

Lønning, J. T. (1989). *Some Aspects of the Logic of Plural Noun Phrases.* Ph.D. Thesis and COSMOS-Report 11, Department of Mathematics, University of Oslo.

Lønning, J. T. (1991). Among Readings. Some comments on "Among collections". In J. van der Does (ed.), *Quantification and Anaphora II* (DYANA Deliverable 2.2.b, pp. 37–51). Edinburgh: Centre for Cognitive Science, University of Edinburgh.

Lønning, J. T. (1997). Plurals and collectivity. In J. van Benthem and A. ter Meulen (eds.), *Handbook of Logic and Language* (pp. 1009–1053). Amsterdam: Elsevier Science Press.

Macias, B. and Pulman, S. G. (1995). A Method for Controlling the Production of Specifications in Natural Language. *The Computer Journal, 38*(4), 310–318.

Manthey, R. and Bry, F. (1988). SATCHMO: A Theorem Prover Implemented in Prolog. In E. L. Lusk and R. A. Overbeek (eds.), *Proc. CADE 88, Ninth International Conference on Automated Deduction* (Lecture Notes in Computer Science, Vol. 310, pp. 415–434). Argonne, Illinois: Springer.

McCune, W. W. (1994). *Otter 3.0 Reference Manual and Guide* (Technical Report ANL-94/6). Argonne, Illinois: Argonne National Laboratory.

McCune, W. W. (2001). *Mace 2.0 Reference Manual and Guide* (Technical Memorandum ANL/MCS-TM-249): Argonne National Laboratory.

Mellish, C. S. (1985). *Computer Interpretation of Natural Language Descriptions*. Chichester, England: Ellis Horwood.

Milsark, G. L. (1977). Towards an Explanation of Certain Peculiarities of the Existential Construction in English. *Linguistic Analysis, 3*, 1–29.

Moltmann, F. (1990a). Semantic Selection and the Determination of the Part Structures in the Use of Natural Language. In M. Stokhof (ed.), *Proceedings of the Seventh Amsterdam Colloquium in Semantics*. Amsterdam: ITLI, University of Amsterdam.

Moltmann, F. (1990b). The Multidimensional Part Structure of Events. In A. Halpern (ed.), *Proceedings of the West Coast Conference on Formal Linguistics (WCCFL) 9* (pp. 361–378). Stanford: Stanford Linguistics Student Association, Stanford University.

Moltmann, F. (1992). Reciprocals and *same/different*: Towards a semantic analysis. *Linguistics and Philosophy, 15*, 411–462.

Moltmann, F. (1995). *Complex Part Structures and Natural Language* (Lecture Notes for the Seventh European Summer School in Language, Logic and Information). Barcelona.

Moltmann, F. (1997). *Parts and Wholes in Semantics*. New York: Oxford University Press.

Montague, R. (1973). The proper treatment of quantification in ordinary English (PTQ). In J. Hintikka and et al. (eds.), *Approaches to Natural Language* (pp. 221–242). Dordrecht: Dordrecht, Reidel. Reprinted as chapter 8 in R. Montague (1974).

Montague, R. (1974). *Formal Philosophy: Selected Papers of Richard Montague. Edited and with an introduction by Richmond H. Thomason*. New Haven and London: Yale University Press.

Monz, C. and de Rijke, M. (2000). Inference in Computational Semantics. *Journal of Language and Computation, 1*, 151–158.

Moran, D. B. and Pereira, F. C. (1992). Quantifier Scoping. In H. Alshawi (ed.), *The Core Language Engine* (pp. 235–250). Cambridge, MA: MIT Press.

Mostowski, A. (1957). On a generalization of quantifiers. *Fundamenta Mathematicae, 44*, 12–36.

Muskens, R. and Thijsse, E. (eds.) (2001). *Computing Meaning* (Vol. 2): Kluwer Academic Publishers.

Pafel, J. (1988). *Die Parameter des relativen Quantorenskopus im Deutschen* (LILOG-Report 48). Stuttgart: IBM Deutschland GmbH.

Parsons, T. (1990). *Events in the Semantics of English: A Study in Subatomic Semantics* (Current Studies in Linguistics). Cambridge, MA: The MIT Press.

Partee, B. H. (1989). *Many Quantifiers.* Proc. 5th Eastern States Conference on Linguistics (ESCOL), Columbus, Ohio State University.

Partee, B. H., ter Meulen, A. and Wall, R. E. (1990). *Mathematical Methods in Linguistics*. Dordrecht: Kluwer Academic Publishers.

Pinkal, M. (1985). *Logik und Lexikon: Die Semantik des Unbestimmten*. Berlin: de Gruyter. Translated and edited as Pinkal (1995).

Pinkal, M. (1991). Vagheit und Ambiguität. In A. von Stechow and D.þWunderlich (eds.), *Semantik/Semantics* (Handbücher zur Sprach- und Kommunikationswissenschaft, Vol. 6, pp. 250–269). Berlin, New York: de Gruyter.

Pinkal, M. (1995). *Logic and Lexicon. The semantics of the indefinite.* Dordrecht: Kluwer. Revised and edited English translation of Pinkal (1985).

Poesio, M. (1993). *Assigning a semantic scope to operators.* Proc. ACL-93, Columbus, OH, June.

Poesio, M. (1994). *Discourse Interpretation and the Scope of Operators.* Ph.D. Thesis, Department of Computer Science, University of Rochester, Rochester, NY.

Poesio, M. (1996). Semantic Ambiguity and Perceived Ambiguity. In K. van Deemter and S. Peters (eds.), *Ambiguity and Underspecification* (pp. 159–201). Stanford: CSLI Publications.

Poesio, M. and Vieira, R. (1998). Corpus-based investigation of definite description use. *Computational Linguistics, 24*(2), 183–216.

Quirk, R., Greenbaum, S., Leech, G. and Svartvik, J. (1985). *A Comprehensive Grammar of the English Language*. London: Longman.

Reichenbach, H. (1947). *Elements of Symbolic Logic*. London: Macmillan.

Reyle, U. (1993). Dealing with Ambiguities by Underspecification: Construction, Representation and Deduction. *Journal of Semantics, 10*, 123–178.

Reyle, U. (1994). Monotonic Disambiguation and Plural Pronoun Resolution. In H. Kamp (ed.), *Ellipsis, Tense and Questions* (DYANA-2 Deliverable, Vol. R2.2.B). University of Stuttgart.

Reyle, U. (1995). Co-Indexing Labelled DRSs to Represent and Reason with Ambiguities. In K. van Deemter and S. Peters (eds.), *Ambiguity and Underspecification*. Stanford: CSLI Publications.

Reyle, U. and Gabbay, D. M. (1994). Direct Deductive Computation on Discourse Representation Structures. *Linguistics and Philosophy, 17*, 343–390.

Roberts, C. (1987). *Modal Subordination, Anaphora and Distributivity.* Ph.D. Thesis, University of Massachusettes, Amherst.

Rothstein, S. (ed.) (1998). *Events and Grammar* (Studies in Linguistics and Philosophy, Vol. 70). Dordrecht: Kluwer Academic Publishers.

Scha, R. (1981). Distributive, Collective and Cumulative Quantification. In J. Groenendijk, T. Janssen and M. Stokhof (eds.), *Formal Methods in the Study of Language.* (pp. 483–512).

Amsterdam: Mathematical Centre.

Scha, R. and Stallard, D. (1988). *Multi-Level Plurals and Distributivity.* Proc. 26th Annual Meeting of the ACL, Buffalo, NY, 1988.

Schein, B. (1993). *Plurals and Events*. Cambridge, MA: MIT Press.

Schubert, L. K. and Hwang, C. H. (2000). Episodic Logic Meets Little Red Riding Hood: A comprehensive, natural representation for language understanding. In L. Iwanska and S. C. Shapiro (eds.), *Natural Language Processing and Knowledge Representation: Language for Knowledge and Knowledge for Language* (pp. 111–174). Menlo Park, CA, and Cambridge, MA: MIT/AAAI Press.

Schubert, L. K. and Pelletier, F. J. (1982). From English to Logic: Context-Free Computation of 'Conventional' Logic Translations. *American Journal of Computational Linguistics, 10*, 165–176.

Schütze, H. (1989). *Pluralbehandlung in natürlichsprachlichen Wissensverarbeitungssystemen.* Diploma Thesis, Institut für Informatik, University of Stuttgart, Stuttgart.

Schwarzschild, R. (1990). *Against groups.* Proc. Seventh Amsterdam Colloquium, Amsterdam.

Schwarzschild, R. (1991a). *On the Meaning of Definite Plural Noun Phrases.* Ph.D. Thesis, University of Massachusetts, Amherst.

Schwarzschild, R. (1991b). *'Together' as Non-Distributivity Marker.* Proc. Eighth Amsterdam Colloquium, University of Amsterdam.

Schwarzschild, R. (1992). Types of Plural Individuals. *Linguistics and Philosophy, 15*, 641–675.

Schwarzschild, R. (1994). Plurals, Presuppositions, and the Sources of Distributivity. *Natural Language Semantics, 2*, 201–248.

Schwarzschild, R. (1996). *Pluralities*. Dordrecht: Kluwer.

Schwarzschild, R. (2002). The Grammar of Measurement. In B. Jackson (ed.), *Proc. Semantics and Linguistic Theory XII*. Ithaca: CLC Publications, Department of Linguistics, Cornell University.

Schwertel, U. (1993). *Probleme der Pluralsemantik* (Research paper 58). University of Konstanz: Fachgruppe Sprachwissenschaft.

Schwertel, U. (1995). *Plurals, Ambiguity and Underspecification.* M.Sc. Thesis, Centre for Cognitive Science, University of Edinburgh, Edinburgh.

Schwertel, U. (2000). *Controlling Plural Ambiguities in Attempto Controlled English.* Proc. 3rd International Workshop on Controlled Language Applications (CLAW), Seattle, Washington, April 29–30.

Schwertel, U., Fuchs, N. E. and Höfler, S. (2003). *Attempto Controlled English (ACE), Language Manual, Version 4.0* (Technical Report). Zurich: IFI, University of Zurich.

Schwitter, R. (1998). *Kontrolliertes Englisch für Anforderungsspezifikationen.* Ph.D. Thesis, Department of Computer Science, University of Zurich.

Schwitter, R., Ljungberg, A. and Hood, D. (2003). *ECOLE - A Look-ahead Editor for a Controlled Language.* Proc. EAMT/CLAW 2003, The Joint Conference of the 7th International Workshop of the European Association for Machine Translation and the 4th Controlled Language Application Workshop, Dublin City University, Ireland, May 15–17.

Schwitter, R., Mollá Aliod, D. and Hess, M. (1999). *ExtrAns - Answer Extraction from Technical Documents by Minimal Logical Forms and Selective Highlighting.* Proc. Third International Tbilisi Symposium on Language, Logic and Computation, Batumi, Georgia, September 12-16.

Sowa, J. F. (1991). Toward the expressive power of natural language. In J. F. Sowa (ed.), *Principles of Semantic Networks*. San Mateo, Calif.: Kaufmann.

Sowa, J. F. (2000). *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. Pacific Grove, CA: Brooks Cole Publishing Co.

Sternefeld, W. (1993). *Plurality, Reciprocity, and Scope* (SfS-Report 13-93). University of Tübingen: Seminar für Sprachwissenschaft.

Stickel, M. E. (1986). Schubert's Steamroller Problem: Formulation and Solutions. *Journal of Automated Reasoning, 2*(1), 89–101.

Sukkarieh, J. (2001a). *Quasi-NL Knowledge Representation for Structurally-Based Inferences.* Proc. 3rd Workshop on Inference in Computational (ICoS-3), Siena, Italy.

Sukkarieh, J. (2001b). *Natural Language for Knowledge Representation.* Ph.D. Thesis, Computer Laboratory, University of Cambridge.

Sutcliffe, G. (1999). *Overview of Automated Theorem Proving* (available at http://www.cs.miami.edu/~tptp/OverviewOfATP.html). University of Miami: Department of Computer Science.

Tarnawsky, G. O. (1982). *Knowledge semantics.* Ph.D. Thesis, Department of Linguistics, New York University.

Tjan, B. S., Gardiner, D. A. and Slagle, J. R. (1992). Representing and Reasoning with Set Referents and Numerical Quantifiers. In T. E. Nagle, J. A. Nagle, L. L. Gerholz and P. W. Eklund (eds.), *Conceptual structures, Current Research and Practice* (pp. 53–66). New York: Ellis Horwood.

Torge, S. (1998). *Überprüfung der Erfüllbarkeit im Endlichen: Ein Verfahren und seine Anwendung.* Ph.D. Thesis, Department of Computer Science, University of Munich, Munich.

van Deemter, K. and Peters, S. (eds.) (1996). *Ambiguity and Underspecification* (CSLI Lecture Notes). Stanford, CA: CSLI.

van den Berg, M. H. (1994). Plurality. In *The Encyclopedia of Language and Linguistics* (Vol. 6, pp. 3198–3200). Oxford: Pergamon.

van den Berg, M. H., Bod, R. and Scha, R. (1994). *A Corpus-based Approach to Semantic Interpretation.* Proc. Ninth Amsterdam Colloquium, ILLC, Department of Philosophy, University of Amsterdam.

van der Does, J. (1991). Among Collections. In J. van der Does (ed.), *Quantification and Anaphora II* (DYANA Deliverable, Vol. 2.2.b, pp. 1–35). Edinburgh: Centre for Cognitive Science, University of Edinburgh.

van der Does, J. (1992). *Applied Quantifier Logics.* Ph.D. Thesis, University of Amsterdam.

van der Does, J. (1993). Sums and Quantifiers. *Linguistics and Philosophy, 16*, 509–550.

van der Does, J. (1994). On Complex Plural Noun Phrases. In M. Kanazawa and C. Pinón (eds.), *Dynamics, Polarity and Quantification* (CSLI Lecture Notes, Vol. 48, pp. 81–115). Stanford, California: CSLI Publications.

van der Does, J. and Verkuyl, H. (1995). Quantification and Predication. In K. van Deemter and S. Peters (eds.), *Ambiguity and Underspecification*. Stanford, CA: CSLI.

VanLehn, K. A. (1978). *Determining the scope of English quantifiers* (Technical Report AI-TR-483). Cambridge, MA: Artificial Intelligence Laboratory, MIT.

Verkuyl, H. (1988). Aspectual Asymmetry and Quantification. In V. Ehrich and H. Vater (eds.), *Temporalsemantik. Beiträge zur Linguistik der Zeitreferenz* (pp. 220–259). Tübingen: Niemeyer Verlag.

Verkuyl, H. (1992). Distributivity and Collectivity: a couple at odds. In M. Kanazawa and C. Pinón (eds.), *Dynamics, Polarity and Quantification* (CSLI Lecture Notes, Vol. 48, pp. 49–80). Stanford, California: CSLI Publications.

Verkuyl, H. (1993). *A Theory of Aspectuality: the Interaction between Temporal and Atemporal Structure* (Cambridge Studies in Linguistics, Vol. 64). Cambridge: Cambridge University Press.

Verkuyl, H. (1999). *Aspectual Issues. Studies on Time and Quantity.* (CSLI Lecture Notes, Vol. 98). Stanford, California: CSLI Publications.

Verkuyl, H. and van der Does, J. (1991). *The Semantics of Plural Noun Phrases* (ITLI Prepublication LP-91-7). University of Amsterdam: Institute for Language, Logic and Information.

Verkuyl, H. and van der Does, J. (1996). The Semantics of Plural Noun Phrases. In J. van der Does (ed.), *Quantifiers, Logic, and Language* (pp. 337–374). Stanford: CSLI Publications.

Vieira, R. (1998). A review of the linguistic literature on definite descriptions. *Acta Semiotica et Linguistica, 7*, 219–258.

Vieira, R. and Poesio, M. (2000). An Empirically-Based System for Processing Definite Descriptions. *Computational Linguistics, 26*(4), 525–579.

von Stechow, A. and Wunderlich, D. (eds.) (1991). *Semantik/Semantics. Ein internationales Handbuch der zeitgenössischen Forschung. An International Handbook of Contemporary Research* (Handbücher zur Sprach- und Kommunikationswissenschaft, Vol. 6). Berlin, New York: de Gruyter.

Westerståhl, D. (1989). Quantifiers in formal and natural languages. In D. Gabbay and F. Guenthner (eds.), *Handbook of Philosophical Logic* (Vol. IV, pp. 1–131). Dordrecht: Reidel.

Wing, J. M. (1988). A Study of 12 Specifications of the Library Problem. *IEEE Software, 5*(4), 66–76.

Wolff, S. R., Macleod, C. and Meyers, A. (1998). *COMLEX Word Classes Manual* (Manual). New York: Proteus Project, New York University.

Woods, W. (1977). Semantics and quantification in natural language question answering. In *Advances in Computers* (Vol. 17, pp. 1–87): Academic Press.

Zwarts, F. (1998). Three Types of Polarity. In F. Hamm and E. Hinrichs (eds.), *Plurality and Quantification* (Studies in Linguistics and Philosophy, Vol. 69, pp. 177–238). Dordrecht: Kluwer Academic Publishers.

Zwicky, A. and Sadock, J. (1975). Ambiguity tests and how to fail them. In J. Kimball (ed.), *Syntax and Semantics* (Vol. 4). New York: Academic Press.

Index

B

C

E

F

G

H

Q

Curriculum Vitae

Uta Schwertel was born in Villingen, Germany, on March 17, 1969. She attended the primary school (1975–1979) in Nußbach and the Schwarzwaldgymnasium (1979–1988) in Triberg, Germany. From October 1988 to October 1994 she studied Theoretical Linguistics and Philosophy at the University of Konstanz, Germany, where she graduated as M.A. in October 1994 with a Master Thesis on formal semantics of plurals which was supervised by Prof. Dr. Urs Egli and Prof. Dr. Arnim von Stechow. From October 1994 to October 1995 Uta Schwertel completed a Postgraduate Programme at the Centre for Cognitive Science, University of Edinburgh, UK, where she graduated as *M.Sc. in Cognitive Science and Natural Language* with a thesis on ambiguity and underspecification which was supervised by Dr. Massimo Poesio and Prof. Dr. Robin Cooper. From November 1995 to December 1996 Uta Schwertel worked as a research scientist at the Centre for Logic and Linguistics at IBM Research Heidelberg, Germany. She worked on semantic construction within the machine translation project Verbmobil under the direction of Prof. Dr. Peter Bosch and in direct collaboration with Dr. Markus Egg. In February 1997 Uta Schwertel joined as a research assistant and Ph.D. student the Requirements Engineering Research Group of Prof. Dr. Martin Glinz at the Department of Information Technology (*Institut für Informatik*) of the University of Zurich, Switzerland. Uta Schwertel was working within the projects *Attempto – Controlled English for Requirements Specifications* and *Practical Applications of Attempto Controlled English* which were lead by Dr. Norbert E. Fuchs. Under the supervision of Dr. Fuchs and Prof. Dr. Michael Hess, who is head of the *Institute of Computational Linguistics* at the University of Zurich, Uta Schwertel was pursuing her research in the field of computational plural semantics which she completed with this Ph.D. thesis in December 2003.