Some aspects of

The Internal Structure of Discourse

The Dynamics of Nominal Anaphora

Martin H. van den Berg

The Internal Structure of Discourse

ILLC Dissertation Series 1996-3

institute for logic, language and computation

For further information about ILLC-publications, please contact

Institute for Logic, Language and Computation
Universiteit van Amsterdam
Plantage Muidergracht 24
1018 TV Amsterdam
phone: +31-20-5256090
fax: +31-20-5255101
e-mail: illc@fwi.uva.nl

Some aspects of

The Internal Structure of Discourse

The Dynamics of Nominal Anaphora

Academisch Proefschrift

ter verkrijging van de graad van doctor aan de
Universiteit van Amsterdam,
op gezag van de Rector Magnificus
Prof.dr P.W.M. de Meijer
ten overstaan van een door het college van dekanen ingestelde commissie
in het openbaar te verdedigen in de Aula der Universiteit
op donderdag 7 maart 1996 te 12.00 uur

door

Martin H. van den Berg

geboren te Amsterdam.

Promotores:
Prof.dr. J.F.A.K. van Benthem
Faculteit Wiskunde en Informatica
Universiteit van Amsterdam
Plantage Muidergracht 24
1018 TV Amsterdam

Prof.dr. R. Scha
Letteren faculteit
Universiteit van Amsterdam
Spuistraat 132
1012 VB Amsterdam

Co-Promotor:
dr. J. M. van der Does
Faculteit Wiskunde en Informatica

Printed and bound by CopyPrint 2000, Enschede.

ISBN 90–74795–45–5

Contents

1 **The Internal Structure of Discourse** ***1***
1 *Long Distance Effects* *2*
1.1 Linkage *4*
1.2 Donkey Sentences *6*
1.3 Determiners in General *8*
1.4 Arbitrary Determiners as Antecedent *9*
1.5 Arbitrary Determiners as Anaphor *11*
1.6 Overview *13*
2 *Dynamic Semantics* *15*
2.1 The Interpretation of Programs *15*
2.2 The Dynamic Interpretation of Text *16*
2.3 Groenendijk and Stokhof *17*
2.4 Truth Conditions of DPL Formulas *23*
2.5 Partial Logic *25*
3 *Generalized Quantifiers* *28*
3.1 General Properties *28*
3.2 Special Properties *29*
4 *Plurality* *30*
4.1 Higher Order Sets *31*
4.2 Quantifier Readings *32*
4.3 Domains and Predicates *34*
4.4 Scope Ambiguities *34*
4.5 Distributivity *36*
Conclusions *36*
2 **Partial Dynamic Predicate Logic** ***39***
1 *Dynamic Predicate Logic* *39*
1.1 Generalized Quantifiers in DPL *40*
1.2 Negation and other Closure Operators in DPL *43*
2 *Partializing the Transitions* *45*
2.1 Partiality and Negation *45*
2.2 A Simple 3-valued Dynamic Predicate Logic *47*
2.3 A Comparison between DPL and DPL* *50*

2.4 Strengthening Negation *52*
2.5 A Comparison between DPL and DPL$^{\neg}$ *56*
3 *Partializing the Truth Conditions* *57*
3.1 Static Boolean Operators *58*
3.2 Additional Unary Operators *59*
3.3 Additional Binary Operators *61*
4 *Partializing the States* *62*
4.1 Partial States *62*
4.2 Atomic Actions *64*
4.3 Undefinedness of Transitions and of States *66*
4.4 Full Dynamic Predicate Logic *67*
5 *Logical Issues* *68*
5.1 Predicate Logic *70*
5.2 Bisimulation *71*
5.3 Partializing the states *76*
Conclusions *78*
3 **Plurality and Generalized Quantifiers** ***81***
1 *Plurality* *82*
1.1 Maximization *82*
1.2 Conservativity *83*
1.3 Readings *84*
2 *Plural Predicate Logic* *87*
2.1 Distributives *89*
2.2 Varieties of Quantification *90*
2.3 Downward Monotonic Quantifiers *92*
2.4 Packaging *93*
3 *Independent Plural Logic* *94*
3.1 Arguments from Dynamics *95*
3.2 Dependencies in Static Logic *96*
3.3 Independent Plural Predicate Logic *99*
4 *Partial Plural Logic* *105*
4.1 Partial Logic *105*
4.2 Partial Assignments *109*
5 *Logical Issues* *109*
5.1 Van der Does, 1992 *110*
5.2 Formal Properties *113*
5.3 Logical Properties *114*
5.4 Collective Dependency Models *115*
5.5 Equivalence of IPPL and PPL (remainder of proof) *118*
Conclusions *120*
4 **Dynamic Plural Logic** ***123***

1 *Dynamic Semantics for Plural* *123*
1.1 Relations *123*
1.2 Generalized Quantifiers and Plurals *124*
2 *States for Dependent Plurals* *126*
2.1 The Nature of States *126*
2.2 A Formal Definition of States *129*
2.3 Properties Holding on States *130*
2.4 Operations on States *133*
3 *Dynamic Plural Predicate Logic* *135*
3.1 Core Logic *135*
3.2 Additional Random Assignments *137*
3.3 Plural Operators *138*
4 *Quantification* *149*
4.1 Dynamic Quantifiers *149*
4.2 Readings of Quantifiers *151*
5 *Logical Issues* *153*
5.1 Definability *153*
5.2 The Logical Landscape *154*
Conclusions *155*
5 **Applications** ***157***
1 *General Principles* *158*
1.1 Some Properties of Generalized Quantifiers *158*
1.2 Singular Referents *159*
2 *Standard Cases* *161*
2.1 Simple Cases *161*
2.2 A Man and his Donkey *163*
2.3 A Sentence that Caused DPL some Problems *164*
2.4 A Sentence that Caused DPL many Problems *165*
3 *General Cases* *166*
3.1 The Benchmark Lists of Chapters 1 and 4 *166*
3.2 Monotonicity Properties and Readings *170*
4 *Information Packaging* *172*
4.1 Adjectival Quantifiers *174*
4.2 Cumulatives and Branching Readings *174*
5 *Future Extensions: Other Anaphora* *176*
5.1 Other Relations *176*
5.2 Other Nominal Anaphors *177*
5.3 Committees and Couples *180*
6 *Future Extensions: Non-nominal Anaphora* *181*
7 *Future Extensions: Discourse Grammar* *182*
7.1 Parallel Structure *182*

7.2 Interactions with VP's *183*
8 *Conclusions of the Thesis* *184*
8.1 General Conclusions *184*
8.2 Conclusions: Linguistics *184*
8.3 Conclusions: Logic *185*

Preface and Acknowledgments

My interest in philosophy originates in high-school, and two teacher were important for this: my Latin teacher Snijder, and my English teacher Pastor. Together with my class-mate Ben Schomakers, I read philosophy, of which Wittgenstein was the one that stuck. He chose mathematics, I physics, but both of us were more interested in philosophy. With him I did my first course in philosophy of language, Fred Landman's Language and Reality (Taal en Werkelijkheid). It was love at first sight. The next step in my development came some years later, during a talk in the Montague Colloquium by Livia Polanyi on *the Bouncing Baby Bunny*. This gave my three things, an understanding for the deep problems posed by real language texts and their recursive structure, the acquaintance with Livia, and the address of the best Spanish restaurant in Amsterdam. Next to these, probably the strongest influences on my model of reality were Troelstra's courses on constructive mathematics.

After a long period of "not knowing what to do", I woke up, years later, finding myself with large parts of the psychology and foundations of mathematics curriculum, a (doctoraal) degree in physics and an almost-degree in philosophy. I was slightly confused to find myself having a paid graduate position at Computational Linguistics.

The research reported in this dissertation was done from 1988 to 1995. Paul Dekker and Jaap van der Does were both formidable opponents in a large number of discussions. I learned a lot in these and hope they too got something from the bargain. It does turn out, that the theories we developed later may seem really different, but, somewhere deep down, are very similar.

Having mentioned one of my promotores, let me mention the others. Remko Scha was a good supervisor during the development of my ideas as well as the final stages. Most of our cooperation did not make it into this dissertation. Johan van Benthem joined the supervising team late, in the last two years, but was invaluable. He pointed me at a large number of interrelations of my research with research in mathematics. More important, he forced me to stop expanding and improving the theory, and start writing it down.

One thing I regret. Of all people I worked with, I worked best with Hub Prüst. He decided against science and went somewhere else. Although all should do what they like best, I consider his disappearance a loss to linguistics, and still think his work on VP-anaphora is damn good. Sadly, then, that everything I did either directly with him or based on our cooperation, failed to survive the final editing stage. Not, I hasten to add, because of quality but because it constituted too different a topic. If you see work of me on anaphora resolution based on discourse grammar, you know where it came from.

Mentioning the final editing stage: I am eternally grateful to my friends Livia, Andras and Mishka, for having me in their house for a few months. It is important to be away from it all when writing your thesis, and away from it all I was. I miss their company immensely.

Except maybe for forgetting to quote one of your supervisors, nothing is more hazardous that writing the acknowledgments. A large number of people influenced my ideas over the years, and it is risky to try to mention them all, because I may forget someone who was essential, but happens to not be in my directly accessible memory on the moment I type this, but here it goes: Renate Bartsch, David Beaver, Dorit Ben-Shalom, Ágnes Bende-Farkas, Johan van Benthem, Inge Bethke, Rens Bod, Astrid Bredt, Gennaro Chierchia, Kees Deemter, Paul Dekker, Jaap van der Does, Kees Doets, Donka Farkas, Tim Fernando, Chris Fox, Jeff Goldberg, Jeroen Groenendijk, Kinga Gárdai, Petra Hendriks, Herman Hendriks, Theo Janssen, László Kálmán, Andras Kornai, Fred Landman, Serge Lapierre, Noor van Leusen, Ieke Moerdijk, Friederike Moltmann, Reinhard Muskens, Marlies Nieberg, Barbara Partee, Livia Polanyi, Hub Prüst, Janina Rado, Craige Roberts, Remko Scha, Jan Scholten, Martin Stokhof, Anne Troelstra, Frank Veltman, Henk Verkuyl and Henk Zeevat.

And, in a more social way, Dik Bakker, Bram Boskamp, Mehdi Dastani, Marijke Elstrodt, Henk-Jan Honing, Christien Kool, Kwee Tjoe Liong, Neza van der Leeuw, Joos Skolnik, and all the students of the department of Computational Linguistics, for being good company. Of these colleagues I have to separately mention Mark van den Berg, because he is the department member that made it necessary for me to start using my middle name.

Martin H. van den Berg
Almere, Januari, 1996

1

The Internal Structure of Discourse

Introduction

This thesis deals with the contribution that terms in a discourse make to subsequent terms in that discourse. This is not an original subject. Discourse semantics, or dynamic semantics as it is often called, is one of the central topics of modern semantics and has been since Kamp's and Heim's seminal papers from 1981. In this thesis dynamic semantics is combined with two other core segments of formal semantics, the theory of plurals, and the theory of generalized quantifiers. The result is a comprehensive formalization of the dynamics of noun-phrases that tries to contribute to a number of issues.

Linguistically, the resulting theory can be used as the semantic component of a discourse grammar. Logically, it gives insights in what *can* be formalized using variations on the standard dynamic binding notions.

The insight gained in the case of plural dynamics is the recognition that plural objects contain elements, and that different plural objects can be related, not only directly, but also because their elements are. Another, more indirect, contribution lies in a more secure rooting of the theory into the received formalism of plural logic by making this logic directly dynamic, rather than importing plural logic into a pre-existing dynamic framework.

The contribution to the theory of generalized quantifiers is that this gives a comprehensive non-ad hoc definition of an externally dynamic notion of generalized quantifiers.

The relevance to the field of relational algebra is that it gives a number of new insights into the properties of partial relational algebra. It is shown that this is not just a partialized version of something well-known, but has in fact interesting properties of its own.

The Structure of this Thesis

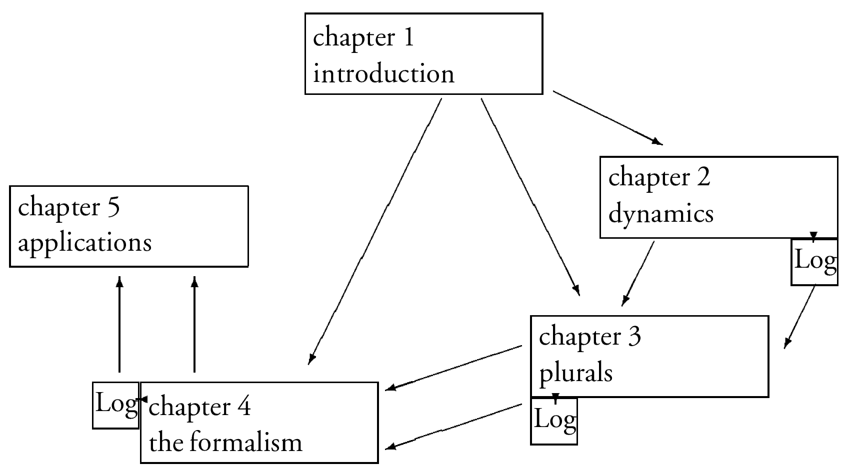
This thesis deals with one subject. I investigate how far one gets formalizing discourse referents using only modifications of the (dynamically) bound variable approach.

This chapter gives some background information needed to understand the rest of this thesis. It consists of four sections. First, in section 1 I discuss the phenomena that I will formalize in this thesis. The next three sections give an introduction to the formal tools needed to understand the thesis. In section 2 I introduce the logical tools needed for chapter 2. I discuss the basics of dynamic semantics, followed by a short discussion of partial logic. These will be combined in chapter 2 to give an improved treatment of negation and other operators. In section 3 and 4, I turn to the components needed for chapter 3: generalized quantifier theory and the received theory of plurals.

Chapter 2 defines the dynamic logical language I use to attack these problems. Chapter 3 discusses plurals and rephrases the received theory of plurals and quantification in a way compatible with the language of chapter 2. In chapter 4, these two sides are combined to form one theory of plural dynamics.

Chapters 2 and 3 only serve to argue for the formal decisions made in chapter 4. It is possible to read chapter 4 without reading chapters 2 and 3. Also, the chapters 2, 3 and 4 all have a section 5 called *logical issues*, that discusses some of the subjects of a more logical nature that arise in the chapter and briefly point out the more prominent aspects of these. However, nothing in these logical sections is presupposed by what follows. Finally, in chapter 5, the theory is applied to the phenomena discussed in chapter 1, and some more suggestions for applications are given. Besides the possibility to skip certain chapters and certain sections, there are also so called *asides*. These are boxes containing a digression on the main text. I prefer this format over a special section containing the digression, because that suggests that it is part of the main text, whereas a box, floating around stresses that it does not belong to the main discussion. If you want, you may think of these as notes to the whole section, rather than to a word or a sentence.

This leads to the existence of several routes through this thesis.



1 Long Distance Effects

Introduction

The study of long distance effects was stirred up by Geach (1972) and Evans (1977,1980). According to Geach, there are essentially two uses of pronouns in natural language. They are either used as bound variables or as pronouns of laziness. The variable use is illustrated by *Every woman loves her mother*. Here *her* is bound by the quantifier *Every woman*. Pronouns of laziness are illustrated by *A* *man who brings his wife flowers is better off than one who brings her vegetables*, where *one* essentially means *a man* and *her* is a placeholder for the phrase *his wife*.

Evans points out that simple placeholdership will not always do. The example *John has some sheep. He keeps them in a barn* illustrates his problem. The pronoun does not stand for *some sheep*, this would imply that only some of John's sheep are kept in the barn, whereas the sentence in fact means that all of John's sheep are kept there. Evans argues that these kind of pronouns, which he calls E-type pronouns (probably after his favorite car), are like pronouns of laziness, but do not directly copy earlier syntactic material. For example, the meaning of *them* above means *the sheep John owns*.

The actual meaning that a pronoun stands for can get very complicated. This belies the obvious observation that the meaning of the anaphor seems to be directly determined by its antecedent rather than via some complex copying. In fact, if the antecedent itself contains anaphoric material, it in its turn has to be resolved by copying in material, and this can go on for a while. It is this that Hans Kamp (1981, 1993) and Irene Heim (1982, 1983, 1990) picked up and began solving in a series of articles and books.

There are in fact at least two separate claims to be found in the work on discourse representations. First, it is the realization that *the sentence is not the only thing of interest in language*, nor even the most important one. Inter-sentential effects are needed to explain a number of phenomena in language. This does not only apply to the usual domain of discourse representation theory: pronouns, but also to phenomena like VP-anaphora and gapping which can only really be understood if the sentence is taken in its context (cf. Prüst 1992).

The other claim is that *what a pronoun refers to is not to an entity in the world* —or, if you prefer, in the model— *but to a preceding expression*. This is mediated by discourse referents (Kamp) or files (Heim).

This latter claim is often misinterpreted to mean that there is an essential intermediate level of representations. Formally, this is not exactly what it means, although it is surely one way of implementing anaphoric reference. If this is what it meant, then the same would already hold for standard predicate logic (PL). Variables refer to abstract positions in an intermediate level assignment structure, and these refer refer to entities. But that is not how we normally look at it; assignments are just a tool to directly interpret variables.

This last step was exactly the move that that Groenendijk and Stokhof (1991) made. They formulated a notion of dynamic binding, inspired by earlier work on the semantics of computer programs, and by Barwise (1987), that made it possible to consider discourse semantics as a classical (i.e. non-representational) theory, in effect showing that the original theory was not essentially representational in the first place. In this dissertation, I will start with the approach of Groenendijk and Stokhof, and take it from there.

In this section, I will start with discussing the phenomena that are treated by the formalization defined in this thesis. I will borrow a term from the theory of programming languages, and call the connection between an antecedent term and the anaphoric term that gets some of its content from that antecedent a **link** between the two terms (cf. aside 1.1).

Aside 1.1 Index Convention

In language the relation between antecedents and anaphors can take different forms, only some directly similar to the direct value transport that occurs in computer languages illustrated by `x = 4; y = x + 4`. The coindexing notation in the examples will follow the common convention that terms that *bind* an index carry that index as superscript, and those that pick it up as subscript (cf. van Eijck and de Vries 1991). I will not distinguish what the anaphoric expression does with the index. If an expression is coindexed with more than one index, the indices are written with a +sign between them (cf. Baker 1992). Here are some examples of this well known notation:

> *A^1 man entered. He_1 wore a^2 hat. It_2 was green. Following $them_1$ were a^3 man and a^4 woman. $They_{3+4}$ came in an^5 old car. The^6_5 windshield was so dirty that it_6 was almost impossible to look through.*

I will also sometimes use a ≠sign, to denote that an expression is coindexed with an antecedent but different from another antecedent: $10 \neq 11$ means that the term is coindexed with 10, but selects a different referent from 11 (cf. example (h) on page 1.1). The actual semantics of such an expression has to wait for chapter 5, section 5.2.

Formally, the indices are generated by the discourse grammar and denote from what antecedent to what anaphor anaphoric links are to be made. No semantic consequences should be read into this notation other than denoting a source-target notation, a linkage between antecedent and anaphor.

1.1 Linkage

The following example, from Groenendijk and Stokhof (1991), is probably the simplest case of long distance linkage that can be found:

> *A^1 man is walking in the park. He_1 whistles.*

The antecedent expression *a man* introduces a value (the man actually walking in the park) that is picked up by the pronoun *he*. Of this value (the man the first sentence is about) it is claimed that he whistles. In DPL, indefinites are interpreted as existential quantifiers that have scope over everything to the right of it. This makes that we can translate the pronoun *he* as a variable bound by the existential quantifier. In DRT a similar effect is achieved by treating both indefinites and pronouns as free variables and replacing the usual universal closure of predicate logic formulas by an existential closure.

This gives a simple scheme, where the pronoun is identified with an entity or set introduced by an antecedent. However, even this simple example can be modified to illustrate that there are other similar relations between expressions in language. The following eight different examples of the simple case of introducing a referent and picking it up later give a good illustration of the range of possible cases.

a. *A^1 man entered the room. He_1 wore a blue sweater.*

b. *A^2 man entered the room. The_2 man wore a blue sweater.*

c. *A^3 painter entered the room. The_3 artist wore a blue sweater.*

d. *An^4 artist entered the room.$^{??}$ The_4 painter wore a blue sweater.*

e. *A^5 man and a^6 woman entered the room. The^7_{5+6} man wore a blue sweater.*

f. *A^8 married couple entered the room. The_8 man wore a blue sweater.*

g. *A^{9} married couple entered the room.* $^{?}$ *HE^{9} wore a blue sweater.*

h. *$Some^{10}$ children entered the room. $Some^{11}_{10}$ boys wore red hats, $another^{12}_{10\neq 11}$ boy wore a blue cap and two^{13}_{10} girls, who were looking at the window, held $their^{14}_{13}$ hats in $their^{15}_{13}$ hands.*

First compare (a) and (b): *He* and *The man* play similar roles. The predication expressed by the second sentence is over the man introduced in the first sentence. Sentence (c) shows that the nouns involved need not be the same, although the relative awkwardness of (d) suggests that the definite's noun should be more general than the antecedent's noun.

The next two sentences, (e) and (f), show that definites can be used to select parts of earlier sets. In (e) it selects the man from the couple that (we know) consists of one man and one woman. The same happens with (f), though here the validity of the resolution needs the knowledge that there is one man in the couple. Once the identification is made, the referent is picked up without any problem at all. Later we will see that a definite always selects the largest subset within the set that the discourse referent that this definite picks up refers to.

Example (g) is different. Note how much harder it is to use (g). It more or less works with stress on *HE*, but it is not clear how much "repair" of the sentence is involved to interpret it at all.

Finally, example (h) shows that not only pronouns and definites are capable of picking up referents. In fact, all quantifiers are.

All these examples are cases of definites picking up subsets of some earlier introduced set, a subset that is sometimes identical to that set. However, the relation between anaphor and antecedent need not be the subset relation. Often, in the case of definites, the noun expresses a relational property, that can stand in place of the subset:

i. *A^{16} car entered the village. The_{16} driver wore a blue sweater.*

This case is similar to (f). Here too, artificial intelligence methods that use world knowledge are needed (cars usually involve one driver) to derive the referent. And again, once we know the referent, the connection is simple to make. The difference is, that here the referent is not directly available as a set introduced in the discourse, but has to be calculated. Both (i) and the much simpler (j)

j. *A^{17} car entered the village. It_{17} was blue.*

are resolved by the discourse grammar in the same way, using configurational clues. In the case of (i), the definite cannot denote a part (world knowledge!) so another form of link has to be decided on. But this latter decision happens *after* the identification has already been made. The logical content of this, the existence of different links than those that just identify (parts of) extensions of discourse referents, is something we have to face if we want to formalize discourse reference. However, the actual resolution mechanisms that identify what an anaphoric phrase refers back to goes beyond the scope of this thesis (but cf. section 7 of chapter 5. Also cf. van den Berg 1996a).

Because the linguistic nature of the connections between antecedent and the expression referring to it in (a–g) decreases gradually rather than that being a clear cut, the soundness of distinguishing pronouns from definites, or anaphors from other referents may be questioned, as it sometimes is. One could maintain that given these similarities, all referents should be treated the same. The obvious scenario, they conclude, is to follow sound Montagovian tradition and take a worst case approach: treat everything with artificial intelligence methods. I will not go as far as they do, but I *will* propose my personal favorite for a worst case analysis: I will assume that all noun phrases are anaphoric and introduce referents.

As said at the beginning, in this thesis I investigate how far one gets using modifications of the (dynamically) bound variable approach. Obviously, the full content of (i) will stay out of reach to us, because the driver is never introduced as an element of (the interpretation of) a linguistic expression. However, it will turn out to be possible to formalize the logical part of the relation between the two. I will define a method to express that the variable bound by *the driver* depends on the variable bound by *A car*. This method is similar to a method called existential disclosure, developed by Dekker (1992).

1.2 Donkey Sentences

Not all expressions simply throw a set in the pond, to be fished out later by an anaphoric expression. In other cases, the value "moves" with a quantifier. Here's an all time classic to illustrate this:

(1) *Every farmer who owns a donkey beats it.*

This sentence has two readings which, according to Chierchia (1992), can be paraphrased using a modified copying approach:

(1′) *Every farmer who owns a donkey beats* ***a donkey he owns.***

(1″) *Every farmer who owns a donkey beats* ***every donkey he owns.***

Both these reading exist in actually occurring natural language. The first is called the *weak reading*, the other the *strong reading* (the names have to do with one particular formulation in one particular theory of discourse quantification. I will discuss this in chapter 2, section 1.1).

Originally, the weak reading was exemplified by the sentence

(2) *Every driver who has a dime puts it in the meter.*

Because of this sentence, the weak reading is often referred to as the *dime reading* and strong reading is called the *donkey reading*. The latter because donkey-sentences have the strong reading as their most natural reading.

The dime-example suffers what 'counter' examples often suffer: it is not similar enough to the original example it is supposed to be a variation of. For one thing, as was pointed out to me by Kalman (p.c.), the aspect of the sentence is different from that in the original sentence, which might very well be an explanation of the meaning difference. To solve this kind of problem, Chierchia came up with the following scenario to illustrates that the original donkey-sentence is itself ambiguous between the weak reading and the strong reading.

(3) *The farmers of a certain village are very aggressive and beat each other up all the time. After consulting the local psychiatrist, they decide to find other ways of ventilating their surplus of aggression. For example, most farmers who have a donkey beat it.*

This of course is open to a similar rebuttal as the one given to the dime-example. This is not quite the same sentence. For this to work, you have to read the last sentence with stress on *donkey* and *it* (the kind of stress known as *contrastive-stress*). In spoken language, intonation resolves at least part of the ambiguity present in the written language, and intonation is as much part of the linguistic form as anything else, so the mere fact that the *written* sentences are the same in both the weak and the strong scenario is not enough to decide that they are the same.

The exact way a meaning is generated from the sentence involves discourse structural considerations that are beyond the scope of this thesis. The point I would like to make here is a slightly different one. The above analysis in terms of modified copying suggests an E-type analysis of these sentences. Especially in the case of the strong reading, the *it* refers, for every farmer, to all the donkeys which that farmer owns.

Any decisions we make concerning how E-type pronouns get their maximized meaning will have an impact on the weak/strong ambiguity. There are three possible ways of dealing with this phenomenon. First, the maximization may be caused by the antecedent, second, it may be caused by an external operator different from both antecedent and pronoun, third, it may be caused by the pronoun.

If the maximization of the set is due to *the antecedent*, the ambiguity will have to lie in that antecedent: the strong reading corresponds to the case where *a donkey* introduces all donkeys (for a particular farmer) and this set is picked up by the pronoun, and the weak reading corresponds the case where *a donkey* introduces one, possibly non-unique, donkey (for a particular farmer).

In DRT the implementation of the strong reading is explained using the second possibility: *an external operator*. It is assumed that the universal quantifier binds all indefinites in its scope, not only *farmer*, but also *donkey*. This is called *unselective binding*. This of course also achieves the purpose, but is slightly unintuitive. A variation of this approach that is more intuitive is to take the weak reading as primitive, and to say that the strong reading corresponds to demanding that all possible weak readings hold. This is in fact the analysis that DPL uses and one that will also re-occur in chapter 5. Note that if this is the method chosen, then we still have to explain cases where there is no embedding quantifier as is the case for the original example (*John has some sheep. He keeps them in a barn*) in terms of either the antecedent or the pronoun.

This shows that if the antecedent or an outside operator does the work, there are not many problems. However, if the effect is to be caused by *the pronoun*, things are hairier. It is not too difficult to imagine an approach that gets the strong reading to work: just take, for any owner, any donkey of that owner you can find, and predicate a beating relation between the owner and the set of these donkeys. The weak reading however is a lot stranger because it does not maximize. It would constitute a non-E-type pronoun in a position where one would expect an E-type pronoun. In fact, it cannot be the normal, non-E-type pronoun, because that is a bound variable, and the antecedent does not bind that position. The weak pronoun is a whole new type of pronoun, one that picks up only one possible value among the many. And one might expect such a type to be productive and to occur in other places too. However, there is no reason to believe that such pronouns actually occur as, say, subjects of stand-alone sentences in a discourse. It is only in special embeddings that these cases turn up. This means that the meaning-difference must have something to do with the difference in context of interpretation, which puts the burden back on some difference in either the antecedent or some external operator. It will not do to say that these kind of pronouns have a limited distribution and just happen, by convention, to only occur in this special kind of contexts. That would just be a rephrasal of the problem and leaving it at that. Conventions should only be invoked to explain conventional properties of language: the syntactic properties.

The above seems to give enough weight to the position that it is the antecedent that maximizes the discourse referent. Of course, this means is that we really have to buy into a plural theory of reference. If the anaphor did all the work, we might conceivably restrict ourself to singular discourse referents, and let the discourse referent pick up all the possible values for a given referent. But if the antecedent does the work, it has to introduce the maximized set into the discourse.

1.3 Determiners in General

One of the big "inventions" of the dynamic semantics approach is the observation that there is a real difference between indefinite, definite and quantificational determiners. This is most obvious in the original formulations of Kamp and Heim, where *indefinites* introduce a new referent, *definites* pick up an old one, and *Generalized quantifiers* bind all referents introduced in their scope, provided they are not already bound by another quantifier. All referents that are not introduced inside the scope of any quantifier are bound by existential (rather than the more usual universal) closure, effectively making such introductions into existential quantifiers over the whole text. In dynamic predicate logic and its descendants, in particular in the work of Chierchia (1992) and van Eijck and de Vries (1991), *indefinites* are existential quantifiers that bind any variable to the right of them, *definites* are free variables, bound by such quantifiers. *Generalized quantifiers* are another kind of animal altogether: they locally introduce a new referent but hide it from the rest of the discourse, in a so called closed context.

There is something sound in this distinction. There are conceptually different things going on when a new referent, or variable in the dynamic predicate logic jargon, is introduced (indefinites), when it is used in the scope of its introducing determiner (quantification), and when it is picked up as an anaphor (definites). It certainly makes sense to distinguish these different uses to shed light on their respective contributions to the totality of discourse linkage. However, I think it goes too far to say that they are related to three disjoint syntactic (sub-) categories. It is just not the case that natural language has special words to introduce referents, other words to pick up referents, and still other words to quantify over predicates. Suppose this were true, then instead of saying something like

(4) *Some*^{1} *children came to the restaurant yesterday. Three*$_1^2$ *boys were*$_2$ *especially annoying. They*$_2$ *threw food at each other.*

we would have to use a sentence that uses distinct words from these distinct categories: some to pick up an old referent (PRON), some to quantify over a formula (QUANT), and some to introduce a new referent for later (INDEF). This means that the sentence we would have to use would be something like (5), where subscripts denote as what the expressions are meant.

(5) *Some*^{1} *children came to the restaurant yesterday. Some*$^{3}_{\mathrm{INDEF}}$ *boys were three*$^{2}_{\mathrm{QUANT}}$ *boys of them*$^{1}_{\mathrm{PRON}}$ *who*$^{2}_{\mathrm{PRON}}$ *were especially annoying. They*$^{3}_{\mathrm{PRON}}$ *threw food at each other.*

This is obviously not what we normally do. For example, the quantifier $Three^2_1$ in (4) is an expression that picks up a referent (PRON, 1) and introduces a new referent (INDEF, 2). The picked up referent functions as the domain of quantification (QUANT).

My own opinions currently lean towards the other extreme. All noun phrases have something of all three components. All noun phrases introduce a new referent, all noun phrases pick up an old referent, and all noun phrases are quantificational, be it not always a very interesting quantifier (This opinion is shared by several other authors cf. van Deemter (1991), Fernando (1994a) amongst others).

In some cases, it seems to be taking things into the absurd to maintain that a noun-phrase introduces a new referent. For example, would you say that a pronoun introduces a new referent? If you don't think so, what would you say the pronoun *he* picks up in the last sentence of the following example:

(6) *Both Mary and Anne took their husbands to the party. Mary had given him a new jacket. He was really pleased with that.*

It seems clear that *he* refers back to the *him* in the second sentence, and that this *him* has the two husbands as antecedent, but but by being in the scope of *Mary* only picks up Mary's husband. I admit that there are alternative explanations. For example, you might say that Mary's husband is more *salient* than Anne's. But personally I regard explanations in terms of saliency a form of defeatism: It explains everything and therefore does not constitute an explanation at all; it just begs the question. There is of course room for a notion of saliency, but as a defined notion, not a primitive one. A theory that gives such a definition would have to predict for any position in the discourse which antecedent can be co-indexed with an anaphoric element in that position. This is part of the tasks of a discourse grammar, and will have to wait for another place (van den Berg 1996a).

Let us now give some more examples of noun-phrases being antecedents, i.e. introducers of referents, and anaphors, i.e. pickers-up of referents.

1.4 Arbitrary Determiners as Antecedent

Any noun-phrase can serve as the antecedent part of an anaphoric link, and introduces the discourse referent in the same way. Quantificational expressions add to this some independent, "quantified" claim about reality. This quantificational part is fairly independent of the discourse part, it may even be ambiguous, without necessarily influencing the nature of the discourse link.

Here is an example that is very similar to the earlier indefinite examples.

(7) *Three women entered the bar. They immediately started to insult people.*

The noun-phrase *three women* introduces a referent, consisting of three women, and the pronoun *they* picks this up. The way referents are introduced by the numeral and picked up by the pronoun is completely analogous to the way indefinites work. Interestingly enough, this is not dependent on what reading of the numeral quantifiers is chosen. Let us look somewhat closer at this.

Traditionally, numerals are the center of a debate in generalized quantifier theory: does this sentence claim there are exactly three women that entered the bar, or is it enough for the truth of this sentence that there are at least three women? The safest position is to say that the sentence is ambiguous between the two readings. But this poses a problem, because this will make this mini-discourse ambiguous too.What does this ambiguity in the discourse consist of? It is illuminating to look at another mini-discourse:

(8) *At least tree women entered the bar yesterday who are a good example of the horrible people living here. They immediately started to insult people and threaten them.*

In this example, *they* picks up all the women that entered, not just three of them. If one of the readings of (7) is an *at least* reading, is it equivalent to (8)? Maybe a bit surprising, the answer is **no**. Both readings of the numeral in (7), introduce a referent that consists of three elements. The ambiguity lies in the circumstances under which the antecedent sentence is true, not in the referents introduced. It is ambiguity of the quantification, not of the dynamics. In the case of the *at least three* reading of (7), the set of three women introduced is not (necessarily) unique, in the *exactly three* reading it is unique. A better name than the at least reading is the *adjectival reading*. This is how these cases are often analyzed: as an indefinite noun-phrase containing a numeral adjective (cf. Bartsch 1973). My analyses will be similar.

The adjectival meaning seems to be a counterexample to the modified copying (E-type) explanation. The paraphrase:

(9) *Three women entered the bar. Every woman who entered the bar immediately started to insult people.*

is not correct for the adjectival reading. For the exactly three reading, this *is* a correct paraphrase, because in that case *every* quantifies over all, that is, the unique three, women.

Summing up. To the pronoun the reading of the antecedent does not make a difference. It still picks up a set of three women introduced in the antecedent. The discourse effect of the the two readings are identical. the difference lies in the claims they make about reality. The predicative impact of the antecedent is different for the different readings, but the dynamic effects are similar. The adjectival reading introduces a, possibly non-unique, set, the “exactly” reading introduces a unique set, *and* claims its uniqueness.

There is another way in which independent variation between the predicative part and the dynamic contribution can occur. For example, in

(10) *Few women failed the test. They had studied a lot because they wanted the degree.*

(11) *A few women failed the test. They had studied a lot because they wanted the degree.*

the quantifiers in both cases demand that the set satisfying the nuclear scope is smaller than the set not satisfying it. The quantificational impact of the two determiners is identical. The dynamic effects, however, are complementary.

The referent introduced by *few* is the complement of the set involved, whereas *a few* introduces the set itself.

Of course, there is a lot more to be said about these examples. Because *few* and *a few* introduce different referents, they are used in different situations. It is in fact difficult to construct a really good minimal pair. In the above, (10) seems much more natural than (11). In both cases, there are contexts in which the alternate set can also be picked up. This does not invalidate the argument that both readings exist, but does suggest that there is a lot more going on here.

1.5 Arbitrary Determiners as Anaphor

Just as any noun-phrase can introduce a discourse referent and be a bit like an indefinite, it can also pick up a referent and be like a definite. To see how this can be, let us look at some simple examples.

(12) *Most men were wearing a raincoat.*

(13) *Every woman knew the answer.*

(14) *Exactly three dogs ate tripe that day.*

The traditional story when interpreting sentences like these is to say that within the *respective domains* of quantification, the number of men wearing a raincoat was larger than the number not wearing one, the number of women that knew the answer was equal to the number of women in the domain, and the number of dogs eating tripe at the designated day added up to three.

For understanding the mechanics of quantification, this analysis is satisfactory. However, for understanding the functioning of determiners that have quantifiers as (part of) their meaning this will not do. One who realized this was Westerståhl, who in 1989 introduced the notion of a context set, derived from the traditional notion of a restricted quantifier in mathematics. Recently, a marriage between this and the dynamic tradition has been proposed by some authors (van Deemter 1991, van den Berg 1993b).

The idea is that the *respective domains* of quantification should be part of the formal representation (i.e. the meaning) of the sentence. Every quantifier can carry its domain of quantification. This begs the question where the context set originates from. Some clues can be found in the following two examples.

(15) *Some children entered.*

a. *They wore a red coat.*

b. *The children wore a red coat.*

c. *The boys wore a red coat.*

d. *Every boy wore a red coat.*

e. *Exactly three boys wore a red coat.*

f. *Three boys wore a red coat, and behind them was one with a blue sweater.*

The first of these, (15.a), constitutes a normal (E-type) pronoun, and the definite in (15.b) behaves exactly alike. Note how similar (15.c) and (15.d) are. They state that of these children every boy wore a red coat. Like the pronoun, this exhibits what, for lack of a better term, may be called the E-type-effect. Both are about the maximal set of boys that is a subset of the children. The same is true for (15.e). The maximal set of boys is the domain of quantification of the quantifier *exactly three*.

The quantification is restricted by two things: the discourse referent and the noun itself. The result is a subset of the value of the discourse referent that satisfies the noun. The above examples show that this subset is in fact the maximal such subset. The value that the quantification as a whole introduces as a referent is a subset of this subset, usually the set satisfying the nuclear scope. As already mentioned in the previous section in the discussion of *few*, it is sometimes the complement of the set satisfying the nuclear scope. To not make things even more confusing, the latter possibility will be ignored for the moment.

This paints the following picture. Every noun phrase is interpreted relative to a context set C, which is the value of a discourse referent. In the actual quantification, two sets, derived from this context set, are involved. First, there is the set $R \subseteq C$ satisfying the quantifier restriction. Second, there is a subset N of R satisfying the nuclear scope. This N is the value of the new discourse referent introduced by the noun-phrase; R is also introduced as a discourse referent. It is a discourse referent that is a maximal set (as subset of the context set) satisfying the noun. This set is the value of any CN-anaphor referring back to this noun-phrase. Example (15.f) illustrates this point. The first conjunct (*three boys wore a red coat*) introduces a referent (N), picked up by *them* in the second conjunct, referring to the three boys wearing a red coat. It also introduce a referent (R) that refers to all boys *among the children*, which is picked up by *one* in the second conjunct.

Note the similarity with programming languages. A definition like `int i = 3` introduces a variable `i` with a value (3) and a type (integers). The value can be picked, as in `j = i`, which corresponds with normal anaphora. But there is also a counterpart of *one*, which takes the form `malloc(size_of(i))`, the operation that makes a new variable of the same type as the old variable.

Definites demand that their nuclear scope holds for the whole subset: they demand identity of the two subsets involved. Pronouns are definites that, having no restriction, do not restrict the subset, so in that case, the maximal subset is the set itself, and both subsets are identical to the context set. Thus pronouns *are* special cases of definites and definites *are* special cases of generalized quantifiers. This also shows why normally it is a reasonable simplification to treat pronouns as free variables, the context set is just copied, there is no real reason to give it a new name. In special cases, particular embeddings of the pronoun may cause the introduced referent to be non-identical to the referent that is picked up, resulting in a recognizable difference in semantics between the old and new referents (cf. the discussion of example (6)).

The above is a simplified account. An important complication involves functional dependencies:

(16) *Every man had to have an operation on one of his legs. They broke them while skiing.*

(17) *Every girl gave every boy a present. She put it on his desk.*

In example (16), the domain of quantification of *one of his legs* covaries with the quantifier *every*: for every man there is a different domain of quantification (the legs of that man) and a different leg introduced. This relationship is picked up again in the second sentence, making sure that every man broke his own leg rather than that of someone else. Example (17) makes a similar point. Obviously, every girl put her own present on the boys desks, not that of someone else. This shows that discourse referents do not have "naked" sets as values, but also contain information on how the elements of the sets they *do* denote relate to the elements of other sets.

Another complication, already mentioned above, is that the domain of quantification and the new discourse referents need not be subsets of the context set:

Seventeen cars entered the village. Most drivers looked tired.

In this case, the sets involved are the drivers of the cars and the subset of drivers that are tired. Despite the more indirect relation, the maximization effect still occurs. The first set has to be the set of drivers of the seventeen village-entering cars, not just some of them, it is still maximized, like any old E-type definite.

1.6 Overview

The following gives an overview of the most important phenomena involved in linkage.

Classes of Links

A number of possible ways of linking an antecedent with its consequent can occur. In the first place, there are **identity links**, the links that most of the dynamic semantics literature concentrates on:

(18) *Some*^{1} *children entered the bar. They*$_1$ *sat down.*

the second possible case is the **subset link** that where the consequent is a subset of the antecedent:

(19) *Some*^{1} *children entered the bar. The*$_1$ *boys sat down at the bar and the*$_1$ *girls walked to the jukebox.*

Beside these, there are two other possibilities. One is the case of **sum link**, mentioned by Kamp and Reyle (1993), where more than one antecedent binds one consequent:

(20) *A*^{1} *man sat down and a*^{2} *woman walked to the jukebox. They*$_{1+2}$ *were here to celebrate his*$_1$ *birthday.*

This is very close to being a dual to the subset case. In the subset case, a set is split up into several subsets, here several sets are combined in one superset.

A last possibility is the combination of *sum link* with *subset link*:

(21) *When a^1 student of the physics department marries a^2 student of the mathematics department, the$^3_{1+2}$ man usually finishes his$_3$ studies before the$^4_{1+2}$ woman does, but she$_4$ gets better notes.*

Taking subsets even further, this amounts to (parallel) subsets on both sides of the link.

In all these cases, the fact that there are two subsets is of course accidental. These four cases are the only possible ones: one-one, one-many, many-one and many-many, and it seems all possible cases are realized in language. Sum cases are not much discussed in the next chapters, because they do not constitute additional interesting problems, being the union of some discourse referents. I will have some small things to say about them in section 5.2 of chapter 5, however.

Internal Structure of Referents

Noun-phrases can be singular or plural. This has some consequences for the way discourse referents are picked up. Definites that pick up a discourse referent that contains more than one element can be one of two things. They can be plural, and **agree with the semantic number** in which case they pick up the set as a whole and do their thing with it, or they can **agree with the syntactic number** of the antecedent that introduced the referent:

(22) *Every^1 man loves a^2 woman. They$_1$ bring them$_2$ flowers to prove it.*

(23) *Every^1 woman gave every^2 man a^3 present. She$_1$ put it$_3$ on his$_2$ desk.*

There will only be a difference between syntactic and semantic number agreement in the case the antecedent is syntactically singular but semantically plural (as for *every*). Syntactic number agreement appear to be subject to much stronger constraints than semantic agreement. Rules that determine accessibility of this kind are lied down in the discourse grammar, and beyond the scope of this dissertation.

In chapter 3 we will see that quantifiers are always ambiguous between distributive (one entity at a time) and collective (all entities at once) readings. For a singular definite to pick up a whole set, it has to be interpreted as a distributive expression, because the alternative would demand the antecedent to be a singular entity, which it isn't. In (23) *she* quantifies over all woman one at a time.

If a referent is introduced in a distributive context, the introduced object will be **functionally dependent** on the set that the distribution is over:

(24) *Every^1 woman bought some^2 friends some^3 presents. She$_1$ put them$_3$ on their$_2$ desks.*

Here the distribution over the discourse referent (^{1}) makes that the discourse referents (^{2},^{3}) which are functionally dependent on it are also recreated the right way: *them$_3$* refers to the respective presents, that the respective women give, and *their$_2$* refers to the corresponding girl-friends.

Aside 1.2 Other Cases than Noun-phrases

In this dissertation I concentrate on noun-phrases, because that is already hard enough. However, other discourse linking behaves in a surprisingly similar way. For example, the maximization (E-type) effect also occurs when modal contexts are linked, as this example from Roberts (1983) shows

A^1 thief might^2 come in. He$_1$ would$_2$ steal the furniture.

Where *might* is the "indefinite" introducing a set of worlds (those worlds in which thieves enter) and *would* is the "definite" picking up those world in an E-type fashion (this was observed and worked out in a slightly different, but similar way, by Kibble (1994)). The notion of functional dependency discussed above also applies here: *might* is a distributive quantifier over worlds, and in every world, a thief is introduced. As in the case *every*, there will be a functional dependence between worlds and thieves in them. In the second sentence, *he* is then interpreted in the scope of the world-definite, which is again distributive, giving the right thief in the right world. This shows that this perspective on functional dependencies gives a method of localizing an entity in a world as a side-effect of a more general mechanism. Whether this really gives new insights into the relation between entities and (partial) worlds is something that requires further research.

2 Dynamic Semantics

I will now give a short introduction to dynamic semantics and other logical tools needed in chapter 2. I start with a small historical note, explaining how dynamic logic came into the picture because of the interest in the semantics of programming languages. The second part shows how independently, there seems to be a demand in natural language semantics for an incremental logic not unlike the dynamic logic of the computer science people. I then define DPL, the logic of Groenendijk and Stokhof (1991) and discuss some of its properties. I end with a section on partial logic, which is the other logical tool needed in chapter 2.

2.1 The Interpretation of Programs

The origins of dynamic logic lie in the semantics of imperative programming languages. A simple C-program fragment like the following

```
x = 3;
y = x + 2;
x = y - 3;
```

has an obvious "meaning". First, the code for the value 3 is put into the location denoted by `x`. Then the content of the location denoted by `x` is retrieved, 2 is added to it, and it is put into the location denoted by `y`. Then the value is retrieved from the location denoted by `y`, 3 is subtracted from it, and the value is put into the location denoted by `x`.

To express the properties of this semantics, algebraic systems (Bergstra and Klop 1986), multi-modal logics (cf. Harel 1981, van Benthem et al. 1993) and type-logical systems (Janssen 1983 were developed.

But computer science was not the only place where people thought about effects from one formula into the next. In linguistics and philosophy of language, similar considerations can be found.

2.2 The Dynamic Interpretation of Text

Text is read (or heard) in a sequential way from beginning to end, and it is interpreted in this same sequential way by interpreting the units of the discourse one at a time. At any moment during this process some initial segment of the text is already interpreted. This initial segment is part of the context at that moment. No sentence of a text is interpreted in a vacuum, it is always interpreted in a context to which the sentences preceding it have contributed. Consider for example *He sees her run* in the following text:

(25) *There is a woman in the park. She is jogging. A man is sitting on a bench. He sees her run. A girl is sitting on a bench opposite him. She is wearing a red sweater. I also see some boys. They are taking their dogs for a walk. They are in a hurry because they want to be home with them before dark.*

The discourse preceding it gives us the information that *He* gets its interpretation from *a man* and *her* from *a woman*. I take it that the interpretation of a sentence takes place in two stages. The first stage I call the coreferencing of the sentence, the second is the conjunction of the resulting translation of the sentence with the context. In the first stage suitable indices are given to all anaphors and quantifiers in the sentence. This process takes place in tandem with the translation of the sentence into a formal language and is part of this translation. The end result of this translation has to result in a defined discourse to be understandable. But it need not be true. We understand factual description in the same manner as fairy tales. What matters for resolution is that all links result in interpretable (defined meanings), not that they result in meanings that evaluate to true.

That the accidental truth is not the decisive factor in determining the indices is also illustrated by (25). In the sentence *She is wearing a red sweater* the pronoun *She* can only be coindexed with the girl sitting on the bench, even if that girl is wearing a blue t-shirt whereas the running woman is wearing a red sweater. The most important thing to realize is that *truth is at best a secondary issue* for discourse resolution.

In the second stage, the translation of a sentence resulting from the first stage is combined with the translation of the preceding discourse, using the appropriate relation, which gives an interpretation of the discourse that results from incorporating that sentence. It is here where *dynamics* comes in. The new sentence, obviously, does not occur in the scope of any expression in the preceding discourse but still has to derive some of its meaning from expressions in that preceding discourse. *Dynamic semantics* formalizes the way in which expressions in one formula can give values to variables in another, to achieve cross-sentential binding, even if the variable is not syntactically in the scope of that expression.

There are a number of theories around that achieve the aims of dynamic semantics in different ways. Dynamic logic, the formalism used in this thesis, implements it be redefining the interpretation of logical formulas, DRT achieves a similar effect by postponing binding of free-variables until certain check-points are reached, either an embedding quantifier, a negation or the end of the text (existential closure). Even formalisms that reconstruct the logical expression to move the variables to be in the scope of the binding expression may be considered implementations of dynamic semantics.

$$S_A \xrightarrow[\text{add referent 1}]{A^1 \textit{ woman is jogging.}} S_B \xrightarrow[\text{add 2, use 1}]{A^2 \textit{ man sees her}_1 \textit{ run.}} S_C$$

Figure 1.1: Sentential Conjunction

That part of the context that is relevant for the value assignment to anaphors in any possible continuation of a discourse is called the *(binding-) state* of the discourse at that point. It is finding a sufficiently rich notion of state, which is not so rich as to become trivial, that takes most of the work. States should be defined in such a way that they give us sufficient information to determine the binding of any pronoun or other anaphor that could occur in the next discourse unit. In fact, most properties of a dynamic logic are fixed as soon as an appropriate choice has been made of the notion of state that we are going to use. This holds in particular for the definition of *conjunction* of two formulas of dynamic logic. Suppose at a given point in a discourse we are in state S_A (fig.1.1). If we now interpret the next unit, incorporating the new information added by this unit, the state changes to S_B. This means that we can regard discourse units as functions from states to states: given an old state as input, the discourse unit gives us a new output state. If we want to take this idea seriously, we are committed to interpret conjunction as function composition:

The interpretation of the concatenation of two sentences s.t is done by first interpreting s in the old state (S_A), followed by the interpretation of t in the resulting new state (S_B) to give the final state (S_C). Giving a dynamic logic in our sense will always consist of:

1. giving its syntax.
2. giving the states.
3. giving the semantics as action of formulas on these states, where conjunction is interpreted as composition.

In this dissertation, *the syntax* of the language will be a variant of standard predicate logic, with some simple extensions to cater for plurals. As you might guess, the interpretation will not be standard at all.

What to take as the *states* and the *actions* on them is a different matter. Groenendijk and Stokhof take assignments as their states, and so will I, at least to begin with. Later on I will complicate matters considerably by taking sets of assignments as states, to keep up with the facts. I will leave the exact discussion of the alternatives to the next chapters, and only discuss Groenendijk and Stokhof here.

2.3 Groenendijk and Stokhof

Groenendijk and Stokhof defined a logic, called *dynamic predicate logic* (DPL), to explain the discourse anaphoric phenomena treated by Kamp (1981) in DRT, by interpreting formulas as relations between assignments. In DPL, *discourse referents* are introduced into the discourse as variables. Under specific conditions, formulas further to the right are then able to pick up that values of these variables from the discourse. every variable. In other words: states are assignments. The definition of DPL is as follows:

2.1 Definition (DPL)
The syntax *of* DPL *is that of ordinary predicate logic with identity. However, the predicate logic used does not have constants. The* semantics *of* DPL *interprets formulas as relations between assignments.*
The interpretation $[\![\phi]\!]^{\mathcal{M}}$ *of a formula* ϕ *as a relation between states is defined with respect to a standard first order model* $\mathcal{M} = \langle D, \mathcal{I} \rangle$, *where* D *is a domain of entities and* $\mathcal{I}$ *a function that assigns to any n-place predicate* a set *of n-tuples in the standard way. I will suppress the model* $\mathcal{M}$ *where possible;* $[\![\phi]\!]$ *is characterized as follows:*

(a) $g[\![Px_1 \ldots x_n]\!]h$ iff $g = h$ and $<g(x_1), \ldots, g(x_n)> \in \mathcal{I}(P)$

(b) $g[\![x = y]\!]h$ iff $g = h$ and $g(x) = g(y)$

(c) $g[\![\phi \wedge \psi]\!]h$ iff $\exists k\,(g[\![\phi]\!]k$ and $k[\![\psi]\!]h)$

(d) $g[\![\exists x \phi]\!]h$ iff $\exists k\,(g \approx_x k \wedge k[\![\phi]\!]h)$

(e) $g[\![\sim\phi]\!]h$ iff $g = h$ and $\neg\,\exists k\,(g[\![\phi]\!]k)$

where $g \approx_x k$ *means that* $g(z) = k(z)$ *for all variables* z *not equal to* x.

A formula is interpreted as a set of transitions that transform an old information state into a new information state. Dynamic logics distinguish between *dynamic conditions* and *truth conditions*. The notions of true and false play a role on two levels. First of all, formulas are interpreted as relations between an input and an output. It makes sense to say, that such a relation holds between an input and an output, or equivalently that it is true for a given input and output. But a formula also expresses a property of the model, and it makes sense to ask whether the formula is true (or false) in a given context. This makes some of the discussions a bit confusing and it makes sense to carefully distinguish the two. The conditions that determine, whether a formula holds for a given input and output, will be called the **dynamic conditions** of the formula. That conditions that determine whether a formula is true in the model, given a context (determined by an input state), will be called the **truth conditions** of the formula. Of course, the two will not be independent: a formula is true in a given state if there is at least one possible output for that state.

(26) $\|\phi\|^g = \mathbf{true}$ if $\exists k\;\, g[\![\phi]\!]k$.

If no such transition exists, the formula is said to be false:

$$\|\phi\|^g = \mathbf{false} \text{ if } \neg\,\exists k\;\, g[\![\phi]\!]k.$$

Let me now return to the definition of DPL. First note that there are no constants in this language, only predicates and variables. Proper names are interpreted as appropriate predicates. This has a technical reason. Referents are introduced by the existential quantifier, restricted by a predicate. If a proper name was interpreted as a constant, it would not introduce a referent, and it would therefore be impossible to refer back to anything introduced by a proper name.

I will now go through the definitions one at a time, and see which transitions they define.

In definitions (a) and (b), *atomic formulas* are interpreted as tests. They are filters that let some states “pass” and others not. Which states a formula accepts depend on the static meaning of the formula. An atomic formula is interpreted *in* a state, it does not change it. The transitions it defines map a state onto that same state, provided the static (i.e. standard) predicate logic interpretation of the formula holds in that state:

(a) $g \xrightarrow{Px} g$ iff $g(x) \in \mathcal{I}(P)$

Let me now introduce a distinction of my own, that will be useful later. Note that to express the interpretation of a predicate as a test, the definition of $g[\![Px_1 \ldots x_n]\!]h$ consists of two parts.

First there is the condition $g = h$, characterizing the transitions which might conceivably hold. This will be called the **dynamic condition**, because it expresses what inputs are connected to what outputs. Because predicates do not change the state but only test it, there are no transitions where the output state is different from the input state.

Secondly the condition $<g(x_1), \ldots, g(x_n)> \in \mathcal{I}(P)$ defines the transitions for the assignments for which the predicate is actually true. This will be called the **static condition**, because it has the form of a condition on the input state that does not involve any output state. Transitions for which a formula is actually true will be called the true transitions.

Note that both conditions are needed, the first expresses which transitions there are, the second which of these are true. In the transition diagram (a) the first condition is expressed by the arrow going from g into g, and the second by the constraint written to the right. In DPL, both transitions for which the static condition is false, and input-output pairs that do not satisfy the dynamic conditions, result in the relation being false: Formulas are interpreted as two-valued relations. In chapter 2, I will exploit the distinction between failure of the dynamic condition and failure of the static condition to great effect.

Definition (c) defines conjunction as relational composition. The transitions corresponding to a conjunction $\phi \wedge \psi$ consists of first a ϕ transition followed by a a ψ transition:

(c) $g \xrightarrow{\phi\wedge\psi} h$ iff $\exists k\ g \xrightarrow{\phi} k \xrightarrow{\psi} h$

An important property of the conjunction is its associativity:

(27) $g[\![(\phi \wedge \psi) \wedge \chi]\!]h$ iff $g[\![\phi \wedge (\psi \wedge \chi)]\!]h$,

which follows directly from the associativity of relational composition[1]. Which combines with the definition of existential quantification to give DPL its particular dynamic binding properties (cf. 28 below).

1. $g[\![(\phi \wedge \psi) \wedge \chi]\!]h$ iff $\exists l\ (g[\![\phi \wedge \psi]\!]l$ and $l[\![\chi]\!]h)$ iff $(\exists k, l\ g[\![\phi]\!]k$ and $k[\![\psi]\!]l$ and $l[\![\chi]\!]h$ iff $\exists k\ (g[\![\phi]\!]k$ and $k[\![\psi \wedge \chi]\!]h$ iff $g[\![\phi \wedge (\psi \wedge \chi)]\!]h$.

The meaning of the existential quantifier as defined in definition (d) is the core notion of dynamic predicate logic. The existential quantifier is often described as *introducing the variable* rather than binding it because a variable bound by an existential quantifier is bound not only in the scope of the existential quantifier, but also in any formulas to the right of the quantification (but cf. (e) to see that there are exceptions to this rule).

As a transition, the definition has the form

(d) $g \xrightarrow[\oplus]{\exists x\ \phi} h$ iff $\exists k\ \ g \approx_x k\ \&\ k \overset{\phi}{\underset{\oplus}{\to}} h$

This definition suggests an interesting simplification. The first conjunct at the right-hand side, $g \approx_x k$, can be interpreted as a transition allowing us to give an independent dynamic meaning to the existential quantifier.

(d′) $g[\![\varepsilon_x]\!]h$ iff $g \approx_x h$

Traditionally, this is called the *random assignment*.

We can now define $[\![\exists x\phi]\!]$ as the conjunction of two transitions: the first of which does the actual introduction, while the second applies the formula to the resulting state.

(d″) $g \xrightarrow[\oplus]{\exists x\ \phi} h$ iff $\exists k\ \ g \overset{\varepsilon_x}{\underset{\oplus}{\to}} k \overset{\phi}{\underset{\oplus}{\to}} h$

This formulation is very useful for proofs concerning the existential quantifier.

One property of the existential quantifier that can be seen directly from the reformulation in terms of ε_x is the dynamic binding potential of dynamic existential quantifiers.

(28) $g[\![(\varepsilon_x \wedge \phi) \wedge \psi]\!]h$ iff $g[\![\varepsilon_x \wedge (\phi \wedge \psi)]\!]h$

This property follows directly from the associativity of conjunction.

Definition (e) defines the DPL negation. The meaning of a negated formula is defined always to be a test. It consists of a *dynamic condition* of the form $g = h$, identical to the dynamic condition in the case of atomic states, and a *static condition* that in the state g, the formula that is being negated should not hold. The negation of a formula, $g[\![\sim\phi]\!]h$, only holds between input state g and output state h such that $h = g$ if there is no k for which $g[\![\phi]\!]k$ holds.

We express the fact that negation turns every formula into a test by saying that negation *closes off* the formula.

In the transition view, there exists a $\sim\phi$ transition from an input state g to the output state g if there is no ϕ transition from g to any output state k. The negation $\sim\phi$ functions as a universal quantification over transitions, stating that no ϕ transition exists.

$$g \overset{\sim\phi}{\underset{\oplus}{\to}} h \ \text{ iff } g = h\ \&\ \neg\,\exists k\ \ g \overset{\phi}{\underset{\oplus}{\to}} k$$

Negation in dynamic logic is in fact a difficult issue. A large part of chapter 2 is dedicated to giving an alternative to the definition of negation. However, this definition is the one that is suggested by the truth conditions, defined in (26), and discussed in section 2.4.

The fact that negation closes off a formula makes it possible to define an important operator, called the **closure** operator, that effectively destroys the binding potential of a formula:

$$!(\phi) := \sim\sim(\phi)$$

Which has the interpretation

$$g[\![!(\phi)]\!]h \text{ iff } g = h \ \& \ \exists k \ \ g[\![\phi]\!]k$$

Note, if you want, that this operator has the form of an existential modality.

A quantifier within a formula that is not embedded under a negation (in that formula) is called an **active** quantifier (in ϕ). Only if a quantifier is not in the scope of a negation can it contribute to the output-state ϕ.

The definitions of disjunction, implication and universal quantifier in terms of the other operators follow the familiar pattern of predicate logic. However, actually choosing a form for the definition is not simple. Groenendijk and Stokhof decided on the following forms:

$$\phi \vee \psi := \sim(\sim\phi \wedge \sim\psi)$$

$$\phi \rightarrow \psi := \sim(\phi \wedge \sim\psi)$$

$$\forall x\phi := \sim(\exists x.\sim\phi)$$

These share an important property: The outer operator in all three cases is a negation. The effect of this is that the output of formulas of the form $\phi \vee \psi$, $\phi \rightarrow \psi$ and $\forall x.\phi$ will always be identical to the input. No dynamic quantifier inside the ϕ and ψ can bind variables following these formulas. This closing-off property of $\vee$, $\rightarrow$ and $\forall$ is expressed by saying that they are **externally static**.

However, inside ϕ and ψ there may well be dynamic quantification taking place. Furthermore note that the definition of the implication is such that it allows active quantifiers inside ϕ to bind variables within $\sim\psi$. The possibility that quantifiers within the antecedent of the implication can bind variables in the consequence of the implication is expressed by saying that $\rightarrow$ is **internally dynamic**. And so is $\forall$, but not $\vee$.

An important property of dynamic expressions is that *if the actual output state does not matter, we can throw it away*. One reason why an output might not matter, is that that output is the output of a formula directly under a negation. This leads to the following fact:

2.2 Fact[2]
$g[\![\sim(\phi \wedge \psi)]\!]h$ and $g[\![\sim(\phi \wedge \sim\sim\psi)]\!]h$ are equivalent.

A direct consequence of this is that

2.3 Corollary
$(\exists x \phi) \rightarrow \psi$ and $\forall x(\phi \rightarrow \psi)$ are equivalent.[3]

This can be used to explain the meaning of the example

If a farmer owns a donkey, he beats it.

This translates into DPL as:

$$(\exists x \ \mathbf{f}(x) \wedge \exists y \ \mathbf{d}(y) \wedge \mathbf{o}(x,y)) \rightarrow \mathbf{b}(x,y)$$

which by the above corollary is equivalent to

$$\forall x \ \forall y \ ((\mathbf{f}(x) \wedge \mathbf{d}(y) \wedge \mathbf{o}(x,y)) \rightarrow \mathbf{b}(x,y))$$

There is one more interesting property of the conjunction that is useful later on. Although the definition of conjunction contains an existential quantifier, this does not mean that there is always a choice. To illustrate this, I first need to define some auxiliary notions. The first of these is a notion $\mathbf{bvar}(\phi)$. This is the set of variables that are bound by an active random assignment inside ϕ. Where an active random assignment is one that is that is not in he scope of a negation operator ($\sim$), which disables the binding force of that random assignment. I will refer to these variables as the variables bound *by* the formula ϕ.

2.4 Definition (Bound Variables and Safe Conjunction)
The set of variables bound by *a formula ϕ, denoted with $\mathbf{bvar}(\phi)$, is defined as follows:*

(a) $\mathbf{bvar}(P\vec{x}) = \mathbf{bvar}(x = y) = \mathbf{bvar}(\sim\phi) = \emptyset$

(b) $\mathbf{bvar}(\phi \wedge \psi) = \mathbf{bvar}(\phi) \cup \mathbf{bvar}(\psi)$

(c) $\mathbf{bvar}(\varepsilon_x) = \{x\}$

If $\mathbf{bvar}(\phi) \cap \mathbf{bvar}(\psi) = \emptyset$, ϕ and ψ are said to be safe for each other. In that case, $\phi \wedge \psi$ is said to be a safe conjunction.

---

2. There is an identity from relational algebra: ($\sim(A \circ B) = \sim(A \circ \sim\sim B)$): (Negations are tests, so we only need to consider an output equal to the input) $g[\![\sim(\phi \wedge \psi)]\!]g$ iff $\neg \exists k \ g[\![\phi \wedge \psi]\!]k$ iff $\neg \exists k, l \ (g[\![\phi]\!]l \& l[\![\psi]\!]k)$ iff $\neg \exists l \ (g[\![\phi]\!]l \& \exists k \ l[\![\psi]\!]k)$ iff $\neg \exists l \ (g[\![\phi]\!]l \& l[\![!\psi]\!]l)$ iff $g[\![\sim(\phi \wedge \sim\sim\psi)]\!]g$.

3. The proof uses transitivity of $\wedge$ and the above fact 2.2: $(\exists x \ \phi) \rightarrow \psi$ is equal to (def. $\exists$) $(\varepsilon_x \wedge \phi) \rightarrow \psi$ is equal to (def. $\rightarrow$) $\sim((\varepsilon_x \wedge \phi) \wedge \sim\psi)$ is equal to (transitivity $\wedge$) $\sim(\varepsilon_x \wedge (\phi \wedge \sim\psi))$ is equal to (from fact 2.2) $\sim(\varepsilon_x \wedge \sim\sim(\phi \wedge \sim\psi))$ is equal to (def. $\rightarrow$) $\sim(\varepsilon_x \wedge \sim(\phi \rightarrow \psi))$ is equal to (def. $\exists$) $\sim(\exists x \ \sim(\phi \rightarrow \psi))$ is equal to (def. $\forall$) $(\forall x \ (\phi \rightarrow \psi))$.

The notion of a safe conjunction is important, because normally, formulas that result from a discourse grammar will satisfy this property. It would be silly to use a variable twice in translating quantifiers, since it is easily avoided. Safe conjunctions have the useful property that the existential quantifier over states used to define the conjunction is vacuous: if the sets of variables bound by the two conjuncts do not overlap, there is only one possible choice for the intermediate state. To show this, let me first give a definition of this intermediate state.

2.5 Definition (mediating assignment)
Let g and h be assignments, ϕ a formula, then we define

$$g_h^{\phi}(x) := \begin{cases} h(x) & x \in \mathbf{bvar}(\phi) \\ g(x) & x \notin \mathbf{bvar}(\phi) \end{cases}$$

I can now formulate the following lemma

2.6 Lemma[4]
If ϕ and ψ are safe, then $\exists k \;\; g[\![\phi]\!]k \;\&\; k[\![\psi]\!]h$ iff $g[\![\phi]\!]g_h^{\phi} \;\&\; g_h^{\phi}[\![\psi]\!]h$.

This allows us to ignore the existential quantifier in the definition of conjunctions when these are safe, an assumption I will always make.

2.4 Truth Conditions of DPL Formulas

Let us now return to the truth conditions of DPL formulas, as given in 26 and repeated here:

$$\|\phi\|^g = \mathbf{true} \text{ if } \exists k \;\; g[\![\phi]\!]k.$$

$$\|\phi\|^g = \mathbf{false} \text{ if} \neg\exists k \;\; g[\![\phi]\!]k.$$

This definition is not just an arbitrary one, it is the only reasonable form. First, note that negation is defined in a way that mirrors this definition: $\|{\sim}\phi\|^g = \mathbf{true}$ exactly if $\|\phi\|^g = \mathbf{false}$.

In fact, the relation between DPL-operators and their PL counterparts goes much further. There exists a canonical mapping ^{N} from DPL into PL, that preserves these truth-values. This map factors into two parts, first a formula ϕ of DPL is mapped onto a possibly different formula ϕ^N of DPL, such that $[\![\phi]\!]$ and $[\![\phi^N]\!]$ are the same relation. Then that formula is interpreted as a PL formula (DPL and PL are syntactically identical, provided we use the form of DPL with the existential quantifier $\exists x$ and not the form with the random assignment ε_x).

Groenendijk and Stokhof call ϕ^N the *normal binding form* of ϕ. The definition of the translation is as follows.

---

4. Suppose for k, $g[\![\phi]\!]k \;\&\; k[\![\psi]\!]h$. then k will differ for some variables from g, and all of the variables it differs in are changed inside ϕ by a random assignment. None of these variables will be assigned values by ψ, because by assumption, all variables changed inside ϕ are left alone by ψ. So for $x \in \mathbf{bvar}(\phi)$ we have $k(x) = h(x)$ and for all other variables, $k(x) = g(x)$. Hence $k = g_h^{\phi}$.

2.7 Definition

*The **normal binding form** of a formula ϕ is defined as follows:*

(a) $(Px_1 \ldots x_n)^N := Px_1 \ldots, x_n$

(b) $(\sim\phi)^N := \sim(\phi)^N$

(c) $(\exists x\ \phi)^N := \exists x\ (\phi)^N$

(d) $$\begin{aligned} (Px_1 \ldots x_n \wedge \phi)^N &:= (Px_1 \ldots x_n)^N \wedge \phi^N \\ (\sim\phi \wedge \psi)^N &:= \sim(\phi^N) \wedge \psi^N \\ (\exists x\ \phi \wedge \psi)^N &:= \exists x\ (\phi \wedge \psi)^N \\ ((\phi \wedge \psi) \wedge \chi)^N &:= (\phi \wedge (\psi \wedge \chi))^N \end{aligned}$$

This transformation preserves dynamic properties: $[\![\phi]\!]$ and $[\![\phi^N]\!]$ are the same relation between input and output[5]. Furthermore, $\|\phi\|^g = \mathbf{true}$, exactly if $\|\phi\|^g$ is true as a formula of standard predicate logic.

The above gives an answer to the question which formula of predicate logic corresponds to a given formula of dynamic predicate logic: given ϕ, the predicate logic formula with the same truth conditions is ϕ^N. This question can also be turned around: given an operator in standard predicate logic, we can ask which operator(s) in dynamic logic correspond to it. It turns out, that there is often more than one candidate in dynamic logic as counterpart for an operator in static logic, because a formula of dynamic logic has both a truth-value and a dynamic effect. Dynamic predicate logic is sensitive to more detail than standard predicate logic is.

One possible source of the proliferation of counterparts of PL in DPL is the choice we have in making dynamic effects carry over from one of the component formulas to the other (internal dynamics) or not. For example, $\phi \wedge \psi$ and $!(\phi \wedge \psi)$ will always have the same truth value. But only the first may have a dynamic effect, whereas the second is a test. Because the meaning of formulas from static logic will not rely on dynamic effects, a closure operator can in fact be inserted anywhere in the formula. So $(!\phi \wedge \psi)$, $!(!\phi \wedge \psi)$, $(\phi \wedge !\psi)$, etc. are all possible counterparts of the conjunction in standard predicate logic.

Another source of multiplication of operators is the fact that DPL exhibits an order dependency. Earlier formulas can bind variables in later formulas, but not the other way around. This means that asymmetric operators will split in two dynamic counterparts, one for each direction. For example, the implication will correspond to an internally dynamic implication $\rightarrow$ and an internally static reverse implication $\leftarrow$:

$$\phi \leftarrow \psi := \sim(\sim\phi \wedge \psi)$$

5. The only non-obvious part may be the conjunction (d). Note the sub-recursion in the left conjunct to get the existential quantifiers to have scope over the whole conjunction to the right.

This is of course only in part a real reverse of the implication, because it is not dynamic. Whereas in the natural language expression *a student can solve the problem, if she wants to*, the referent introduced by the noun-phrase *a student* can be picked up by *she*, variables introduced inside ϕ cannot bind variables that occur inside ψ.

2.5 Partial Logic

It will turn out, in chapter 2, that in order to improve on the above definitions, we need a particular kind of three-valued logic. The reason is that while some outputs may make a formula true and other outputs make the formula false, not all outputs need be appropriate. Consider again the same, simple example A^1 *man entered the room*. Output assignments that do not assign a man to referent 1 are plainly irrelevant. Whether true or false, this is about men. Even more pertinent, consider an output that assigns another value to x_2 than the input does. Such a unexpected change in the value of a not mentioned variable seems inappropriate. As I will argue in chapter two, it makes sense to distinguish this from outputs that *are* appropriate, and make the relation that interprets the expression be true or false. To achieve this, the logic will be partialized.

Extending logic to more than two truth values is a well discussed move. There are two ways to go about this. First, we can take the algebraic approach. This means that we add one (or more) values to the range that propositions take, and add operators to regain functional completeness. Second, there is the semantic approach. Which means we add a third value, with a specific, intended meaning. In that case, functional completeness is not what we want, we only want operators that fit the intended meaning of the third value. In particular: what are the operators that correspond to the standard logical operators? Obviously, the second case will always result in a subset of the operators of the first case, which is, after all, functionally complete.

For the algebraic approach, cf. for example Muskens's dissertation (1996). For the semantic approach, there are at least two reasons to add truth values. First, there may be reason to add a value (which I will write as ?) to express a *lack of information*: a value that expresses that we do not know whether a given proposition is true or false. This is used to good effect by van Eijck (1991b) to formalize a theory of generalized quantifiers where set membership may not be totally known. Second, an extra value (which I will write as $\star$) can be added to express that true or false *does not apply* in a certain case. How the logical operators extend to an added value is different for these two cases. Let us look at them one at a time.

The first question is: suppose ? is one of the conjuncts of a conjunction. If the other conjunct is also ?, the answer is simple: if I do not know what one truth value is *and* I do not know what the other truth value is, then the result will still be unknown. If the other conjunct is false, the result is also known: if I do not know what the truth value is *and* I know that other conjunct is false, then the conjunction will be false. The only problem occurs in the case that the other conjunct is true. In that case, everything depends on the unknown value (?), hence the result is also unknown. This is called the strong Kleene conjunction (29a). For disjunction it works the other way around, if one argument is true, it does not matter that the other is unknown, the result will be true. However, if it is false, an unknown other value will leave us in the dark about the net result: it is still unknown (29b). If ? is negated, we know as much as we started with: nothing. The resulting negation is called the weak negation (29c). This leads to the following logical operators:

(29)

a)

$\wedge^s$	$\top$	$\bot$	?
$\top$	$\top$	$\bot$	?
$\bot$	$\bot$	$\bot$	$\bot$
?	?	$\bot$	?

b)

$\vee^s$	$\top$	$\bot$	?
$\top$	$\top$	$\top$	$\top$
$\bot$	$\top$	$\bot$	?
?	$\top$	?	?

c)

	$\sim^w$
$\top$	$\bot$
$\bot$	$\top$
?	?

Let us now turn to the other third truth value. The value $\star$ expresses that both true and false are inappropriate. In that case, the conjunction of it with either true or false will also be inappropriate. The result is the weak Kleene conjunction (30a). Negation is more subtle, and depends on what we want it to mean. If the truth of a negation of an expression means that the expression is false, this leads to the same weak negation (30b). But there is another negation, strong negation, which expresses that the negated formula is not true (30c).

(30)

a)

$\wedge^w$	$\top$	$\bot$	$\star$
$\top$	$\top$	$\bot$	$\star$
$\bot$	$\bot$	$\bot$	$\star$
$\star$	$\star$	$\star$	$\star$

b)

	$\sim^w$
$\top$	$\bot$
$\bot$	$\top$
$\star$	$\star$

c)

	$\sim^s$
$\top$	$\bot$
$\bot$	$\top$
$\star$	$\top$

Of course, it is also possible to add both values to a logic, resulting in a four-valued logic. Another move that can be made is to stick with a three-valued logic, but add both conjunctions, which would result in the algebraic, functionally complete case again. Many-valued logics have been studied extensively in the literature (Haack 1974, 1978, Urquhart 1986, Blamey 1986). I will restrict myself to the logical fragment that I will use later.

Strict-partial Predicate Logic

In this thesis, I will need the inappropriate-value variant of three valued logic, because in natural language, anything that is irrelevant, will stay irrelevant under embedding. I will call this third value $\star$ the undefined value. The corresponding logical operators are all *strict*, they give the result true or false only if none of the arguments is undefined. As a functionally complete set of strict operators, we can take weak negation and weak conjunction, together with one additional operator, called the presupposition operator, which I write as $+$ (Krahmer (1995), following Beaver (1995) uses the (partial derivative) symbol ∂ to denote this operator):

(31)

	$+$
$\top$	$\top$
$\bot$	$\star$
$\star$	$\star$

Its most important use, as the name suggests, is to encode presuppositions. The presupposition operator makes a formula that can be true, but never false. The idea is, that presupposed information in an expression χ is embedded under this operator. in that case, if χ is true or false, the presupposed information *has* to be true. For example, in $+\phi \wedge \psi$, ϕ is presupposed: it is the consequence of both this formula and of its negation $\sim(+\phi \wedge \psi)$, because this last formula is equivalent[6] with $+\phi \wedge \sim\psi$.

Because there are three values, it is only necessary to specify two cases, the third case follows automatically. I will write $\|\phi\|^g_+$ to mean that ϕ is true under assignment g, $\|\phi\|^g_-$ to mean that ϕ is false under assignment g, and $\|\phi\|^g_d$ to mean that ϕ is defined (true or false) under assignment g. It follows that $\|\phi\|^g_d$ iff $\|\phi\|^g_+$ or $\|\phi\|^g_-$.

In terms of this, strictness can be defined as follows.

2.8 Definition (Strictness)
A one-place operator Φ is said to be strict strict if it satisfies

(32) $\|\Phi(\phi)\|^g_d \Rightarrow \|\phi\|^g_d$

and a two-place operator Ψ is said to be strict if

(33) $\|\Psi(\phi, \psi)\|^g_d \Rightarrow \|\phi\|^g_d \;\&\; \|\psi\|^g_d$

Traditionally, three valued logics are defined in terms of the cases true and false. But this is not at all essential. In the light of the strictness of the formulas, it makes more sense to define the logic by specifying under what circumstances a formula is defined and under what circumstances it is true. Of course, a formula is false iff it is defined but not true, ($\|\phi\|^g_-$ iff $\|\phi\|^g_d$ and not $\|\phi\|^g_+$). Note that for a predicate P we need to have two interpretation functions: $\mathcal{I}^+(P)$ will denote the extension for which the predicate is true, and $\mathcal{I}^d(P)$ will denote the extension for which the predicate is defined (cf. Muskens 1996, definition 33, where $\mathcal{I}^+(P)$ and $\mathcal{I}^-(P)$ are used in the definition of a three-valued logic).

2.9 Definition (Strict-partial Predicate Logic)
The syntax of SPL *is that of* PL*, The semantics of* SPL *is relative to a model $\langle \mathcal{D}, \mathcal{I}^+, \mathcal{I}^d \rangle$, with $\mathcal{I}^+ \subseteq \mathcal{I}^d$.*

(a) $\|Px_1 \ldots x_n\|^g_d$ iff $<g(x_1) \ldots g(x_n)> \in \mathcal{I}^d(P)$
$\|Px_1 \ldots x_n\|^g_+$ iff $<g(x_1) \ldots g(x_n)> \in \mathcal{I}^+(P)$

(b) $\|\sim\phi\|^g_d$ iff $\|\phi\|^g_d$
$\|\sim\phi\|^g_+$ iff $\|\phi\|^g_d$ and not $\|\phi\|^g_+$

(c) $\|+\phi\|^g_d$ iff $\|\phi\|^g_+$
$\|+\phi\|^g_+$ iff $\|\phi\|^g_+$

6. If $\sim(+\phi \wedge \psi)$ is true, both conjuncts must be defined, and at least one of them must be false. $+\phi$ is defined only if it is true, so ψ has to be false, so $\sim\psi$ is true, so $(+\phi \wedge \sim\psi)$ is true.
If $\sim(+\phi \wedge \psi)$ is false, both ϕ and ψ must be true, so $+\phi \wedge \sim\psi$ is false.

(d) $\|\phi \wedge \psi\|_d^g$ iff $\|\phi\|_d^g$ and $\|\psi\|_d^g$
$\|\phi \wedge \psi\|_+^g$ iff $\|\phi\|_+^g$ and $\|\psi\|_+^g$

(e) $\|\exists x\phi\|_d^g$ iff $\exists h \approx_x g\ \|\phi\|_d^h$
$\|\exists x\phi\|_+^g$ iff $\exists h \approx_x g\ \|\phi\|_+^h$

Of these, the conjunction, negation and existential quantifiers are strict extensions of the corresponding two-valued operators. They are defined whenever all their arguments are defined, and then have their usual, two-valued, meaning. That predicates are allowed to have all three values can be used to encode sortal information. For example $\mathbf{walk}$ might be such that $\mathcal{I}^d(\mathbf{walk})$ are exactly those entities with legs, and $\mathcal{I}^+(\mathbf{walk})$ those of them that walk. In this thesis, I will assume that predicates are always defined. In that case, the only other operator that gives rise to three valuedness is the presupposition operator $+$.

3 Generalized Quantifiers

I will now turn to the logical tools needed in chapter 3: generalized quantifiers and plurality.

Let us start with some basic facts concerning quantifiers (cf. (van Benthem 1986a, Westerståhl 1989, van Benthem 1991)).

Most important for our purposes is that a quantifier is a *functor* which assigns a relation among relations to each non-empty domain. Concentrating on the special case of relations between two sets of sets (two 1-place relation), also called type $<1,1>$ quantifiers (pronounced "one-one quantifiers"), this becomes:

3.1 Definition
A type $<1,1>$ quantifier *is a functor which assigns to each non-empty domain E a two place relation Q_E between sets:*

$$Q_E \in \wp(\wp(E) \times \wp(E))$$

In case of natural language quantification not all such functors are realized. They have to satisfy some general constraints such as CONS, EXT, ISOM, and FIN, which we shall introduce now.

3.1 General Properties

The following constraints are standard in the literature.

3.2 Definition (CONS)
A quantifier Q is conservative (CONS) *iff for all E and $A, B \subseteq E$: $Q_E AB \Leftrightarrow Q_E A\ A \cap B$*

But for a few exceptions (*only* is perhaps the only one), all natural language quantifiers have CONS.

3.3 Definition (EXT)
A quantifier Q has extension (EXT) *iff for all E, E' and $A, B \subseteq E \subseteq E'$: $Q_E AB \Leftrightarrow Q_{E'} AB$*

EXT says that Q remains constant under growth of E. In combination with CONS it yields the principle UNIV, which captures the idea that the domain of quantification coincides with the first argument.

3.4 Definition (UNIV)
A quantifier has UNIV *iff for all E and $A, B \subseteq E$: $Q_E AB \Leftrightarrow Q_A A\ A \cap B$*

3.5 Proposition
A quantifiers Q has UNIV *iff Q is* CONS *and has* EXT.

In case of UNIV we can be sloppy in regard to the domain E by stipulating that QAB is short for $Q_E AB$ for some superset E of A and B.

A further constraint on Q reflects the intuition that quantifiers are about quantities rather than qualities: they should be insensitive to permutations on the base domain.

3.6 Definition (ISOM)
A quantifier Q has ISOM *iff for all bijections $\pi : E \longrightarrow E'$:*

$$Q_E AB \Leftrightarrow Q_{E'} \pi(A)\pi(B)$$

where $\pi(A) = \{\pi(a) : a \in A\}$.

Let us agree that from now on a **quantifier** is a functor Q which is CONS, and has EXT and ISOM. It can be shown that the truth of a statement QAB only depends on the cardinals $\|A \cap B\|$ and $\|A - B\|$.

With a view to natural language semantics its seems reasonable to restrict attention to finite domains.

3.7 Definition (FIN)
FIN *is the constraint that only finite E are considered.*

Finally, one often adopts constraints to the effect that Q is *non-trivial* (not always true or always false for pairs of sets). Since this constraint has no role to play here I refer to Westerståhl (1989) for details.

3.2 Special Properties

Besides the general properties, a quantifier Q may have different kinds of special properties. We mention the most prominent.

3.8 Definition (MON)
The right monotone *determiners Q are those with:*

MON↑: *if* QAB *and* $B \subseteq B'$, *then* QAB': $\dfrac{\textit{all women walk}}{\textit{all women move}}$

MON↓: *If* QAB *and* $B' \subseteq B$ *then* QAB': $\dfrac{\textit{no men move}}{\textit{no men walk}}$

The left monotone *determiners* Q *have:*

↑*MON: If* QAB *and* $A \subseteq A'$ *then* $QA'B$: $\dfrac{\textit{some women walk}}{\textit{some people move}}$

↓*MON: If* QAB *and* $A' \subseteq A$ *then* $QA'B$: $\dfrac{\textit{all people walk}}{\textit{all women move}}$

Examples of MON↑ determiners are: *all*, *some*, *at least two*, while *Not all* and *no* are MON↓ determiners. *Some* and *not all* are ↑MON; *All* and *no* are ↓MON. There are also *non-monotonic* determiners; e.g., *exactly two* and *an even number of* are neither MON↑ nor MON↓. MON↑ and MON↓ quantifiers are called *monotone increasing* and *monotone decreasing* respectively by Barwise and Cooper (1981); ↑MON and ↓MON correspond to upward and downward versions of their *persistence*.

3.9 Definition
A quantifier Q *is* continuous *iff*

$$QA \bigcup_{i \in I} B_i \iff \exists i \in I \;\; QAB_i$$

Some is a prime example of a continuous quantifier.

I will leave it at this for the terminology of generalized quantifier theory. I will have some more things to say about, among other things, lifts on more complex domains in the logical issues section of chapter 3.

4 Plurality

Introduction

Languages often distinguish between singular and plural noun phrases. In such languages, singular NPs denote entities one at a time, while plural NPs denote entities as a group. A theory of plurals should be able to refer both to entities (singular objects) and to sets of entities (plural objects).

The theory of plurals I take as my starting point in this thesis is the received treatment of plurals as fully generalized by van der Does (1992). His work is based on the plural logic, formalizing the behavior of numerals and standard quantifiers, by Scha (1981), which itself was based on Bennett (1974) and Bartsch (1973). In this logic, the basic objects are sets of entities, the singulars being represented by singleton sets. Scha's logic has for a long time been reluctant to give way to criticism, and the generalizations of van der Does show its stamina. Except for some minor changes to the definition of collective, the plural logic defined in chapter 3, and made dynamic in chapter 4, is identical to that of van der Does, if we restrict ourselves to to the cases discussed in his dissertation.

Another comprehensive theory of plural dynamics is that given by Kamp and Reyle (1993). The starting point of that work is different: a given theory of dynamics is extended with a notion of plural object, but no changes to the dynamics are made, I will argue that this is unlikely to work, given the difference in behavior of plural and singular pronouns. In this thesis, I start from the other end and give a dynamic interpretation to the theory of plurals. Despite the differences, the two theories give surprisingly similar interpretations to a large class of cases. Surprising, because the way the results are arrived at are so different.

The plural theory used by Kamp and Reyle is another one —that of Link (1983)— which formalizes singulars and plurals as objects in a complete boolean lattice[7]; but this is a minor difference. Such a lattice will always be equivalent to a lattice of sets, and in fact the whole chapter in Kamp and Reyle's book (1993) dealing with plurals can be read as if it is about sets.

4.1 Higher Order Sets

If the entities we have to deal with are plural objects, how complex should such objects be? There seem to be no nominal or verbal expressions which denote objects as complex as sets of sets. It may seem, therefore, that we need not go beyond sets of entities. However, some of the more successful theories in the literature do include higher order types. For example, Landman (1989) and Hoeksema (1981) suggest that semantics may need the whole of Cantor's universe up to level ω. Because the logic in this thesis does not have higher sets it is important to understand why, though the arguments for higher types are valid, they do not apply to the issue at hand.

There are two applications of plural logic that occur, and these should not be confused. One application formalizes the way in which language uses plural nouns. The other is the formalization of the way we structure the world and reason about it using the language. The difference can be understood by looking at a classic example of Hoeksema, indicating why he, Landman and others argue for higher sets:

(34) *Blücher and Wellington and Napoleon fought against each other.*

In (34) the set {**blücher**, **wellington**, **napoleon**} cannot be denoted by the noun-phrase because of the asymmetry between the two bracketings [*Blücher and Wellington*] *and Napoleon fought against each other*, a true statement, and *Blücher and* [*Wellington and Napoleon*] *fought against each other*, a

7. This more abstract view of plural objects rather than sets is sometimes argued to make the similarity between plurals and mass terms more obvious, but I regard the counter arguments of van der Does (1992) and Landman (1989) convincing, and personally think that the similarity would be just as well, if not better, illustrated by interpreting mass-terms as (dense) sets. Even though the elements of such a set may be somewhat of an enigma, they are hardly more counterintuitive than a plural "object" consisting of three earrings, the sandwich I had for lunch, a dream of Bermuda, and two slightly deaf waiters. A related interpretation of mass-terms as sets, which seems very promising has recently been proposed by Chierchia (1996).

false statement. This asymmetry cannot be expressed when the noun-phrase corresponds to a set of individuals. However, as van der Does (following Schwarzschild (1990)) argues, this constitutes at most an argument against the interpretation of conjunction (*and*) as set-formation. In fact, the difference between the two occurrences of *and* is expressed in spoken language. On the truthful reading of (34), there is a shorter pause between *Blücher and Wellington* than there is between this term and *Napoleon*. If you agree with me that intonation belongs to syntax proper, you can see this as a difference in the syntax. At the same time, I do not think that we have to choose between one or the other theory. Both are equally valid; they just happen to formalize different domains. The simple set-oriented approach of van der Does (et.al.) is related to the internal, linguistic, structure of language, whereas the *all sets are wild* approach of Landman (et. al.) deals with the external, informational, structure of language, i.e. the structure of the entities in the domain (cf. Scha and Stallard 1988, Schwarzschild 1992b Chierchia 1996). The former formalizes how language expresses information about the world, the latter formalizes how we reason with this information. In this thesis, which deals exclusively with the internal structure of language, there is no need to look beyond simple sets of entities. It is for that reason that the line of van der Does (et. al.) is chosen.

4.2 Quantifier Readings

After discussing some alternatives, van der Does (1992) argues for three readings of quantifiers, which generalize the treatment of numerals by Scha (1981). These readings are the *collective* reading, the *distributive* reading and the *pseudo-distributive* reading. The readings can best be understood by looking at a simple example:

(35) *Four women lifted a piano.*

Let the collection of all sets of women be denoted by $\mathbf{X}$, and the collection of all sets that lift a piano by $\mathbf{Y}$ (note that a set may consist of one element). Then the readings are:

Collective A set consisting of four women lifted a piano together The collective reading says that there is a set of women ($A \in \mathbf{X}$) the cardinality of which is 4 ($\#A = 4$), and which is also the extension of piano lifting group ($A \in \mathbf{Y}$).

Distributive The total number of women such that every one of these lifted a piano is four The distributive reading says that the (maximal) set of women ($A \in \mathbf{X}$) such that all elements lifted a piano ($\forall a \in A\ \{a\} \in \mathbf{Y}$) has cardinality 4 ($\#A = 4$).

Pseudo-distributive[8] This reading says that if we add all groups of women lifting pianos together, the resulting total group has four elements. The pseudo-distributive reading says that the set of women ($A \in \mathbf{X}$) that were involved in lifting a piano ($\forall a \in A\ \exists B \subseteq A\ a \in B\ \&\ B \in \mathbf{Y}$) has

7. The name pseudo-distributive was also used by Schein (1993), van der Does (1992) calls it neutral$_2$ and Scha (1981) calls it collective$_2$.

cardinality 4 ($\#A = 4$). Or put differently, the union of all sets that lifted a piano has cardinality 4.

Note that the collective reading is true if there is at least one such set of women, there might be other sets of other cardinalities. In this the reading differs from the other two. These imply that there is one unique set with the appropriate cardinality. In the case of the distributive interpretation, this is the set of all women who individually lifted a piano, in the case of the pseudo-distributive interpretation, the set of all women involved in lifting a piano as member of an all women group. However, note that the set of women has to be a set of women that did the lifting together without assistance: the set has to be in the extension of $\mathbf{Y}$, and not just a subset of some set in $\mathbf{Y}$. This means that if I say *Four women lifted a piano* in order to describe a situation in which only one piano was lifted but by four women and two men together, I would be speaking falsely.

In my view all three readings demand a form of *maximality* of the sets involved. In case of the (pseudo-) distributive no-one disagrees with that, but I think that the previous discussion shows that also the collective has some kind of maximality built in[9]. This maximality should not be confused with uniqueness. Below it will be shown that the pseudo-distributive interpretation involves a very strong form of uniqueness claim on the set involved: it demands that this set contains all individuals involved in the lifting, and the distributive exhibits an slightly weaker form of uniqueness claim: it demands that this set contains all individuals involved in the individual liftings, but the collective does not involve any uniqueness claim at all. It is shown by van der Does and Verkuyl (1995) that the use of unique collections drastically reduces the number of readings.

It is instructive to look at another definition of neutral quantification that van der Does discusses and rejects (he denotes it with N_3):

Neutral$_3$ The total number of women such that every one of these was involved in lifting a piano is four. The neutral reading says that the (maximal) set of women ($A \in \mathbf{X}$) such that all elements were involved in lifting a piano ($\forall a \in A\ \exists B \ni a\ \ B \in \mathbf{Y}$) has cardinality 4 ($\#A = 4$).

This one differs from the rest in not demanding that the women were on their own. Suppose, for example, that there are four disjoint groups lifting pianos, and every group consists of one women and three men. In that case, none of the readings will apply except the neutral$_3$ reading. As van der Does also notes, this reading is not very probable in general. A sentence like *four women and twelve men lifted a piano* is a more likely utterance describing the situation (it allows the pseudo-distributive reading), but (35) seems inappropriate in that case. It is clear that there are phrases in language, like *involved in* or *contribute to*, that result in the neutral$_3$ reading as (part of) their meaning. However, the neutral$_3$ reading does not correspond to a reading of the quantifier.

9. In van der Does (1992) the collective is not demanded to be maximal, only the distributive and pseudo-distributive are. His definition of collective is equivalent with the notion of adjectival collective defined later.

4.3 Domains and Predicates

It is notoriously difficult to get one's intuitions straight about plurals. For one, there is the question as to what the domain is in which quantification is done. Where do I look for women lifting pianos to verify a sentence like (35)? In my field of vision? in my town? The whole world?

To fix the domain of quantification, Westerståhl (1989) introduces the notion of a context set. This set contains exactly the entities that the quantification is about. The name *context set* suggests that the domain of quantification is determined by the context. In the next chapter, I will identify context sets with discourse referents. This means that a quantifier is anaphoric to an earlier quantifier (including indefinites) by virtue of its domain of quantification. This will give a dynamic account of Westerståhl's original intuitions about context dependency of quantifiers. In this chapter I will assume that the quantifier is interpreted in a domain that is just right: it contains exactly what we need and nothing more. As far as I see, this is what almost everyone else does.

There is also the question of what exactly makes a set to be in the extension of a predicate. In example (35), I tacitly assumed that the predicate expresses a kind of event-compactness: a group is in the predicate corresponding to *lift a piano* if that group as a whole was involved in the lifting of one piano. To me this seems the only realistic choice, but other choices *are* possible. For example, one may define a set to be in the extension of the predicate if the elements are involved in the property expressed by that predicate. This would make distributive, pseudo-distributive and neutral coincide.

4.4 Scope Ambiguities

If a sentence contains two quantifiers, new possibilities arise due to the interactions of the quantifiers. If we assume, that the above discussion on quantifier readings holds for quantifiers in any position, both quantifiers have three readings each. The result is that there are 3×3 readings for (36):

(36) *Four women lifted two pianos.*

The combination of two quantifiers also gives rise to a new reading, called the cumulative reading by Scha (1981). This reading involves a simultaneous quantification *Four women lifted pianos and two pianos were lifted by women*. These are reasonably straight-forward, and will be discussed further in chapters 3 and 5. But another kind of ambiguity can be found in the literature: those caused by quantifier scope inversions.

It is difficult to trace the ultimate origins of this idea, but surely Montague contributed a lot in this respect. It has often been claimed that

(37) *Every man loves a woman*

has two readings, one where the universal quantifier has wide scope (ws), and one where it has narrow scope (ns). The two readings are

(ws) $\forall x\,(\mathbf{man}(x) \rightarrow \exists y\,(\mathbf{woman}(y) \wedge \mathbf{love}(x,y)))$,

(ns) $\exists y\,(\mathbf{woman}(y) \wedge \forall x\,(\mathbf{man}(x) \rightarrow \mathbf{love}(x,y)))$.

There are two separate questions to consider here. First there is the question whether or not there are two different readings to begin with. And second, if so, whether a difference in scope is the right representation for that meaning difference.

As to the first point: it was observed by a number of people (cf. Verkuyl 1993) that in the simple case (37) the wide scope reading actually implies the narrow scope reading. This might suggest that there is no ambiguity in this case. This position is strengthened by the observation that as soon as there is no such implication relation, there does not seem to be a reading where the object quantifier has wide scope[10]. For example, in the distributive reading of (36), where every woman lifted two pianos, the object quantifier cannot take scope over the subject quantifier. although

(38) $4x(\mathbf{woman}(x) \wedge 2y(\mathbf{piano}(y) \wedge \mathbf{lift}(x, y))),$

is a possible meaning of the sentence, the expression with the two scopes reversed:

$$2y(\mathbf{piano}(y) \wedge 4x(\mathbf{woman}(x) \wedge \mathbf{lift}(x, y))).$$

is not a possible one. This last expression would mean that it is possible that each piano was lifted by four different women, involving eight women. This does not seem the case.

Still, I do not concur with the opinion that sentences are not ambiguous in this way. The reason is that it turns out, that the distributive reading of (36) *is* ambiguous in a way similar to (38). There is one reading, the standard distributive, which states that every one of the women lifted two pianos, but not necessarily the same ones. This is the reading formalized by (38). And there is a second reading that demands that the same two pianos are lifted by each individual woman.

In chapter 5 I will argue that this last reading is in fact a variant of the same process that gives rise to the cumulative reading mentioned above. It can be paraphrased as

> *Take the women that lifted two pianos. There are exactly four such women and exactly two pianos.*

Compare this with the traditional cumulative:

> *Take the women that lifted pianos. There are exactly four such women and exactly two pianos.*

The same analyses can be applied to the original example. The reading that was called the narrow scope reading (of the universal quantifier) before corresponds to

> *Take the men that love a woman. The set of men comprises all men and there is exactly one woman.*

---

10. This relates to the issue of island constraints, which is something I cannot go into here.

In chapter 5 an interpretation of the cumulative reading is given that can explain this example, and also explain how a sentences like *No man danced with no woman* can have a reading which means that no dancing took place between a man and a woman. As you might guess, this means there is more than one possible version of the cumulative, only one of which is the one defined by Scha (1981)

In as far as this chapter is concerned I conclude that in English in simple sentences subjects always have wide scope over the object. In general, in English quantifier scopes seem to follow the left to right order of the sentence very closely. The exceptions to this are rare (Also cf. Scha 1981, Liu 1990, Verkuyl 1993).

4.5 Distributivity

Distributivity is a very important notion: useful, but much abused. It pays to have a closer look at it. The notion is used to describe properties of predicates (or formulas in general) and quantifiers. What leads us to come up with this notion are sentence like the following:

(39) *Every man ate a hotdog.*

(40) *Four men ate a hotdog.*

The sentence (39) implies that there are (at least) as many hotdogs as there are men, or, equivalently, that there is a 1-1 (1-many) correspondence between hotdogs and men eating them. Sentence (40) has a similar reading, in which there are (at least) four hotdogs, one for each man. This reading is called the distributive reading.

Distributive readings are special, because they are what are often called the really quantificational readings. A distributive quantifier inspects the elements of its arguments one at a time, making it possible to exactly count each element's contribution, putting the "quant" in quantificational to work. NP's translating as real quantifiers are often distinguished from definites and indefinites. Although in this work I try to argue that all determiners should be treated in a uniform way, the differences in behavior of these different classes have to be explained, and understanding distributivity constitutes an important part of understanding the different classes of quantifiers.

Conclusions

I now have given all the background you need to read the rest of the thesis. The important facts from this chapter are the following.

- All determiners introduce referents, even those that normally just seem to pick up a referent and predicate something of it or seem just to quantify over a domain. In fact, in large part, that domain *is* a referent.
- for understanding discourse, truth of that discourse isn't all important, even if, in the end, we want to know whether a text describes what it describes truly. It makes sense to keep track of whether a discourse is defined, independently of whether it is true.

If you want, you may now turn to chapter 4, and see what the result is of adding everything together, and then see it applied in chapter 5 to a large number of cases. But if you also want to know why the theory has exactly the form it has, you will have to first read chapters 2 and 3.

2

Partial Dynamic Predicate Logic

Introduction

A theory of long distance dependencies between terms is an important part of the semantic component of any formal treatment of discourse. Over the decade, beginning with the well known discourse representation theories (DRT) of Kamp (1981) and Heim (1982), several dynamic semantic formalisms have been developed to provide treatments of long distance dependency. Although the work of Kamp and Heim is the best known work in this area, in this chapter I discuss variations on a later, more traditionally formatted formalism: Groenendijk and Stokhof's (1991) dynamic predicate logic (DPL). I have chosen DPL as the basis for the system developed here largely because the formatting conventions adopted in DPL allow for the easy adaptation to dynamic semantics of theories of plurals and generalized quantifiers that are now commonly phrased in standard predicate or type logic.

This chapter consists of five sections.

In section 1, I repeat the definition of Groenendijk and Stokhof's DPL, discussed in chapter 1. I briefly discuss the traditional formalization of generalized quantifiers in that framework, a subject that will become more important in the next chapters. I finish with a discussion of some problems with negation in DPL. In section 2, I suggest to partialize the dynamics of the logic, resulting in a variant of DPL that has a dynamic negation. Care is taken to not partialize the truth conditions: formulas are still true or false. In section 3, additional operators licensed by the partial logic are introduced, resulting in a partialization of the static truth conditions. In Section 4, a third partialization is proposed: replacing total assignments by partial assignments to account for the difference between introduced and non-introduced discourse referents. This section rounds up by proposing a logic called full dynamic predicate logic (FDPL).

Finally, in section 5, more mathematical issues are raised, and it is argued that the given definition (of FDPL) is also a natural one from a mathematical point of view.

I now start with DPL as developed by Groenendijk and Stokhof.

1 Dynamic Predicate Logic

In chapter 1, I discussed the basic dynamic predicate logic (DPL), defined by Groenendijk and Stokhof. The definition is repeated here for ease of reference (cf chapter 1 section 2.3 for details). I only replaced the $\exists x\ \phi$ clause by a definition of ε_x.

1.1 Definition (DPL)
The syntax *of* DPL *is that of ordinary predicate logic with identity. However, the predicate logic used does not have constants. The* semantics *of* DPL *interprets formulas as relations between assignments.*
The interpretation $[\![\phi]\!]^{\mathcal{M}}$ *of a formula* ϕ *as a relation between states is defined with respect to a standard first order model* $\mathcal{M} = \langle D, \mathcal{I}\rangle$*, where* D *is a domain of entities and* $\mathcal{I}$ *a function that assigns to any n-place predicate* a set *of n-tuples in the standard way. I will suppress the model* $\mathcal{M}$ *where possible;* $[\![\phi]\!]$ *is characterized as follows:*

(a) $g[\![Px_1 \ldots x_n]\!]h$ iff $g = h$ and $<g(x_1), \ldots, g(x_n)> \in \mathcal{I}(P)$

(b) $g[\![x = y]\!]h$ iff $g = h$ and $g(x) = g(y)$

(c) $g[\![\phi \wedge \psi]\!]h$ iff $\exists k\,(g[\![\phi]\!]k$ and $k[\![\psi]\!]h)$

(d) $g[\![\varepsilon_x]\!]h$ iff $g \approx_x h$

(e) $g[\![{\sim}\phi]\!]h$ iff $g = h$ and $\neg\,\exists k\ (g[\![\phi]\!]k)$

where $g \approx_x k$ *means that* $g(z) = k(z)$ *for all variables* z *not equal to* x.

A formula is true in a given state if there is at least one possible output for that state ([1:2.4]).

(1) $\|\phi\|^g = \mathbf{true}$ if $\exists k\ g[\![\phi]\!]k$.

If no such transition exists, the formula is said to be false:

$\|\phi\|^g = \mathbf{false}$ if $\neg\,\exists k\ g[\![\phi]\!]k$.

1.1 Generalized Quantifiers in DPL

In standard DPL, only two quantifiers are discussed, the existential and the universal. These two behave in completely different ways. The existential quantifier is a dynamic operator. It introduces a referent in the discourse that can be picked up. On the other hand, the universal quantifier is a static operator. Its definition in terms of the negation ($\forall = {\sim}\exists{\sim}$) makes that its argument is closed off.

There does not seem a way around making this distinction in DPL. The existential quantification $\exists x \phi$ introduces some value for a variable that satisfies ϕ. In some sense, a value that satisfies the scope can be considered one of the values that the quantification *is about*. But the universal quantifier $\forall x \phi$ has to inspect all values for x, and check whether they all satisfy ϕ. If it makes any sense at all to speak of something the quantification *is about* then it is all the values of x together. But that would involve a treatment of plural discourse referents, and that will have to wait until chapters 3 and 4. If we limit ourselves to singular referents, claiming that the universal quantifier does not introduce a referent seems to be the best choice.

Given these two paradigms for quantifying, the dynamic and static ones, we can ask the question: Which of the two is best suited for the treatment of arbitrary generalized quantifiers in a dynamic logic like DPL. As the general discussion in chapter 1, section 1 suggests, the singular indefinite is somewhat special in being one of the few quantifiers that introduces one singular value (though not necessarily unique) in the context. The case of the universal quantifier, which involves several values together, is much more typical for quantification in ordinary language.

Because DPL deals exclusively with singular discourse referents, and generalized quantifiers need plural ones, the most effective treatments in the literature of dealing with generalized quantifiers within standard DPL follow the example of the universal quantifier in being externally static (Chierchia 1992, van Eijck and de Vries 1991).

Generalized quantifiers are relations between sets of entities. The first step to finding an expression for generalized quantifiers in dynamic logic is therefore finding a good definition of the *set corresponding to a formula* ϕ. In static predicate logic, the set assigned to x by a formula ϕ (where $g[x := d]$ denotes the assignment h such that $h(x) = d$ and $g \approx_x h$) can be written as:

$$\langle\phi\rangle_x(g) := \{d \mid \mathcal{M}, g[x := d] \models \phi\}$$

Another way of writing this is to use the notation $\|\phi\|^{g,\mathcal{M}}$, suppressing the model

$$\langle\phi\rangle_x(g) := \{g(x) \mid \|\phi\|^g\}$$

And it is this formulation that we can use in dynamic logic. In (1) it is defined when a dynamic formula is true. This definition can be substituted, resulting in:

(2) $$\langle\phi\rangle_x(g) := \{g(x) \mid \exists h \;\; g[\![\phi]\!]h\}$$

which is the expression for a set corresponding to a dynamic formula.

In terms of this, we can now copy the definitions of generalized quantifiers in standard generalized quantifier theory. But there is a problem in dynamic logic that does not occur in static logic. There is more than one way to phrase the definition, which does not mean the same when interpreted dynamically. To illustrate this, let us first give a meaning to $\forall x\ (\phi, \psi)$ in terms of these sets that is equivalent to what the definition of the universal quantifier in the previous sections would assign to $\forall x\ (\phi \to \psi)$.

In all but the simplest cases, ψ will contain variables bound by quantifiers inside ϕ, the internally bound variables. It is therefore impossible to use the set $\langle\psi\rangle_x(g)$ to represent the nuclear scope, because that would have the internally bound variables in ψ take incorrect values. To account for quantifiers in ϕ binding variables in ψ, ψ is only considered in the dynamic scope of ϕ. This has the important consequence that dynamic quantifiers will always be conservative. Furthermore, the reason we can do no more than this —the fact that otherwise the nuclear scope is ill-defined— does not give much hope that a definition for dynamic non-conservative quantifiers can be found. It seems that once we are committed to dynamics, we are also committed to conservativity. Dynamic semantics therefore supplies an independent argument for only allowing conservative quantifiers.

There are essentially two ways in which ψ can be in the context of ϕ, either we calculate the set $\langle \phi \wedge \psi \rangle_x$, or we calculate the set $\langle \phi \to \psi \rangle_x$. In static predicate logic, these two sets are necessarily the same (cf. Kanazawa (1993a)) but dynamically these are very different, because $\to$ contains a negation, which universally quantifies over output states (cf. chapter 1, corollary 2.3).

The two resulting forms of universal quantification are:

(3) $g[\![\forall x(\phi, \psi)]\!]h$ if $g = h$ and $\langle \phi \rangle_x(g) \subseteq \langle \phi \to \psi \rangle_x(g)$

(4) $g[\![\forall x(\phi, \psi)]\!]h$ if $g = h$ and $\langle \phi \rangle_x(g) \subseteq \langle \phi \wedge \psi \rangle_x(g)$

The first Chierchia names the **strong reading**, and the second he calls the **weak reading**. Of these, the strong definition is similar to the definition of universal quantification given in the previous sections.

This schema can be generalized to other quantifiers, so given a generalized quantifier Q, the dynamic quantifiers Q^w and Q^s are defined by:

(5) $[\![Q^s x(\phi, \psi)]\!](g, h)$ if $g = h$ and $\mathcal{Q}\left(\langle \phi \rangle_x(g), \langle \phi \to \psi \rangle_x(g)\right)$

(6) $[\![Q^w x(\phi, \psi)]\!](g, h)$ if $g = h$ and $\mathcal{Q}\left(\langle \phi \rangle_x(g), \langle \phi \wedge \psi \rangle_x(g)\right)$

Taking this from the other end, Kanazawa observes that there are two ways of expressing conservativity, which in static logic are equivalent, but in dynamic logic are different:

(7) Q is strongly conservative if: $Qx(\phi, \psi)$ iff $Qx(\phi, \phi \to \psi)$,
(8) Q is weakly conservative if: $Qx(\phi, \psi)$ iff $Qx(\phi, \phi \wedge \psi)$.

The difference is caused by the universal quantifier that is hidden in the implication. Remember that $((\exists x \phi) \to \psi)$ is equivalent to $(\forall x(\phi \to \psi))$. In other words, if the weak version is the right expression of conservativity, $Q(\exists x \phi, \psi)$ is equivalent to $Q(\exists x \phi, \exists x \phi \wedge \psi)$, but if the strong version is the right one, we have $Q(\exists x \phi, \forall x(\phi \to \psi))$. These are totally different.

If we apply both readings to the *(universal) donkey sentence*:

Every farmer who owns a donkey beats it.

We get for the strong reading the one we already saw earlier, and which can be paraphrased as

Every farmer who owns a donkey beats every donkey he owns.

But the weak reading results in a different meaning, which can be paraphrased as:

Every farmer who owns a donkey beats some donkey he owns.

Kanazawa then points out that, contrary to what is claimed in Chierchia (1992), not all quantifiers seem to be ambiguous between the weak and the strong version, but there seems to be a relationship between the (right-) monotony properties of a quantifier and which reading they prefer. There is a discussion of this in section 3.2 of chapter 5.

The resulting meaning of the dynamized quantifier always results in a test ($g = h$). Given the argument in the first chapter, that quantifiers translating natural language expressions need always have a dynamic effect, such quantifiers have limited use for real language.

But its consequences are very interesting as far as they go. It gives a quantifier with internal dynamic behavior, that has the right truth behavior. As an example, consider:

Most farmers who have a donkey beat it.

The pronoun *it* gets bound by the quantifier in the subject. The weak quantifier reading gives as a translation:

(9) $g[\![\mathsf{MOST}x\ (\ \mathbf{f}(x) \wedge \exists y.\mathbf{d}(y) \wedge \mathbf{h}(x,y)\ ,\ \mathbf{b}(x,y)\ )]\!]h$

$$\text{iff } g = h \text{ and } \mathsf{MOST}(\ \langle \mathbf{f}(x) \wedge \exists y.\mathbf{d}(y) \wedge \mathbf{h}(x,y)\rangle_x(g), \langle \mathbf{f}(x) \wedge \exists y.\mathbf{d}(y) \wedge \mathbf{h}(x,y) \wedge \mathbf{b}(x,y))\rangle_x(g)).$$

In words this last line reads:

Most farmers who have a donkey are farmers who have a donkey they beat.

1.2 Negation and other Closure Operators in DPL

Negated formulas are closed. They are tests just like an atomic formula. The output state of a negated formula is by definition always the same as the input state. Any changes that might have been made to the state by quantifiers inside the negated formula are lost. Because of this, the negation of a formula ϕ hides all the quantifiers that might occur in ϕ.

Consequently, all logical operators defined in terms of negation, and these include disjunction ($\vee$), two kinds of implication ($\rightarrow$ and $\leftarrow$) and universal quantification ($\forall$), are also closed, and can have no outward dynamic effects. This also holds for the double negation, which therefore does not preserve the meaning of a formula ($[\![\sim\sim\phi]\!] \neq [\![\phi]\!]$).

This is in accordance with Discourse Representation Theory (DRT), the theory developed by Kamp (1981) which DPL is intended to mimic, so this is exactly what we expect. In the tradition of DRT the closure property of negation and of the operators defined in terms of negation is seen as an advantage. It is a property that is used to explain why certain configurations of quantifiers and pronouns do not occur in natural language, namely exactly those where the actual existential quantifier is hidden under a negation, as is the case with the quantifiers corresponding to *no*, ($\sim\exists$) and *every* ($\sim\exists\sim$):

*No man walks in the park. *He is ill.*

*Every man walks in the park. *He whistles.*

Although we do not want to argue against this evidence, evidence to the contrary is also discussed by Groenendijk and Stokhof (1991). For example the following seem perfectly acceptable ((10) slightly less so than (11)):

(10) *No man walks in the park. They are ill.*

(11) *Most men walk in the park. They whistle.*

The first of these might be explained as a CN-anaphor, which are anaphors where the pronoun refers back to the *common noun* in the preceding sentence and not to the *noun phrase*, but (11) cannot be explained along these lines.

Groenendijk and Stokhof also quote another famous example

(12) *It is not true that Sarah does not have a car. It is right outside.*

This seems a good example showing that double negation-like phenomena in natural language really allow for dynamic binding. But this is somewhat of a two-sided sword. For suppose we know that in fact Sarah has three cars. Then the second sentence of (12) is not felicitous. But *They are right outside* would be perfectly suitable as second sentence. So might this be, as is sometimes argued, an entirely different phenomenon? Is it maybe something like that famous marble that Partee lost?

(13) *I lost ten marbles but only found nine…   …Maybe it's under the couch.*

If the pause between the two sentences is long enough, *it* can in fact be used to refer to the lost marble, even if it is not syntactically present in the context.

The usual approach to (13) is to let the pronoun pick up some referent that is salient in the context. This process never fails; there is always something more salient than other things, so provided we find a good theory of saliency, we could well do without dynamic logic (or DRT) altogether, and use artificial intelligence methods instead. But there does seem to be a difference in the amount of processing needed to understand examples like (13), and examples like (12). So what does happen? One suggestion might be, that negation blurs the singular/plural distinction a bit. In chapter 5, I will argue that a more general phenomenon, namely the ambiguity of any indefinite noun-phrase between a singular and distributive reading, is responsible for the particular binding in (12), and that *it* is just an E-type pronoun, picking up the set of cars owned by Sarah.

Even if we believe that DRT and DPL are only meant to formalize the behavior of singular pronouns, this still leaves the apparent correctness of sentences like

(14) *Just one woman loved a man. He was a blond.*

(15) *Every girl gave every boy a present. She put it on his desk.*

to be explained (also cf. van der Does (1993), pg. 239).

I regard the evidence *against* "closed" negation to be much stronger than that the evidence *in favor* of it. Any theory should at least explain (15) and (12), and preferably also (11) and (10)

In the rest of this chapter I will develop a variant of DPL that does have a dynamic negation. The resulting logic will not always give right predictions about the behavior of pronouns. Giving up closed negation means giving up the correct predictions that follow from it. However, in chapter 5 I will regain most of these predictions in terms of the interactions of the singular number operator (defined there) and the singularity or plurality of the discourse referent that a given pronoun refers to. If the pronoun is singular and the referent plural the interaction would make the formula undefined. Other properties of discourse that are correctly predicted by using closed negation have to do with the syntactic component of the discourse grammar. For example, sometimes a pronoun follows the semantic number of an antecedent, sometimes the syntactic number, and these need not be the same, as (15) shows. In (van den Berg 1996a) I address this side of the story.

2 Partializing the Transitions

In this section and the next, I will move from total to partial dynamic and truth conditions, or to put it an other way, I will interpret formulas as three valued relations. This move is argued on the basis of a re-analysis of negation, and on a related general consideration of what it means to be a meaningful transition between states corresponding to some formula. In this section I will restrict attention to partializations that effect the dynamic conditions, but still result in total truth conditions. This means that although the relations that interpret the formulas are partial relations and are not defined for all input-output-pairs, there will always be an output for any given input for which the relation is defined: every formula is either true or false. I prove this in propositions 2.2 and 2.15. In the next section, I will then use the newly gained power to introduce new operators that *do* have partial truth conditions.

Dekker (1993) also discusses a dynamic alternative to negation, but his approach is somewhat different, and uses the much more complex framework of dynamic Montague grammar. The resulting operator is not the same as the one defined here.

2.1 Partiality and Negation

It is instructive to see how far we get if we define negation in a more traditional fashion, as an operator complementing the truth-values. Given the radically different way dynamic logic is defined, the first obstacle is to answer the question as to what it would mean to be more traditional. Then we will have to change the other definitions so that the resulting logic with the new negation stays as close to the original DPL as possible.

The simplest choice is what I will call *weak dynamic negation*:

(16) $g[\![-\phi]\!]h$ holds iff $g[\![\phi]\!]h$ does not hold.

The problem we face is that in dynamic logic the formal mechanism that encodes anaphoric binding and the truth definition of formulas are not orthogonally given. In DPL, a formula ϕ is said to be true in an assignment g iff $\exists h \;\; g[\![\phi]\!]h$. Consequently, a formula can only be true if there is an output state that *makes* it true. In DPL it is made sure that formulas are strictly two valued by definition. Formulas are said to be false if it is not true, i.e. if there is no output making the formula true.

There is a strong tension between the way states are "carried over" from one formula to the next, carrying anaphoric information, and the definition of negation as a complement. This already becomes apparent if we look at how predicate formulas behave under negation.

In static predicate logic (PL) the definition for atomic formulas is

$$\|Px\|^g \;\; \text{iff} \;\; g(x) \in \mathcal{I}(P).$$

It is instructive to rewrite the DPL definition in terms of this and to give both the clause that defines when a formula holds, and the clause that defines when a formula does not hold.

(17) $g[\![Px]\!]h$ *holds if* $g = h$ *and* $\|Px\|^g$ *holds,*
$g[\![Px]\!]h$ *does not hold, if either* $g \neq h$ *or (*$g = h$ *and* $\|Px\|^g$ *does not hold).*

The clause specifying when the expression does not hold is of course just the expected "otherwise" clause, the reason for splitting the clause up is to stress that there are two reasons for the formula not to hold, either the static condition is not satisfied, or the dynamic condition. Combining (17) with (16) gives the translation of the (weakly-) negated predicate.

(18) $g[\![-Px]\!]h$ *holds if either* $g \neq h$ *or (*$g = h$ *and* $\|Px\|^g$ *does not hold).*
$g[\![-Px]\!]h$ *does not hold, if* $g = h$ *and* $\|Px\|^g$ *holds.*

Because the weak negation's only action is to invert when the formula does and does not hold, the conditions under which the weakly negated formula is true are rather strange: One situation in which the weakly negated formula holds is when the input and output states are different.

That this is really not what we want can be seen if we start with a predicate $\overline{P}$ in PL that is the complement of of P in PL. First negate the predicate in static logic ($g\|\overline{P}(x)\|$ is true iff $g\|Px\|$ is false), and then make this dynamic by substituting it in (17):

(19) $g[\![\overline{P}x]\!]h$ *holds if* $g = h$ *and* $\|Px\|^g$ *does not hold,*
$g[\![\overline{P}x]\!]h$ *does not hold, if either* $g \neq h$ *or (*$g = h$ *and* $\|Px\|^g$ *holds).*

As you can see (18) and (19) are not equivalent, they differ for their respective $g \neq h$ clauses. That $[\![-Px]\!] \neq [\![\overline{P}(x)]\!]$ shows that negation already gives the wrong result for atomic formulas. We don't have to wait for the existential quantifiers to get us into trouble, we are in it from the beginning

The form this problem takes does suggest its own solution. The two versions of the negated predicate only differ where the two assignments are different. That an assignment is in the output set just because it is non-equal to the input assignment is a side effect of the chosen way of representing formulas as relations. It has nothing to do with the interpretation of an atomic formula as a a filter that throws away certain possibilities. These particular assignments are totally irrelevant to the actual predicate that is supposed to test them.

The best way to to understand this is to return to the transitions-metaphor, and re-inspect the explanation given above of the definition of predication in DPL. Formulas characterize a transition between an input-state and an output-state. In the case of predicates, the definition consists of two parts. A condition of the form $g = h$, which characterizes which transitions there are, and a clause of the form $g(x) \in \mathcal{I}(P)$, characterizing which of these transitions are true transitions.

$$g \xrightarrow{Px} h \text{ iff } g = h,$$

$$g \xrightarrow[\oplus]{Px} g \text{ iff } g(x) \in \mathcal{I}(P)$$

Transitions are written as an arrow, and of these, the true transitions are labeled with a "$\oplus$" sign. Every now and then, it makes sense to consider those transitions that are *not* true transitions. In that case I will write this as:

$$g \xrightarrow[\ominus]{Px} g \text{ iff } g(x) \notin \mathcal{I}(P)$$

A transition labeled with a "$\ominus$" sign.

No longer are there transitions between every pair of assignments, the set of transitions corresponding to a formula is a proper subset of the set of pairs of states. Of these transitions some are true (marked with a $\oplus$) and some false (marked with a $\ominus$). Given a formula ϕ, specifying for what input g there is a transitions to output h is saying for what h, $g[\![\phi]\!]h$ is defined. Similarly, specifying for what input g there is a true transition to output h means saying for what h, $g[\![\phi]\!]h$ is true.

The next thing to do is to redefine the other components of the logic along the same lines, and see how the resulting logic looks. I will use $[\![\phi]\!]^+$ to denote the true transitions that correspond to ϕ and $[\![\phi]\!]^d$ to denote the transitions according to ϕ (d for defined). I will also sometimes use $[\![\phi]\!]^-$, defined by

$$g[\![\phi]\!]^-h := g[\![\phi]\!]^dh \text{ and not } g[\![\phi]\!]^+h$$

to denote valid transitions that are not true. Below I will often define expressions by not only giving a definition of $[\![\phi]\!]^+$ and $[\![\phi]\!]^d$, but also specifying the resulting expression for $[\![\phi]\!]^-$ for the reader's convenience.

2.2 A Simple 3-valued Dynamic Predicate Logic

We have to redefine the other logical constants to fit the 3-valued logic, but this is not too difficult.

A similar argument as for the predication holds for the existential quantifier: $g[\![\varepsilon]\!]^+h$ should express that h is made out of g by changing the value for x, without changing any of the other values. Random assignment does not have any test-like properties. It therefore does not make sense for it to be false, an output state is either relevant, and in that case it makes the random assignment true, or irrelevant. In other words, every transition is a true transition:

(20) $g \xrightarrow{\varepsilon_x} h$ iff $g \approx_x h$,

$g \xrightarrow[\oplus]{\varepsilon_x} h$ iff $g \approx_x h$,

For the other logical operators we reason as follows. We do not want to loose information that *is* relevant, but we do want to exclude all irrelevant information. This means that if a formula ϕ is defined in terms of sub-formulas, any transition corresponding to ϕ should be defined in terms of the transitions corresponding to the sub-formulas.

What we are looking for is the strict extension of the logical operators from 2 to 3 valued logic. The definition that this leads to is the following.

For negation we add a clause saying that $-\phi$ is defined if (and only if) ϕ is. In terms of the transitions, this is to say that the signs of the transitions switch places.

(21) $g \xrightarrow{-\phi} h$ iff $g \xrightarrow{\phi} h$

(22) $g \xrightarrow[\oplus]{-\phi} h$ iff $g \xrightarrow{\phi} h$ but not $g \xrightarrow[\oplus]{\phi} h$

$-\phi$ defined if ϕ is, $-\phi$ is true if ϕ is false.

The definition for the conjunction is possibly even simpler. A transition for $\phi \wedge \psi$ is the composition of transitions of ϕ and ψ; a true transition for $\phi \wedge \psi$ is the composition of true transitions of ϕ and ψ:

(23) $g \xrightarrow{\phi\wedge\psi} h$ iff $\exists k\ g \xrightarrow{\phi} k \xrightarrow{\psi} h$

$g \xrightarrow[\oplus]{\phi\wedge\psi} h$ iff $\exists k\ g \xrightarrow[\oplus]{\phi} k \xrightarrow[\oplus]{\psi} h$

It may not be immediately obvious when $g \xrightarrow[\ominus]{\phi\wedge\psi} h$ holds. In fact, this splits up in two components, first of all, it should be a transition, and second it should not be a true condition. Writing this out we get:

(24) $g \xrightarrow[\ominus]{\phi\wedge\psi} h$ iff $\neg\exists k\ g \xrightarrow[\oplus]{\phi} k \xrightarrow[\oplus]{\psi} h$ and $\exists m\ (g \xrightarrow[\oplus]{\phi} m \xrightarrow[\ominus]{\psi} h$ or

$g \xrightarrow[\ominus]{\phi} m \xrightarrow[\oplus]{\psi} h$ or $g \xrightarrow[\ominus]{\phi} m \xrightarrow[\ominus]{\psi} h)$

At least one of the components of any transition has to be false and a (valid) transition should exist.

Combining all this we arrive at a definition of an alternative for DPL. This definition has a similar form to that of its static counterpart [1:2.5]. It is given is by first specifying for a formula when it is defined, and then when it is true. Although not really necessary, I also add the clause for falsity for ease of reference, this will of course always be the difference of the defined and the true clause ($[\![\phi]\!]^- = [\![\phi]\!]^d - [\![\phi]\!]^+$).

2.1 Definition (DPL*)
The Strictly-partial modification of Dynamic Predicate Logic DPL* *has the same syntax as* DPL, *with* $-\phi$ *replacing* $\sim\phi$. *Its semantics is defined as follows:*

Aside 2.1 Conjunction Is a Dynamic Weak Kleene Conjunction

To see what the content of the definition of conjunction is, we can look at what this definition means for the conjunctions of something really simple: tests. For this we only have to consider outputs that are identical to inputs. Inspection of the $g[\![\phi \wedge \psi]\!]g$ shows that this conjunction is in fact a very familiar one: $g[\![\phi \wedge \psi]\!]^d g$ holds if $g[\![\phi]\!]^d g$ and $g[\![\psi]\!]g$ holds; $g[\![\phi \wedge \psi]\!]^+ g$ holds if $g[\![\phi]\!]^+ g$ holds and $g[\![\psi]\!]^+ g$ holds. This is the weak Kleene conjunction ([1:2.5]).

$\wedge$	$\top$	$\bot$	$\star$
$\top$	$\top$	$\bot$	$\star$
$\bot$	$\bot$	$\bot$	$\star$
$\star$	$\star$	$\star$	$\star$

(a) $g[\![Px_1 \dots x_n]\!]^d h$ if $g = h$
$g[\![Px_1 \dots x_n]\!]^+ h$ if $g = h$ and $\|Px_1 \dots x_n\|^g$
$g[\![Px_1 \dots x_n]\!]^- h$ if $g = h$ and not $\|Px_1 \dots x_n\|^g$

(b) $g[\![x = y]\!]^d h$ if $g = h$
$g[\![x = y]\!]^+ h$ if $g = h$ and $g(x) = g(y)$
$g[\![x = y]\!]^- h$ if $g = h$ and $g(x) \neq g(y)$

(c) $g[\![\phi \wedge \psi]\!]^d h$ if $\exists k\ (g[\![\phi]\!]^d k$ and $k[\![\psi]\!]^d h)$
$g[\![\phi \wedge \psi]\!]^+ h$ if $\exists k\ (g[\![\phi]\!]^+ k$ and $k[\![\psi]\!]^+ h)$
$g[\![\phi \wedge \psi]\!]^- h$ if $\neg \exists k\ (g[\![\phi]\!]^+ k$ and $k[\![\psi]\!]^+ h)$ and $\exists k\ ((g[\![\phi]\!]^- k$ and $k[\![\psi]\!]^+ h)$ or
$(g[\![\phi]\!]^+ k$ and $k[\![\psi]\!]^- h)$ or $(g[\![\phi]\!]^- k$ and $k[\![\psi]\!]^- h)$

(d) $g[\![\varepsilon_x]\!]^d h$ if $g \approx_x h$
$g[\![\varepsilon_x]\!]^+ h$ if $g \approx_x h$
$g[\![\varepsilon_x]\!]^- h$ never

(e) $g[\![-\phi]\!]^d h$ if $g[\![\phi]\!]^d h$
$g[\![-\phi]\!]^+ h$ if $g[\![\phi]\!]^- h$
$g[\![-\phi]\!]^- h$ if $g[\![\phi]\!]^+ h$

(f) $g[\![!\phi]\!]^d h$ if $g = h$ and $\exists k\ g[\![\phi]\!]^d k$
$g[\![!\phi]\!]^+ h$ if $g = h$ and $\exists k\ g[\![\phi]\!]^+ k$
$g[\![!\phi]\!]^- h$ if $g = h$ and $\exists k\ g[\![\phi]\!]^d$ and $\neg \exists k\ g[\![\phi]\!]^+ k$

Proposition 2.2 will show that $g[\![!\phi]\!]g$ is defined for every g. Note that always $[\![\phi]\!]^+ \subseteq [\![\phi]\!]^d$ and, by definition, $[\![\phi]\!]^d = [\![\phi]\!]^+ \cup [\![\phi]\!]^-$.

Reasoning about this logic is reasonably easy, because it essentially consists of two DPL-like logics on top of each other, $[\![\dots]\!]^d$ and $[\![\dots]\!]^+$. For example, just as in the case of DPL, the associativity of the conjunction follows from the associativity of relational composition:

$g[\![(\phi \wedge \psi) \wedge \chi]\!]^d h$ iff $g[\![\phi \wedge (\psi \wedge \chi)]\!]^d h$ and $g[\![(\phi \wedge \psi) \wedge \chi]\!]^+ h$ iff $g[\![\phi \wedge (\psi \wedge \chi)]\!]^+ h$

And consequently, we do have the same dynamic binding property:

$g[\![(\varepsilon_x \wedge \phi) \wedge \psi]\!]^d h$ iff $g[\![\varepsilon_x \wedge (\phi \wedge \psi)]\!]^d h$ and $g[\![(\varepsilon_x \wedge \phi) \wedge \psi]\!]^+ h$ iff $g[\![\varepsilon_x \wedge (\phi \wedge \psi)]\!]^+ h$

As in the case of DPL, there is, next to the above dynamic conditions, a canonical[1] definition for the truth-conditions:

(25) $\|\phi\|^g$ is **defined** if $\exists h \;\; g[\![\phi]\!]^d h$
$\|\phi\|^g$ is **true** if $\exists h \;\; g[\![\phi]\!]^+ h$
$\|\phi\|^g$ is **false** if $\exists h \;\; g[\![\phi]\!]^d h$ and $\neg \exists h \;\; g[\![\phi]\!]^+ h$

These truth conditions lead to an important proposition

2.2 Proposition[2]
Any formula ϕ of DPL is defined for any assignment g, i.e. the truth conditions $\|\phi\|^g$ of the formulas of DPL* are two-valued.*

This proposition shows that in DPL*, the transitions may be partialized, but the truth-conditions are still total.

All this raises a question. If DPL* has similar dynamic properties as DPL, and has total truth conditions, how different is it from DPL?

2.3 A Comparison between DPL and DPL*

Comparing formulas ϕ of DPL* and formulas ϕ' of DPL can be done using different levels of scrutiny. Between formulas of partial logic, the notions strong and weak equivalence are defined. Furthermore, the dynamics adds a third notion: static equivalence.

2.3 Definition (Strong, Weak, and Static Equivalence (Internal))
Let ϕ and ϕ' be formulas of DPL. Two expressions are said to be **strongly equivalent**, if they have exactly the same values for all inputs and outputs.*

$$\phi \equiv^s \phi' \text{ if for every } g, h (g[\![\phi]\!]^+ h \Leftrightarrow g[\![\phi']\!]^+ h) \text{ and } (g[\![\phi]\!]^d h \Leftrightarrow g[\![\phi']\!]^d h).$$

---

1. Canonical in the sense that it is possible to define a notion of normal binding form similar to the definition for DPL in chapter 1.

2. The proof is by induction over the complexity of the formula. The base cases are as follows: Predicates are defined for every output that is the same as the input, and $\|Px\|^g$ is true whenever $g(x) \in \mathcal{I}(P)$, and false whenever $g(x) \in \mathcal{I}(P)$. The case of n-ary predicates is similar. Random assignments are always true, because there is always some output. For the conjunction $\|\phi \wedge \psi\|^g$ we reason as follows. By induction, $\|\phi\|^g$ will be defined, which means that we can find a k such that $g[\![\phi]\!]^d k$, then again by induction, $\|\psi\|^k$ will be defined, i.e. there will be an h such that $k[\![\psi]\!]^d h$, hence there will be an h such that $g[\![\phi \wedge \psi]\!]^d h$. In the case of negation and closure, $\| - \phi\|^g$ and $\|!\phi\|^g$ are defined whenever $\|\phi\|^g$ is (which, by induction, it always is).

*Two expressions are said to be **weakly equivalent**, if they are true for the same inputs and outputs.*

$$\phi \equiv^w \phi' \text{ if for every } g, h(g[\![\phi]\!]^+ h \Leftrightarrow g[\![\phi']\!]^+ h).$$

*Two expressions ϕ and ϕ' are **statically equivalent**, if they have the same truth conditions: $\|\phi\|$ and $\|\phi'\|$ are identical.*

$$\phi \equiv^0 \phi' \text{ if for every } g(\exists h \;\; g[\![\phi]\!]^+ h \Leftrightarrow \exists h \;\; g[\![\phi']\!]^+ h).$$

In general, static equivalence is not as trivial as it seems, because in a three-valued logic, we have to consider the possibility that the static truth values might be undefined.

These notions of equivalence can be extended to equivalences between formulas of different logical systems. In particular, I will use them in a moment to compare DPL with three-valued dynamic logics like DPL*. This can be done by considering DPL a logic that is potentially three valued, but just happens to never result in the third, $\star$, value.

2.4 Definition (Strong, Weak, and Static Equivalence (External))
Let ϕ' be a formula of DPL and ϕ' of DPL. Then ϕ' is **strongly equivalent** with ϕ if they are true on the same pairs of assignments and false on the same pairs of assignment.*

$$\phi \equiv^s \phi' \text{ if for every } g, h(g[\![\phi]\!] h \Leftrightarrow g[\![\phi']\!]^+ h) \text{ and always } g[\![\phi']\!]^d h.$$

*Two expressions are said to be **weakly equivalent**, if they are true for the same outputs and inputs.*

$$\phi \equiv^w \phi' \text{ if for every } g, h(g[\![\phi]\!] h \Leftrightarrow g[\![\phi']\!]^+ h).$$

*Two expressions ϕ and ϕ' are **statically equivalent**, if they have the same truth conditions: $\|\phi\|$ and $\|\phi'\|$ are identical.*

$$\phi \equiv^0 \phi' \text{ if for every } \; g(\exists h \;\; g[\![\phi]\!] h \Leftrightarrow \exists h \;\; g[\![\phi']\!]^+ h).$$

Given that DPL* formulas are never defined for all pairs, no DPL* formula will ever by equivalent to a DPL formula.

It turns out that DPL* contains a sub-logic that is weakly equivalent to DPL. The only thing missing to see this is to give a definition of the DPL-negation within DPL*: a negation that closes off the context. The following shows that the DPL* expression $-!\phi$ defines a negation that is weakly equivalent to $\sim$ in DPL.

2.5 Lemma[3]
If ϕ_{dpl} is weakly equivalent with ϕ_*, $\sim\phi_{\text{dpl}}$ is weakly equivalent with $-!\phi_*$

Using this lemma, we can show that DPL* has a sub-logic that is weakly equivalent with DPL. Let $^\circ$: DPL $\rightarrow$ DPL* be the function that maps every formula on the syntactically equivalent formula of DPL*. It is defined as follows

2.6 Definition (Embedding of DPL into DPL*)
The mapping $^\circ$:DPL$\rightarrow$DPL* *is defined by:*

$$\begin{array}{ll} Px_1 \ldots x_n^\circ & := Px_1 \ldots x_n \\ (\phi \wedge \psi)^\circ & := (\phi)^\circ \wedge (\psi)^\circ \\ (\varepsilon_x)^\circ & := \varepsilon_x \\ (\sim\phi)^\circ & := -!(\phi)^\circ \end{array}$$

Only the negation step is non-trivial. We can now prove the following proposition:

2.7 Proposition[4]
If ϕ is a formula of DPL, *then ϕ is weakly equivalent with ϕ°.*

One corollary of this is that any formula ϕ will be true exactly when ϕ° is. Everything formalized in DPL can be formalized in DPL$^\circ$, resulting in the same predictions. Because of this, I will from now on use the image DPL$^\circ$ in DPL* when I discuss DPL, rather than use the original interpretation in two-valued logic. This proposition shows that this does not matter for the truth values.

But doing that would not explain why sometimes a more dynamic negation has to be chosen. In the following section, I will define a negation that has the same truth conditions as the DPL one for all the standard examples, but is at the same time dynamic in a similar way to the weak dynamic negation.

2.4 Strengthening Negation

To see why weak negation will not do, I first show what goes wrong when we use it. Weak negation is not very useful as a negation to translate natural language expressions, because it allows too much. For example, the following holds:

2.8 Fact[5]
If ϕ does not contain ε_x as a subformula, $-(\varepsilon_x \wedge \phi)$ and $(\varepsilon_x \wedge -\phi)$ are strongly equivalent.

---

3. $g[\![\sim\phi_{\mathrm{dpl}}]\!]h$ iff ($g = h$ and for no k: $g[\![\phi_{\mathrm{dpl}}]\!]k$). By the assumption this is equivalent to ($g = h$ and for no k: $g[\![\phi_*]\!]^+k$) which by definition equals ($g = h$ and not $g[\![!\phi_*]\!]^+g$). By two valuedness ($g[\![!\phi_*]\!]^dg$) this is equal to ($g = h$ and $g[\![!\phi]\!]^-g$) which is equal to $g[\![-!\phi]\!]^+h$.

4. The proof is by induction on the complexity of the formulas.
For the atomic case: $g[\![Px_1 \ldots x_n]\!]h$ iff $g = h$ and $g(x) \in \mathcal{I}(P)$, iff $g[\![Px_1 \ldots x_n]\!]^+h$ iff $g[\![Px_1 \ldots x_n^\circ]\!]^+h$; $g[\![\varepsilon_x]\!]h$ iff $g \approx_x h$ iff $g[\![\varepsilon_x^\circ]\!]h$. For the conjunction: $g[\![\phi \wedge \psi]\!]h$ iff $\exists k$ ($g[\![\phi]\!]k$ and $k[\![\psi]\!]h$)
iff (by ind.) $\exists k$ ($g[\![\phi^\circ]\!]^+k$ and $k[\![\psi^\circ]\!]^+h$) iff $g[\![\phi^\circ \wedge \psi^\circ]\!]^+h$. For the negation the result follows from lemma 2.5.

Written as a quantifier this has the form $[\![-\exists x\phi]\!] = [\![\exists x - \phi]\!]$. One “negation” that behaves this way is so called *denial*:

(26) *It was not a man who came in, it was a woman.*

However, denial is completely different from other forms of negation. Although it might be possible that weak negation can help in understanding denial, it falls outside of the scope of this dissertation.

In DPL this problem does not occur, because the negation states that there is no true output state, effectively quantifying over output-states. The proof shows that it makes essential use of the fact, proven in lemma 1:2.6, that the output state (h) assigns to x the value actually chosen by ε_x. If ϕ does contain ε_x the variable selected by the first ε_x will be erased by the second, and the proof won't go through. For example

2.9 Fact[6]
$-(\varepsilon_x \wedge (Px \wedge \varepsilon_x))$ *and* $(\forall x(-Px) \wedge \varepsilon_x)$ *are strongly equivalent.*

This leaves us in the following situation: DPL negation ($\sim\phi$) exhibits the right truth conditions, it complementizes the truth-values, but it is a static negation. Weak dynamic negation *does* have the required dynamic effect of complementing the dynamic effect, but it results in the wrong static truth values.

The next step will be to define a negation operator $\neg$ that has the right truth conditions *and* the right dynamic conditions. It turns that a suitable negation can be defined by simply adding these two conditions together, and demanding that the negation is a complementizer both for the (3-valued) dynamics and for the static truth values.

---

5. Defined part: $g[\![-(\varepsilon_x \wedge \phi)]\!]^d h$ iff $g[\![\varepsilon_x \wedge \phi]\!]^d h$ iff $\exists k\ \ g[\![\varepsilon_x]\!]^d k$ & $k[\![\phi]\!]^d h$ iff $\exists k\ \ g[\![\varepsilon_x]\!]^d k$ and $k[\![-\phi]\!]^d h$ iff $g[\![\varepsilon_x \wedge -\phi]\!]^d h$.
Truth part: use lemma 2.6 in chapter 1. we have $g[\![-(\varepsilon_x \wedge \phi)]\!]^+ h \iff g[\![-(\varepsilon_x \wedge \phi)]\!]^d h$ & not $g[\![\varepsilon_x \wedge \phi]\!]^+ h$ (def.) $\iff$ ($g[\![\varepsilon_x]\!]^d g_h^\phi$ and $g_h^\phi[\![\phi]\!]^d h$) & not ($g[\![\varepsilon_x]\!]^+ g_h^\phi$ or $g_h^\phi[\![\phi]\!]^+ h$) (by lemma 2.6).
Now we know $g[\![\varepsilon_x]\!]^+ g_h^\phi = g[\![\varepsilon_x]\!]^d g_h^\phi$ so ($g[\![\varepsilon_x]\!]^+ g_h^\phi$ or $g_h^\phi[\![\phi]\!]^+ h$) can only be false if $g_h^\phi[\![\phi]\!]^+ h$ is false. We know $g_h^\phi[\![\phi]\!]^d h$ holds, so ($g_h^\phi[\![\phi]\!]^d h$ & not $g_h^\phi[\![\phi]\!]^+ h$) $= g_h^\phi[\![-\phi]\!]^+ h$ holds.
Putting this together with $g[\![\varepsilon_x]\!]^+ g_h^\phi$ gives $g[\![\varepsilon_x \wedge -\phi]\!]^+ h$.

6. The proof uses the obvious fact, that if I change the value of x and then again, I could just as well have done it once. This means the first change can be hidden under a closure without doing anything to the net result.
Defined part: $g[\![-(\varepsilon_x(\wedge Px \wedge \varepsilon_x))]\!]^d h$
iff $\exists d\ \ g[x := d][\![Px]\!]^d g[x := d]$ & $\exists e\ \ g[x := d][\![\varepsilon_x]\!]^d g[x := d][x := e]$ & $g[x := d][x := e] = h$
iff $\exists k\ \ g[\![\varepsilon_x]\!]^d k$ & $k[\![Px]\!]^d k$ & $\exists e\ \ g[\![\varepsilon_x]\!]^d g[x := e]$ & $g[x := e] = h$
iff $g[\![-!(\varepsilon_x(\wedge Px \wedge \varepsilon_x))]\!]^d h$.
Truth part: $g[\![-(\varepsilon_x(\wedge Px \wedge \varepsilon_x))]\!]^+ h$ iff defined and not $g[\![\varepsilon_x(\wedge Px \wedge \varepsilon_x)]\!]^+ h$
iff not $\exists d\ \ g[x := d][\![Px]\!]^+ g[x := d]$ & $\exists e\ \ g[x := d][\![\varepsilon_x]\!]^+ g[x := d][x := e]$ & $g[x := d][x := e] = h$
iff not $\exists k\ \ g[\![\varepsilon_x]\!]^+ k$ & $k[\![Px]\!]^+ k$ & $\exists e\ \ g[\![\varepsilon_x]\!]^+ g[x := e]$ & $g[x := e] = h$ iff $g[\![-!(\varepsilon_x(\wedge Px \wedge \varepsilon_x))]\!]^+ h$.

Dynamically, the minimal condition on a negation is to demand that if the negated formula $\neg(\phi)$ is true for some output, the original formula ϕ is false for that output, and if the negated formula $\neg(\phi)$ is false for some output, the original formula ϕ is true for that output. One consequence of this is that if a negated formula $\neg(\phi)$ is defined for some output h, the original formula ϕ should be defined for the output h. This is expressed by the following

2.10 Definition (**Dynamic Conditions on Negation**)
For all g and h:

$$g[\![\neg(\phi)]\!]^{+}h \Longrightarrow g[\![\phi]\!]^{-}h$$

$$g[\![\neg(\phi)]\!]^{-}h \Longrightarrow g[\![\phi]\!]^{+}h$$

I used $g[\![\neg(\phi)]\!]^{-}h$, the expression for the relation not holding, instead of the somewhat longer $g[\![\neg(\phi)]\!]^{d}h \;\&\; \neg g[\![\neg(\phi)]\!]^{d}h$. I will use this notation, because in the discussion of the negation it makes sense to keep our eyes directly focussed on the $+$ and $-$ parts, which are the parts complementation directly deals with. **Statically**, the minimal conditions on a negation is to demand that if $\|\neg(\phi)\|^{g}$ is true for some assignment g then $\|\phi\|^{g}$ is false, and if $\|\neg(\phi)\|^{g}$ is false for some assignment g then $\|\phi\|^{g}$ is true.

2.11 Definition (**Static Conditions on Negation**)
For all g:

$$\|\neg(\phi)\|^{g} = \mathbf{true} \Longrightarrow \|\phi\|^{g} = \mathbf{false}$$

$$\|\neg(\phi)\|^{g} = \mathbf{false} \Longrightarrow \|\phi\|^{g} = \mathbf{true}$$

We can give this a similar form to (25) by substituting the definition of the (static) truth values (25) into this:

(27) $\exists h\ \ g[\![\neg(\phi)]\!]^{+}h \Longrightarrow (\exists h\ \ g[\![\phi]\!]^{d}h \;\&\; \neg\,\exists h\ \ g[\![\phi]\!]^{+}h)$

(28) $(\exists h\ \ g[\![\neg(\phi)]\!]^{d}h \;\&\; \neg\,\exists h\ \ g[\![\neg\phi]\!]^{+}h) \Longrightarrow \exists h\ \ g[\![\phi]\!]^{+}h$

The weakest operator satisfying both constraints is defined by by combining the conditions the operator should satisfy and replacing $\Rightarrow$ by iff :

$$g[\![\neg(\phi)]\!]^{+}h \text{ iff } (g[\![\phi]\!]^{-}h \;\&\; \exists k\ \ g[\![\phi]\!]^{d}k \;\&\; \neg\,\exists k\ \ g[\![\phi]\!]^{+}k)$$

$$g[\![\neg(\phi)]\!]^{-}h \text{ iff } (g[\![\phi]\!]^{+}h \;\&\; \exists k\ \ g[\![\phi]\!]^{+}k)$$

If we now remove from these the two sub-formulas that do not contribute a constraint ($\exists k\ \ g[\![\phi]\!]^d k$ in the first and $\exists k\ \ g[\![\phi]\!]^+ k$ in the second clause[7]) and replace $(g[\![\phi]\!]^- h\ \ \&\ \ \neg\exists k\ \ g[\![\phi]\!]^+ k)$ by the equivalent $(g[\![\phi]\!]^d h\ \&\ \neg\exists k\ \ g[\![\phi]\!]^+ k)$ this results in:

2.12 Definition (strong dynamic negation)
The strong dynamic negation $\neg\phi$ of a formula is defined by:

$$g[\![\neg(\phi)]\!]^+ h \ \text{ iff } \ (g[\![\phi]\!]^d h\ \&\ \neg\exists k\ \ g[\![\phi]\!]^+ k)$$

$$g[\![\neg(\phi)]\!]^- h \ \text{ iff } \ g[\![\phi]\!]^+ h$$

Note the similarity with the definition of negation in DPL:

$$g[\![{\sim}\phi]\!] h \ \text{ iff } \ \ g = h \text{ and } \neg\exists k\ \ g[\![\phi]\!] k.$$

The DPL definition consists of a dynamic condition $g = h$, expressing what are the transitions, and a static condition $\neg\exists k\ (g[\![\phi]\!] k$ expressing what are the true transitions. The definition of $\neg\phi$ has the same static condition, but a different dynamic condition $(g[\![\phi]\!]^- h)$. This similarity gives rise to a sometimes useful alternate form for the definition of strong negation in terms of the static and weak negation:

$$g[\![\neg(\phi)]\!]^+ h \ \text{ iff } \ (g[\![-\phi]\!]^+ h\ \&\ g[\![{\sim}\phi]\!]^+ g)$$

$$g[\![\neg(\phi)]\!]^- h \ \text{ iff } \ g[\![-\phi]\!]^- h$$

Because of the similarity between the negations $\sim$ and $\neg$, they behave in comparable ways if the dynamic effect of $\neg$ is not used. For example, the following holds

2.13 Lemma[8] *$\neg\phi$ and ${\sim}\phi$ are statically equivalent.*

I end with the definition of the resulting logic:

2.14 Definition (The Logic DPL$^\neg$)
DPL$^\neg$ is the result of adding the operator $\neg$ to DPL*.

Adding $\neg$ to the logic does not change the nature of the truth conditions, as shown by:

2.15 Proposition[9]
Any formula of DPL$^\neg$ is defined for every assignment g. I.e. the truth conditions $g\|\phi\|$ of the formulas of DPL$^\neg$ are two-valued.

---

7. If $g[\![\phi]\!]^- h$ then $g[\![\phi]\!]^d h$, so $\exists k\ \ g[\![\phi]\!]^d k$ and if $g[\![\phi]\!]^+ h$ then $\exists k\ \ g[\![\phi]\!]^+ k$.

8. $\exists h\ \ g[\![{\sim}\phi]\!]^+ h \iff \exists h\ \ g[\![-!\phi]\!]^+ g\ \&\ g = h.\, g[\![-!\phi]\!]^+ g \iff \exists h\ \ g[\![!\phi]\!]^- h \iff \exists h\ \ g[\![\phi]\!]^d h\ \&\ \neg\exists k\ \ g[\![\phi]\!]^d k \iff \exists h\ \ g[\![\neg\phi]\!] h.$
$\exists h\ \ g[\![{\sim}\phi]\!]^- h \iff \exists h\ \ g[\![-!\phi]\!]^- g\ \&\ g = h.\, g[\![-!\phi]\!]^- g \iff g[\![!\phi]\!]^+ g \iff \exists h\ \ g[\![\phi]\!]^+ h \iff \exists h\ \ g[\![\neg\phi]\!]^- h.$

9. The proof is an extension of that for DPL*. We only have to add a proof for $\neg$, which follows from lemma 2.13. For any ϕ, we know ${\sim}\phi$ is two-valued, but then, by 2.13, so is $\neg\phi$.

2.5 A Comparison between DPL and DPL$^\neg$

Because DPL$^\neg$ contains DPL* as a proper sub-logic, the translation procedure of DPL into DPL* also works for DPL$^\neg$. But a more interesting embedding of DPL into DPL$^\neg$ can be found. I will show that in a large number of interesting cases, we can replace $\sim\phi$ by $\neg\phi$ without changing the truth-conditions. Before I can give this, I need to prove some facts about formulas. The following lemma states some facts about the relation between statically equivalent formulas that occur in some conjunction.

2.16 Lemma[10] *The following hold for any ϕ, ϕ' such that $\phi \equiv^0 \phi'$.*
(i) For any ψ: $(\psi \wedge \phi) \equiv^0 (\psi \wedge \phi')$.
(ii) For ψ not containing any free variables that are bound by either ϕ or ϕ': $(\phi \wedge \psi) \equiv^0 (\phi' \wedge \psi)$.

It turns out that for phenomena discussed using DPL it actually does not matter to the truth conditions whether $\sim$ or $\neg$ is used as negation. For example, the following holds:

2.17 Fact[11] $(\sim(\phi \wedge \sim\psi)) \equiv^0 (\neg(\phi \wedge \neg\psi))$.

The DPL-translation of the classic donkey sentence *if a farmer owns a donkey he beats it* (cf. Groenendijk and Stokhof (1991)) is an instantiation of this general form:

$$\sim((\varepsilon_x \wedge \mathbf{f}(x) \wedge \varepsilon_y \wedge \mathbf{d}(y) \wedge \mathbf{o}(x,y)) \wedge \sim(\mathbf{b}(x,y)))\,.$$

If we now replace every occurrence of $\sim$ by $\neg$, we get a statically equivalent formula:

$$\neg((\varepsilon_x \wedge \mathbf{f}(x) \wedge \varepsilon_y \wedge \mathbf{d}(y) \wedge \mathbf{o}(x,y)) \wedge \neg(\mathbf{b}(x,y)))\,.$$

All formal considerations aside, it obvious why this is true: the extra dynamic possibilities that $\neg$ offers are not used, so why would they matter?

It turns out that *all* sentences, that when translated into DPL result in the correct interpretations, are of a form that allows us to replace all subformulas of the form $\sim\phi$ by $\neg\phi$ without changing the truth conditions. This is not as surprising as it may seem. According to DPL-conventions, a translation of a sentence containing a free variable is a marked (discourse-ungrammatical) sentence. For example, because the universal quantifier closes off its scope, the pronouns *he* and *it* in

---

10. (i) $\exists h\ g[\![\psi \wedge \phi]\!]^+h \iff \exists h,k\ g[\![\psi]\!]^+k\ \&\ k[\![\phi]\!]^+h \iff \exists k\ g[\![\psi]\!]^+k\ \&\ \exists h\ k[\![\phi]\!]^+h \iff \exists k\ g[\![\psi]\!]^+k\ \&\ \exists h\ k[\![\phi']\!]^+h$ (ass.) $\iff \exists h,k\ g[\![\psi]\!]^+k\ \&\ k[\![\phi']\!]^+h \iff \exists h\ g[\![\psi \wedge \phi']\!]^+h$. The proof for $[\![\psi \wedge \phi]\!]^d$ is identical.
(ii) Let $x_1, \ldots, x_n$ be the variables bound by ϕ, and $x_{n+1}, \ldots, x_{n+m}$ the variables bound by ϕ'. Because ψ does not mention these variables, $g[\![\psi]\!]^+h \iff g'[\![\psi]\!]^+h'$ for $g' = g\overrightarrow{[x_i := d_i]}$ and $h' = h\overrightarrow{[x_i := d_i]}$ (a standard locality principle). Now suppose for $\exists k,h\ (g[\![\phi]\!]^+k\ \&\ k[\![\psi]\!]^+h)$. by assumption, $\exists l\ g[\![\phi]\!]^+l$. But l can only differ from k in the variables that ϕ and ϕ' change, so $k[\![\phi]\!]h \iff k'[\![\phi]\!]^+h'$ for some h' that assigns the same values to $x_1, \ldots, x_{n+m}$ that k' does. hence $\exists k,h\ (g[\![\phi']\!]^+k\ \&\ k[\![\psi]\!]^+h) \iff \exists h\ (g[\![\phi' \wedge \psi]\!]^+h)$. The proof for $[\![\psi \wedge \phi]\!]^d$ is identical.

11. By lemma 2.13, $\sim\psi \equiv^0 \neg\psi$. By lemma 2.16.(i) $(\phi \wedge \sim\psi) \equiv^0 (\phi \wedge \neg\psi)$. By applying lemma 2.13 on both sides, $\sim(\phi \wedge \sim\psi) \equiv^0 \neg(\phi \wedge \sim\psi)$ and $\sim(\phi \wedge \neg\psi) \equiv^0 \neg(\phi \wedge \neg\psi)$, therefore finally $\sim(\phi \wedge \sim\psi) \equiv^0 \neg(\phi \wedge \neg\psi)$.

(29) *Every farmer owns a donkey. *He beats it.*

are translated as free variables, which explains, according to DPL, the markedness of the second sentence. As we already discussed in the first chapter, sentences like this do often have a well defined meaning, and whether sentences like this are indeed incorrect is doubtful. However, assume for the moment that this is true. Then we can turn this around. A sentence that DPL judges as correct will get an interpretation that does not contain free variable. If it does not, replacing $\sim\phi$ by $\neg\phi$ will not change the truth conditions. Quantifiers inside ϕ will not bind any new variable-occurrences, because such occurrences would have been free variables before. Hence the dynamic contribution of $\neg\phi$ is not used and the truth conditions stay the same.

This informal argument can be made more precise

2.18 Fact[12] (Traditional Formulas)
Let a traditional formula be a formula ϕ of DPL *such that ϕ does not contain any free variables and let $\phi^\bullet$ be the result of replacing all occurrences of $\sim$ by $\neg$, then $\phi \equiv^0 \phi^\bullet$.*

This shows that we do not need closed negation to discuss the standard examples. However, it also shows that when we do use $\neg\phi$, we lose the incorrectness predictions of DPL. No longer can we rely on free variables to flag incorrectness of a sentence in a given context. Only after a long detour, will we be able to give exactly the same predictions, where appropriate, in chapter 4. But there the incorrectness will make essential use of the distinction between singular and plural referents, and claim that the second sentence of (29) is doubtful because a singular pronoun tries to pick up a plural-valued referent, an attempt that is prone to fail.

3 Partializing the Truth Conditions

Introduction

So far, I was careful to partialize the transitions only so far that the truth-conditions are still two-valued. I showed that by partializing the logic, a more dynamic definition of negation can be given, while the traditional examples that are interpreted using dynamic logic are still given the same meaning. In doing so, I only gave variants to the operators that are defined in DPL. The new freedom given by the notion of partialized defined transitions allows us to define new operators, that do not preserve the two-valued behavior. The first operators one might think of in such a case are the partial operators discussed in section 2.5 of chapter 1. But I start with another class of operators that to a certain degree could just as well have been defined in the original DPL, and that is the class of parallel operators.

12. The proof, the most important steps of which are proved by lemma 2.16, is left to the reader.

3.1 Static Boolean Operators

The basic conjunction of dynamic logic is a serial conjunction. We saw that when two sentences are interpreted

(30) *A man walked in the park. He whistled.*

the first sentence is interpreted in the current information state, and then the second sentence is interpreted in that state as changed by the first sentence. Composition, then, is the basic form of combining sentences, and the corresponding combination of formulas is the serial conjunction defined in DPL.

From a more formal view point, at least one other conjunction suggests itself. A conjunction in which both conjuncts are interpreted in the same information state, and their output states combined afterwards. The following defines such a conjunction (as always, I add the $[\![\ldots]\!]^-$ clause for clarity):

3.1 Definition (Static Conjunction)
static conjunction is defined by

(31) $g[\![\phi \cap \psi]\!]^d h$ iff $g[\![\phi]\!]^d h$ and $g[\![\psi]\!]^d h$
$g[\![\phi \cap \psi]\!]^+ h$ iff $g[\![\phi]\!]^+ h$ and $g[\![\psi]\!]^+ h$
$g[\![\phi \cap \psi]\!]^- h$ iff $g[\![\phi \cap \psi]\!]^d h$ and $(g[\![\phi]\!]^- h$ or $g[\![\psi]\!]^- h)$

This conjunction demands that there are independent ϕ-transitions and ψ-transitions leading from the same input-state to the same output state. As transitions the two conjunctions compare as follows:

serial conjunction $\wedge$	*parallel conjunction* $\cap$
$g \overset{\phi}{\rightarrow} k \overset{\psi}{\rightarrow} h$	$g \underset{\psi}{\overset{\phi}{\rightrightarrows}} h$
$g \overset{\phi}{\oplus\!\!\rightarrow} k \overset{\psi}{\oplus\!\!\rightarrow} h$	$g \underset{\psi}{\overset{\phi}{\oplus\!\!\rightrightarrows}} h$

This operator $\cap$ is part of its own boolean family, In particular, we can define a static disjunction by $\phi \cup^- \psi := -(-\phi \cap -\psi)$. This definition can be written out as

(32) $g[\![\phi \cup \psi]\!]^d h$ iff $g[\![\phi]\!]^d h$ and $g[\![\psi]\!]^d h$
$g[\![\phi \cup \psi]\!]^+ h$ iff $g[\![\phi]\!]^+ h$ or $g[\![\psi]\!]^+ h$
$g[\![\phi \cup \psi]\!]^- h$ iff $g[\![\phi \cup \psi]\!]^d h$ and $(g[\![\phi]\!]^- h$ and $g[\![\psi]\!]^- h)$

Both $\cap$ and $\cup$ are very useful as auxiliary operators. They will not have a direct use, because in natural language, utterances always have a definite order, and the dynamic conjunction will always be more appropriate.

Let us now try to pursue the definition of new additional operators somewhat more systematically.

Aside 2.2 Program Disjunction is not a Reasonable Operator

The operator $\cup$ defined here is not the program disjunction that is often written with the same symbol. That operator is defined as follows:

$$g[\![\phi \oplus \psi]\!]^d h \iff g[\![\phi]\!]^d h \text{ or } g[\![\psi]\!]^d h$$

$$g[\![\phi \oplus \psi]\!]^+ h \iff g[\![\phi]\!]^+ h \text{ or } g[\![\psi]\!]^+ h$$

The problem with this operator is, that the output state need only be a defined output state for one of the disjuncts. An example due to Visser (1995a), illustrates this best. In $(\varepsilon_x \oplus \varepsilon_y) \wedge Pxy$, an output of ε_x will bind x but leave y free, and an output of ε_y will bind y but leave x free. Furthermore, they cannot bind simultaneously. Adding this operator will make the notions of free and bound variable non-welldefined. Surely a worrying development. In section 5, this operator will have a use as an auxiliary operator in comparing three valued dynamic logic with two valued dynamic logic from a more mathematical perspective.

3.2 Additional Unary Operators

The discussion in section 2 suggests two basic classes of unary operators on formulas: **closed operators** and **dynamic operators.** Φ is called closed when the equality of input and output is part of its definition. For closed operators, $g[\![\Phi(\phi)]\!]^d h$ can only hold if $g = h$. If the expressions $g[\![\Phi(\phi)]\!]^d h$ can also be true for outputs that are not identical to the input, the operator is said to be dynamic.

In chapter 1 a *strict* partial predicate logic was defined [1:2.5]. Finding a similar notion of strictness in the dynamic case gives some more problems. It turns out that there are (at least) two possible counterparts of definition 2.8 on page 27, one corresponding to dynamic operators, and one to closed operators:

(33) $g[\![\Phi(\phi)]\!]^d h \Rightarrow g[\![\phi]\!]^d h$ (dynamic strictness)

(34) $g[\![\Phi(\phi)]\!]^d h \Rightarrow (g = h \ \& \ \exists h \ \ g[\![\phi]\!]^d h)$ (static strictness)

Dynamic strictness is the most natural one: any $\Phi(\phi)$ transition has to be a ϕ transition. This corresponds directly with the static case: if $\Phi(\phi)$ is defined, ϕ has to be defined. Note that one consequence is that the operator Φ cannot contribute a dynamic effect of its own. The notion of static strictness is meant to deal with closed operators.

Dynamic Unary Operators

Table 2.1 lists all strict operators that can be written as a (set-theoretic) polynomial in terms of $[\![\phi]\!]$ alone. These are the same operators as those defined for strict-partial predicate logic (SPL) (these mirror definition 2.9 in chapter 1).

Some of the operators have names. **Weak dynamic negation** $-\phi$, which was discussed in full above, is the local complement; defined whenever ϕ is, and true whenever ϕ is false. **Presupposition** $+\phi$ is an operator that restricts definedness of a formula to the true part of it; $+\phi$ is defined if ϕ is true. **Definedness** $\mathcal{D}\phi$, does more or less the opposite; it is true whenever ϕ is defined.

The other operators can be defined in terms of these. In fact, if we use $\cup$, we can define $\mathcal{D}\phi := (\phi \cup -\phi)$, leaving us with two primitive unary operators, $-$ and $+$.

	operator $\Phi(\phi)$	$g[\![\Phi(\phi)]\!]^+h$ holds when	$g[\![\Phi(\phi)]\!]^d h$ holds when	name of operator
1	ϕ	$g[\![\phi]\!]^+h$	$g[\![\phi]\!]^d h$	***identity***
2	$-\phi$	$g[\![\phi]\!]^-h$	$g[\![\phi]\!]^d h$	***weak negation***
3	$\mathcal{D}\phi$	$g[\![\phi]\!]^d h$	$g[\![\phi]\!]^d h$	***definedness***
4	$-\mathcal{D}\phi$	*never*	$g[\![\phi]\!]^d h$	
5	$+\phi$	$g[\![\phi]\!]^+h$	$g[\![\phi]\!]^+h$	***presupposition***
6	$-+\phi$	*never*	$g[\![\phi]\!]^+h$	
7	$+-\phi$	$g[\![\phi]\!]^-h$	$g[\![\phi]\!]^-h$	
8	$-+-\phi$	*never*	$g[\![\phi]\!]^-h$	
9	$+-+\phi$	*never*	*never*	

Table 2.1: Unary operators.

The following fact about the *presupposition operator* is the counterpart of the proof in section 2.9 of chapter 1:

3.2 Fact[13]
If no variable bound by ϕ is also bound in ψ, $-(+\phi \wedge \psi) \equiv^s (+\phi \wedge -\psi)$

Note that this is one operator where we lose two-valuedness. If the logic contains a presupposition operator, not only the dynamics (the behavior of $g[\![\phi]\!]h$), but also the truth theory (the behavior of $\|\phi\|^g$) becomes partial. No longer need it be true for every g such that $\exists h\ \ g[\![\phi]\!]h$. This is of course to be expected from presuppositions. This operator will be used to great effect in the dynamic quantifier theory and, somewhat more surprisingly, in the definition of $\neg$ that I discuss next.

Dynamic Negation Revisited

The closure operator is the only important example of an operator that closes off the formula. Whatever the output of its argument is, it is thrown away and replaced by the input. The result is a test:

(35) $g[\![!\phi]\!]^d h$ iff $g = h$ and $\exists k\ \ g[\phi]^d k$,
$g[\![!\phi]\!]^+ h$ iff $g = h$ and $\exists k\ \ g[\phi]^+ k$,
$g[!\phi]^- h$ iff $g = h$ and $\exists k\ \ g[\phi]^d k$ and $\neg\, \exists k\ \ g[\phi]^+ k$.

13. The proof is similar to the proof for fact 2.8. Defined part: $g[\![-(+\phi \wedge \psi)]\!]^d h$ iff $g[\![+\phi \wedge \psi]\!]^d h$ iff $\exists k\ \ g[\![+\phi]\!]^d k$ & $k[\![\psi]\!]^d h$ iff $\exists k\ \ g[\![\phi]\!]^+ k$ & $k[\![\psi]\!]^d h$ iff $\exists k\ \ g[\![\phi]\!]^+ k$ & $k[\![-\psi]\!]^d h$ iff $g[\![+\phi \wedge -\psi]\!]^d h$
Truth Part: $g[\![-(+\phi \wedge \psi)]\!]^+ h$ iff $g[\![+\phi \wedge \psi]\!]^- h$ iff $g[\![+\phi \wedge \psi]\!]^d h$ & $\neg g[\![+\phi \wedge \psi]\!]^+ h$.
iff $g[\![+\phi \wedge -\psi]\!]^d h$ & $\neg\, \exists k\ \ g\| + (+\phi)\| k$ or $k[\![\psi]\!]^+ h$. iff $g[\![+\phi \wedge -\psi]\!]^d h$ & $\neg g[\![+\phi]\!]^+ g_h^\phi$ or $g_h^\phi[\![\psi]\!]^+ h$
iff $g[\![+\phi \wedge -\psi]\!]^d h$ & $g[\![+\phi]\!]^+ g_h^\phi$ and $\neg g_h^\phi[\![\psi]\!]^+ h$ iff $g[\![+\phi \wedge -\psi]\!]^d h$ & $g[\![+\phi]\!]^+ g_h^\phi$ and $g_h^\phi[\![-\psi]\!]^+ h$
iff $g[\![+\phi \wedge -\psi]\!]^d h$ & $g[\![+\phi \wedge -\psi]\!]^+ h$ iff $g[\![+\phi \wedge -\psi]\!]^+ h$.

Combining the closure operator with the operators $\cap$ and $+$, it is possible to give a definition of $\neg\phi$ in terms of other operators

3.3 Fact[14] $\neg\phi \equiv^s +-(!\phi \wedge -\phi) \cap -\phi$.

A simpler form, that almost works, is $-!\phi \wedge -\phi$. This is weakly equivalent to $\neg$ but not strongly, it is false for too many outputs: $g[\![-!\phi \wedge -\phi]\!]^- h$ holds for an output h if $g[\![-!\phi]\!]^- g$, i.e. if there is a k such that $g[\![\phi]\!]^+ k$, whether or not h makes $-\phi$ true or false.

The actual definition derives from observing exactly when this simpler one goes wrong. It goes wrong for outputs that make $(!\phi \wedge -\phi)$ true, and it goes right for outputs that make it false. So the simple thing to do is to restrict attention to those outputs that make this formula false. First we add $-$, to swap dynamic truth values, and then $+$, to restrict to the true case. This makes a formula that is defined exactly when $(!\phi \wedge -\phi)$ is false: $+-(!\phi \wedge -\phi)$. By making this a static conjunction with $-\phi$, the left conjunct will determine when the negation is defined, the right whether it is true or false for some input and output.

3.3 Additional Binary Operators

The two basic classes of binary operators, dynamic and static operators, satisfy *strictness* principles similar to the static principle of chapter 1 (definition 2.8):

(36) $g[\![\Phi(\phi,\psi)]\!]^d h \Rightarrow (\exists k\ \ g[\![\phi]\!]^d k \ \&\ k[\![\psi]\!]^d h)$ (dynamic strictness)

(37) $g[\![\Phi(\phi,\psi)]\!]^d h \Rightarrow (g[\![\phi]\!]^d h \ \&\ g[\![\psi]\!]^d h)$ (static strictness)

Dynamic strictness says that a transition for $\Phi(\phi,\psi)$ should consist of first doing a ϕ transition and then a ψ transition. Remember, transitions need not be true transitions: all dynamic operators $(\wedge, \vee, \rightarrow)$ satisfy this constraint. Static strictness says that a transition for $\Phi(\phi,\psi)$ should consist of simultaneously doing a ϕ transition and an ψ transition. These are the static boolean connectives ($\phi \cap \psi$, and operators defined in terms of it). And this might not even be the whole story. Krahmer (1995) defines a variation of the conjunction in a logical system called double-negation DRT, that for some outputs satisfies the dynamic strictness and for some others the static strictness (cf. page 75 for some more discussion of this.) Worse, the program disjunction operator $\oplus$ (cf. aside 3.1 on page 59) only satisfies a watered down version of strictness: only one of the disjuncts needs to be defined.

14. The proof is in two steps. First we observe that if both are defined, $-\phi$ and $\neg\phi$ have the same truth value. Then we prove that $+-(!\phi \wedge -\phi)$ is, if defined, always true because of the presupposition operator, and defined exactly for the cases that $\neg\phi$ is defined for.
The proof of this last statement proceeds as follows: $g[\![+-(!\phi \wedge -\phi)]\!]^+ h \iff g[\![(!\phi \wedge -\phi)]\!]^- h \iff$ one of the following three: (a) $\exists m\ \ g[\![!\phi]\!]^+ m[\![-\phi]\!]^- h$ or (b) $\exists m\ \ g[\![!\phi]\!]^- m[\![-\phi]\!]^+ h$ or (c) $\exists m\ \ g[\![!\phi]\!]^- m[\![-\phi]\!]^- h$ which become (a) $g[\![!\phi]\!]^+ g[\![\phi]\!]^+ h$ or (b) $g[\![!\phi]\!]^- g[\![\phi]\!]^- h$ or (c) $g[\![!\phi]\!]^- g[\![\phi]\!]^+ h$
of these, (a) is equivalent with $g[\![\phi]\!]^+ h$, (b) is equivalent with $\neg\exists m\ \ g[\![\phi]\!]^+ m \ \&\ g[\![\phi]\!]^- h$ and (c) can never be true. (a) + (b) form $(g[\![\phi]\!]^+ h$ or $(\neg\exists m\ \ g[\![\phi]\!]^+ m \ \&\ g[\![\phi]\!]^- h))$ which is equivalent with $g[\![\neg\phi]\!]^d h$. this proves the identity.

Of course, it also possible to mix static and dynamic operators. The result is that there are a large number of possible operators, even if we restrict ourselves to operators corresponding to one particular operator in standard predicate logic. For example, if we ask ourselves what the implication looks like in this framework, the first thing we may consider is to directly translate all the constituents into dynamic counterparts. This results in **strong dynamic implication**:

(38) $g[\phi \to \psi]h = g[\neg(\phi \wedge \neg\psi)]h$ ((strong) implication)

Proposition 2.13, which states that when the dynamic effect of $\neg$ is not used, $\neg$ and $\sim$ behave the same, shows that for all standard examples this operator will in fact result in similar predictions as those of the implication in DPL. In such cases $\|\neg(\phi \wedge \neg\psi)\|$ has the same truth values, though not the same dynamic properties, as $\|{\sim}(\phi \wedge {\sim}\psi)\|$.

A large number of alternative schemata for the implication can be given, exchanging one or both of $\neg$ by either $-$ or $\sim$, or even $\wedge$ by $\cap$. For example, $-(\phi \wedge -\psi)$, $-(\phi \wedge \neg\psi)$ or $\neg(\phi \wedge -\psi)$. It remains to be seen whether these alternative definitions have a place in semantical analyses of language where, as I argued in chapter 1, closure does not seem to be very natural and expressions are always given in a certain order. On the other hand, the definition of $\neg$ in terms of the operator $\cap$ shows that these operators do have use within the formal language.

The next thing I will do is introduce a third form of partialization. having partialized the relations between states and the truth value in a state (truth conditions), I will now partialize the states themselves. Although formally this amounts to even more logical possibilities, it also makes it possible to express certain natural constraints on the logical language. In particular, the condition that formulas never bind a variable twice, necessary for certain proofs, and the condition that information always increases in discourse, can be expressed once states are partialized.

4 Partializing the States

There are some properties of formulas that seem very intuitive, but cannot be formally expressed. The most important of these is *information increase*, most naturally expressed as: no formula removes information from the state. One reason that we cannot do this is that so far, the states of the logic are total assignments. This means that if a variable is given a new value, the old value gets lost, which means losing the information about that variable. The first half of this section redefines the logic discussed in the previous two sections. Formulas are interpreted as relations between partial assignments.

4.1 Partial States

To express the constraint that information should increase, we have to find a way of expressing whether a variable actually has been introduced in the preceding context or not. We can achieve this by considering (*finite*) partial assignments as states. In chapter 3 and 4, I will show that partial assignments can be used to great effect to explain a number of linguistic phenomena, especially phenomena involving plurals.

The following lists advantages of partializing states, starting with one we already offered[15].

1. One advantage of partial assignments is that it enables us to distinguish between variables given a value in the preceding text, for which the assignments are defined, and variables not yet bound, for which the assignments are not defined.
2. At the start of a discourse, no variables have been introduced. Using partial assignments we can uniquely express this unique initial (totally undefined) state, as the partial assignment that does not assign any values to variables.
3. Starting from the totally undefined state, updating states incrementally will always result in assignments that only give values to a finite number of variables. Therefore states involved in actual discourse will always be finite objects.
4. Partial assignments admit a natural order of information increase. If we start with an assignment, and then make a new one out of it by keeping the values already defined fixed and add some new variables, we get an assignment that encodes more information. This can then be used to express the property that formulas only increase information.

Besides these four properties, there are two more, mentioned here for completeness.

5. In chapters 3 and 4, it will be shown, that partial assignments are necessary to implement a notion of subset in a plural logic which preserves the relations that the individual elements have with other sets.
6. Finally, it turns out that partialization of both truth-values and assignments is a necessary ingredient for the definition of a dynamic type-logic. However, the discussion of this argument is left to another place (van den Berg (1996a)).

In the following, the following notation will be used

4.1 Definition (Partial Assignments)
A partial assignment g is a function that maps any variable x of the language onto $E \cup \{\star\}$, where $\star$ is not an element of E. If $g(x) = \star$, $g(x)$ is said to be undefined, *and g is said to be* undefined for x. *The assignment that assigns $\star$ to all variables is called the* initial state *or the* initial assignment *and written as* $\mathbf{1}$. *The definition of equivalence upto a variable is the same as always:*

$$g \approx_x h \text{ iff } g(y) = h(y) \text{ for all } y \neq x.$$

A partial assignment h is said to **extend** *g, if for all x, $g(x) = \star$ or $g(x) = h(x)$. In that case h assigns the same values to the same variables as g, and is defined for some more variables. This is written as follows*

(39) $g \preceq h$ iff $\forall x \; g(x) = \star$ or $g(x) = h(x)$. *(assignment extension)*

15. Similar observations have been made by Beaver (1995), Fernando (1994b) and Krahmer (1995), amongst others.

Redefining the above dynamic logic to be able to deal with partial assignments is not difficult. The only cases we really have to reconsider are the atomic cases, because only these depend on the actual form of the assignments. The complex cases are defined inductively in terms of simpler ones, and do not depend on the actual structure of the assignments.

4.2 Atomic Actions

There are two classes of atomic actions in dynamic predicate logic: predicates applied to variables, which are tests, and random assignments, which introduce new variables. It will turn out, that the change in the definition for the interpretation of predicates is minimal. But the case of the random assignments is a different story. It turns out that there is a choice to be made, and we will have to make up our mind as to which is the most appropriate one.

Predicates

The definition of the predicate case (Px) has to take the possibility into account, that x need not be assigned a value in a given context g. When this is the case, the predicate cannot be said to be true or false: the undefined value does not have properties. If $g(x) = \star$, $g[\![Px]\!]g$ has to be undefined. This leads to the following definition, which is essentially the same as before, except for the check on the existence of all the arguments. Again I add a $[\![\ldots]\!]^-$-clause for clarity:

4.2 Definition

Predicates are defined as before, except that their arguments are required to exist.

(40)
$$g[\![Px_1\ldots x_n]\!]^d h \text{ iff } g = h \text{ and } g(x_1) \neq \star \text{ and } \ldots \text{ and } g(x_n) \neq \star$$
$$g[\![Px_1\ldots x_n]\!]^+ h \text{ iff } g[\![Px_1\ldots x_n]\!]^d h \text{ and } {<}g(x_1),\ldots,g(x_n){>} \in \mathcal{I}(P)$$
$$g[\![Px_1\ldots x_n]\!]^- h \text{ iff } g[\![Px_1\ldots x_n]\!]^d h \text{ and } {<}g(x_1),\ldots,g(x_n){>} \notin \mathcal{I}(P)$$

Note that this means, that formulas that contain free variables will now by definition be undefined. Only closed sentences, i.e. formulas in which all variables have been assigned a value, have a denotation; open sentences, i.e. formulas with one or more free variables do not.

Random Assignments

Whereas for predicates there is an obvious redefinition, the case of random assignments is not so straightforward. There are some choices to be made here. First, it has to be decided whether it will be allowed to have a random assignment assign a new value to a variable that has already been given a value by a preceding random assignment (rebinding of the variable). In the case of total assignments, we have no choice but to overwrite the values of variables with new values, because all assignments assign values to all variables. If we decide to keep it that way, the DPL definition (page 20) can be simply copied.

4.3 Definition (DPL Random Assignment)

DPL Random Assignment η' introduces a new value to a variable without inspecting the old value (if there is any).

(41) $g[\![\eta'_x]\!]^d h$ if $g \approx_x h$
$g[\![\eta'_x]\!]^+ h$ if $g \approx_x h$
$g[\![\eta'_x]\!]^- h$ never

Notice that one of the values that the output h can assign to x is the undefined value $\star$.

This choice goes a bit against the spirit of taking partial assignments, which, amongst other things, were introduced to encode which variables are still free for binding and which variables are already in use. The alternative is not to allow assigning new values to variables that already have a value assigned to them, and only allow introduction of new values for new variables. But this still admits for two different approaches. The first possibility is to allow random assignments to bind a variable that is already in use, but only to the value it already has.

4.4 Definition (Safe Random Assignment)
Safe Random Assignment η introduces a new value to a variable in cases where the variable is new, and leaves the value alone in cases where there already is one.

(42) $g[\![\eta_x]\!]^d h$ if $g \approx_x h$ and ($g(x) = \star$ or $g(x) = h(x)$)
$g[\![\eta_x]\!]^+ h$ if $g[\![\eta_x]\!]^d h$
$g[\![\eta_x]\!]^- h$ never

This definition was used in van den Berg (1990), and similar a one was used by van der Does (1993).

The effect of this definition is that it disables any subsequent attempt to quantify over an already introduced variable. The first time that a random assignment is used in a formula it functions as a normal assignment, but the next time it doesn't do anything at all. This means that you can translate both definites and indefinites as introducing random assignments. For example *A man entered. he smiled* translates as $(\eta_x \wedge \mathbf{m}(x) \wedge \mathbf{e}(x)) \wedge (\eta_x \wedge \mathbf{s}(x))$, which is equivalent to $(\eta_x \wedge \mathbf{m}(x) \wedge \mathbf{e}(x) \wedge \mathbf{s}(x))$, the second η_x has no effect. The translation of *he smiled*, $(\eta_x \wedge \mathbf{s}(x))$, may at first not seem too unreasonable. However, to have indefinites and definites (pronouns) translate the same way seems a bit counter intuitive. And in chapter 1 it was shown that under certain circumstances a definite needs to be able to introduce its own referent, so in that case another way of picking up old referents have to be defined anyway (cf. the discussion of example 6 on page 9).

For this reason, I choose a third definition of random assignment, called **(guarded) random assignment**. This encodes the fact that a variable introduced has to be new. it is undefined if a variable is introduced that already has been given a value by a preceding random assignment.

4.5 Definition ((Guarded) Random Assignment)
Guarded random assignment (ε) only assigns a value if the variable is not yet in use.

(43) $g[\![\varepsilon_x]\!]^d h$ if $g \approx_x h$ and $g(x) = \star$
$g[\![\varepsilon_x]\!]^+ h$ if $g[\![\varepsilon_x]\!]^d h$
$g[\![\varepsilon_x]\!]^- h$ never

I will from now on refer to this as *the* random assignment. Note, that like with the two earlier random assignments, $g[\![\varepsilon_x]\!]^+g$ is allowed. This also means that

4.6 Fact[16]
the following facts hold:
(i) $(\varepsilon_x \wedge \varepsilon_x) \equiv^s \varepsilon_x$
(ii) $g[\![\varepsilon_x \wedge Px \wedge \varepsilon_x]\!]^+h$ *is always undefined.*
(iii) *if* $g[\![\varepsilon_x \wedge x = x]\!]^+h$ *then* $h(x) \neq \star$
(iv) *if* $g[\![\varepsilon_x]\!]^+h$, *then* $g[\![\eta'_x]\!]^+h$ *and* $g[\![\eta_x]\!]^+h$.

It might be considered an unfortunate choice to allow random assignments to fail to introduce a referent. The reason this is chosen as the primitive definition is that, in light of the last fact (4.6.iii), random assignments that are certain to introduce a defined element can easily be defined.

4.7 Definition (Effective Random Assignments)
Random assignments that are guaranteed to introduce a defined value are defined by.
$\bar{\eta}'_x := \eta'_x \wedge x = x$
$\bar{\eta}_x := \eta_x \wedge x = x$
$\bar{\varepsilon}_x := \varepsilon_x \wedge x = x$

Because under normal circumstances, an introduced variable will be used as an argument of a predicate, which will also demand definedness, this seems to be somewhat superfluous.

The reason why I prefer ε is the following. in the light of fact 4.6.iv: when a formula that only contains ε as random assignment is defined, we can substitute either η' or η for ε at will. This shows that provided use formulas that only bind any variable once, the choice of random assignment is arbitrary. However, if we *do* use ε, definedness of the formula will guarantee that variables are bound only once.

4.3 Undefinedness of Transitions and of States

The different notions of "partial" (and corresponding notion of "defined") deserve some special attention. It is important to realize that the three ways of partializing the logic really are different. As shown in sections 2 and 3, there is a difference between partial dynamic conditions (which might still have total truth conditions), and partial truth conditions. In this section the states are partialized, adding a third form of partiality.

---

16. (i) $g[\![\varepsilon_x \wedge \varepsilon_x]\!]^+h \iff \exists k\ (g[\![\varepsilon_x]\!]^+k\ \&\ k[\![\varepsilon_x]\!]^+h)$. Now $g \approx_x k$, and the definition of ε_x demands that $k(x) = \star$, i.e. $g = k$. $(g[\![\varepsilon_x]\!]^+g\ \&\ g[\![\varepsilon_x]\!]^+h) \iff g[\![\varepsilon_x]\!]^+h$.
(ii) $g[\![\varepsilon_x \wedge Px \wedge \varepsilon_x]\!]^+h \iff \exists k\ (g[\![\varepsilon_x]\!]^+k\ \&\ k[\![Px]\!]^+l\ \&\ l[\![\varepsilon_x]\!]^+h) \iff \exists k\ (g[\![\varepsilon_x]\!]^+k\ \&\ k[\![Px]\!]^+k\ \&\ k[\![\varepsilon_x]\!]^+h)$ (predicates are tests). By the definition of predicates, $k(x) \neq \star$ for $k[\![Px]\!]^+k$ to hold and $k(x) = \star$ for $k[\![\varepsilon_x]\!]^+h$ to hold, which can never be the case. Hence the conjunction is always undefined.
(iii) $g[\![\varepsilon_x \wedge x = x]\!]h \iff \exists k\ (g[\![\varepsilon_x]\!]k\ \&\ k[\![x = x]\!]h) \iff \exists k\ (g[\![\varepsilon_x]\!]k\ \&\ k(x) \neq \star\ \&\ k(x) = k(x)\ \&\ k = h) \implies k(x) \neq \star$.
(iv) $g[\![\varepsilon_x]\!]^+h \iff (g(x) = \star\ \&\ g \approx_x h)$. Then $g \approx_x h$, hence $g[\![\eta'_x]\!]^+h$, and $g \approx_x h\ \&\ (g(x) = \star$ or $g(x) = h(x))$, hence $g[\![\eta_x]\!]^+h$.

The three notions are connected. If a variable x is not given a value by g, there will not be a $g[\![Px]\!]^d g$ transition, and $\|Px\|^g$ will be undefined. However, as we already saw, $g[\![\varepsilon_x]\!]g$ may be defined, even though $g(x)$ is not.

4.4 Full Dynamic Predicate Logic

None of the other definitions of dynamic operators pose any special problems because they are all defined inductively over simpler cases. Now we have the atomic actions of the dynamic logic in place, we can copy the definitions for the other operators from the case of logic over total assignments. The definitions are as given in 44–51 (Note that I did not try to find a smallest set of operators). Because definitions for these operators have been given before, be it for total states, I list them here in a slightly different format, as relations, to give an alternative perspective. I will also not explicitly list the $[\![\phi]\!]^-$ interpretation.

(44) $[\![Px_1 \ldots x_n]\!]^d := \{<g, h> \mid g = h \;\&\; g(x_1) \neq \star \;\&\; \ldots \;\&\; g(x_n) \neq \star\}$
$[\![Px_1 \ldots x_n]\!]^+ := \{<g, h> \mid g = h \;\&\; <g(x_1), \ldots, g(x_n)> \in \mathcal{I}(P)\}$

(45) $[\![\varepsilon_x]\!]^d := \{<g, h> \mid g \approx_x h \;\&\; g(x) = \star\}$
$[\![\varepsilon_x]\!]^+ := [\![\varepsilon_x]\!]^d$

(46) $[\![-\phi]\!]^d := [\![\phi]\!]^d$
$[\![-\phi]\!]^+ := [\![\phi]\!]^d - [\![\phi]\!]^+$

(47) $[\![+\phi]\!]^d := [\![\phi]\!]^+$
$[\![+\phi]\!]^+ := [\![\phi]\!]^+$

(48) $[\![\phi \wedge \psi]\!]^d := [\![\phi]\!]^d \circ [\![\psi]\!]^d$
$[\![\phi \wedge \psi]\!]^+ := [\![\phi]\!]^+ \circ [\![\psi]\!]^+$

(49) $[\![\phi \cap \psi]\!]^d := [\![\phi]\!]^d \cap [\![\psi]\!]^d$
$[\![\phi \cap \psi]\!]^+ := [\![\phi]\!]^+ \cap [\![\psi]\!]^+$

(50) $[\![!\phi]\!]^d := \{<g, g> \mid \exists k \;\; g[\![\phi]\!]^d k\}$
$[\![!\phi]\!]^+ := \{<g, g> \mid \exists k \;\; g[\![\phi]\!]^+ k\}$

(51) $[\![\neg\phi]\!]^d := [\![+ - (!\phi \wedge -\phi)]\!]^+$
$[\![\neg\phi]\!]^+ := [\![\neg\phi]\!]^d \cap [\![-\phi]\!]^+$

Any logic that uses these operators or a subset of these is a strict-partial dynamic predicate logic. But not all operators are as reasonable as others. I now define the sub-logic that is linguistically interesting (as proved by its applications described in later chapters).

4.8 Definition ($\mathrm{FDPL_s}$)
Full dynamic predicate logic (for singulars) is the dynamic logic that only sees the operators

$$\{\wedge, \neg, +, !, \{\varepsilon_x\}_{x \in \mathrm{VAR}}\}$$

As other dynamic predicate logics, $\mathrm{FDPL_s}$ does not have constants.

The choice of operators is not only guided by practical considerations: these are exactly the operators that are strict, and behave reasonably as operations on processes (cf. section 5.2).

Conclusion

This finishes the discussion of the logical language proposed as an alternative to DPL. As the propositions and lemmas show, despite the fact that FDPL is a rather different logic, the actual predictions coincide with those of DPL on the standard cases.

5 Logical Issues

Introduction

The purpose of this final section, and of its counterparts in the next two chapters, is to identify a number of themes, issues and questions which arise naturally when we look at our semantic system for natural language from a purely logical point of view. None of the discussions here are final or complete. Rather, they serve to wet the appetite.

First, let us summarize the formal arguments in favor of a three-valued approach to dynamic semantics. I have observed that tests are naturally three-valued: allowing for genuine outcomes 'true' and 'false' if the current state does not change, and being 'undefined' if it does. Then also, variable shifts ε_x seemed either 'successful' or 'inappropriate' (rather than 'failed'). Thus, I decided to work with a three-valued version of the two-valued relational algebra underlying the original version of dynamic predicate logic. Note that this is a partialization in the superstructure of semantic actions, so to speak, which can happen even if the underlying computational states remain total. A fact witnessed by the distinction between partializing dynamic effect in section 2 and partializing truth values in section 3. The resulting formal system may be viewed as a joint generalization of relational algebra and partial logic. As it happens, this is not a simple merge: many interesting questions arise.

Relational Algebra

Let me summarize the role which is played by Relational Algebra in the workings of DPL. First, as I observed, many core examples of dynamic semantic equivalences amount to general truths concerning relational composition and/or strong negation. For instance, associativity of composition $((A \circ (B \circ C)) = ((A \circ B) \circ C))$ underlies the dynamic scope of indefinites in discourse (a similar point was already made by Barwise (1987), concerning the semantics of noun phrases). Likewise, the truth conditions of implication required an algebraic calculation using the more complex valid law $\sim(A \circ \sim\sim B) = \sim(A \circ B)$. Much is known about the relevant features of this Relational Algebra (cf. Németi 1991, Németi 1993, van Benthem (1996)). For instance, even though relational algebra as a whole is undecidable, the fragment with just the operators $\{\circ, \sim\}$ is decidable. One way of seeing this is by embedding this fragment into Propositional Dynamic Logic (Harel 1981, Goldblatt 1987), using the simple valid equivalence $\langle \sim A \rangle \, \phi \Leftrightarrow \neg \langle A \rangle \top \ \& \ \phi$. And the latter system is decidable. Moreover, we know that this particular vocabulary is natural from a semantic point of view, as $\{\circ, \sim\}$ (in conjunction with Boolean union) define precisely those algebraic operations which which are 'safe for bisimulation' in the sense of van Benthem (1993). (I shall define these notions below.) That is, they fit naturally with what is arguably the most natural notion of process equivalence from the computational literature. As we will see, the decidability property is preserved in the move to partial dynamics, and the bisimulation facts also have a natural counterpart in partial logic.

Partial Logic

The other logical ingredient in our semantic system is (three-valued) Partial Logic. There is an extensive literature on variants of propositional logic with new partial operators in such a setting. These operators give rise to new semantic phenomena, such as 'persistence' of definite truth values (true, false) (cf. Langholm (1988), Visser (1985a), Muskens (1996)) as well as issues of semantic complexity (Thijsse (1992)) and proper axiomatization (Jaspers (1994)). I shall not attempt to survey this literature here. It may suffice to note that there exist finite functionally complete repertoires of (persistent) partial operators — and that the basic systems of partial propositional logic remain decidable. One way of seeing this is via effective translations into classical propositional logic (cf. Feferman 1984, van Benthem 1986b, Fenstad 1996).

Against this background, what should count as an appropriate 'generalization' of standard two-valued relational algebra? One way of achieving this is as follows. The operations of composition and strong negation have definitions which may be written inside predicate logic:

$$A \circ B \iff \lambda xy.\exists z.(xAz \ \& \ zBy)$$

$$\sim A \iff \lambda xy.(x = y \ \& \ \neg\exists z.xAz)$$

But then, the latter definitions can be interpreted in a standard three-valued version of predicate logic themselves. For instance, then, 'truth' of the positive part of a composition $x(A \circ B)^+y$ will mean that $\exists z.(xA^+z \ \& \ zB^+y)$ while 'falsity' will mean that $\forall z.(xA^-z$ or $zB^-y)$. However, this definition does not lead to a reasonable logic. For example, for tests xPx, interpreted as tests, the interpretation of P^+ has to be that part of the diagonal relation such that P holds, For the negative part, there are two reasonable interpretations: P^- can either be the complement inside the diagonal, or the complement relative to all pairs. Now consider $(P \circ P)$, then $(P \circ P)^+$ is $\exists z(xPz \ \& \ zPx)$ which, for tests, is equivalent to xP^+x.

But now look at $(P \circ P)^-$. If the first choice is taken, xP^-y is only defined if $x = y$, because the negative extension lies on the diagonal. Then, for any $z \neq x$, xP^-z is undefined, so

$\forall z(xP^{-}z \text{ or } zP^{-}x)$ is never defined, which is different from $xP^{-}x$. In fact, as soon as xAz is undefined for even one value, a conjunction with it will always be either true, or undefined.

This leaves the second choice of taking the whole complement. But that is not different from what happens in the original two-valued logic. The conjunction will then also result in a two valued result for any conjunction of two-valued components. You might think, that this gives a reasonable definition for three valued dynamic logic, because we can always add operators that introduce partiality. You might even think it is a good idea that the standard elements, tests and conjunction, are identical to a two-valued system. But as I argued in the text, it does not make sense that the identity process 1 (the relation that holds exactly on the diagonal) would be false if the input is different from the output. In a three-valued logic, it should be undefined.

For this reason, I chose a slight different definition. This definition is given in the natural form of a strict-partial logic. I do not define when the formula is 'true' and 'false', I define when it is 'defined' at all, and within that range, when it is 'true'.

$$x(A \circ B)^{d}y \iff \exists z.(xA^{d}z \;\&\; zB^{d}y),$$
$$x(A \circ B)^{+}y \iff \exists z.(xA^{+}z \;\&\; zB^{+}y),$$

the negative part is a derived definition: the defined-part minus the true-part:

$$x(A \circ B)^{-}y \iff x(A \circ B)^{d}y \;\&\; \neg x(A \circ B)^{+}y$$

This has interesting consequences consequences for consideration about the corresponding programs.

5.1 Predicate Logic

In addition to general considerations, some results are the result of peculiarities of the predicate logical language. Both total and partial versions of DPL are not a general relational algebra, but a special a very specialized theory of variable assignment and quantification, formulated in relational terms. One question we could ask ourselves, is whether this particular theory of relations validates relational identities that would not hold in general relational algebra. This question is raised by van Benthem (1996) (cf. also Cepparello (1995)). In the context of possible dynamic negations, Visser (1995b) gives the answer that no new laws get validated.

Many important properties depend on the specific form of predicate logical formulas. One is the existence of *normal binding forms*. For any DPL formula there is a strongly equivalent formula that has the same truth value in DPL as it has when read as a formula of predicate logic. This means, that (partial) DPL is also of interest as generalization of (partial) predicate logic.

A number of questions arise naturally.

axiomatization What is a complete axiomatization for the different variants of DPL defined here. In particular, it would be useful to be able to replace the semantic arguments used in this chapter by simpler algebraic calculations. The problem is that because the dynamics make long distance effects possible, it is difficult to find good local properties. For example, we saw (fact 3.2) that the rule

$$\frac{-(+\phi \wedge \psi)}{(+\phi \wedge -\psi)}\ (\mathbf{bvar}(\phi) \cap \mathbf{bvar}(\phi) = \emptyset)$$

has a constraint involving the variables bound by the two conjuncts.

Definability One thing left to be done is analyze the relationships between the different logical systems defined above. For example, is it really impossible to define $\neg$ in terms of the operators of DPL*, or $\cap$ in terms of the other operators? There is strong suspicion that DPL* weaker is than DPL$^\neg$, and that this in its turn is weaker than FDPL.

5.2 Bisimulation

In addition to this algebraic, 'formal language' approach, there is the semantic one. In this chapter, I introduced operators where the need arose, coming up with what seem the more obvious definitions, given the constraints of strict-partial logic. It is also possible to proceed top-down, and give some general logical constraints and ask what the natural operators are that satisfy these principles. Cf. (Thijsse 1992) for a discussion of the unary propositional case.

An important notion is that of *safety for bisimulation*. this is the notion that expresses that when the same basic facts hold on a model and all basic processes behave the same, then the logical operators should behave the same on those models (cf. (van Benthem 1993) and (van Benthem et al. 1993)) for details).

We can easily generalize the modal notion of 'bisimulation' to the current setting. Indeed, Thijsse (1992) has a proposal for bisimulations in partial modal propositional logic, being essentially a binary relation between states in two models leaving all three truth values at correlated states the same, and satisfying the usual zigzag clauses for the binary relations. In my case, I also have to partialize those relations

As in the earlier definitions, I will give the definitions in terms of defined and true. But, for this section alone, I will not restrict attention to strict-partial logic, but for a moment consider the whole of partial dynamic logic. To be able to also talk about the non-strict part of partial logic, we introduce two extra logical operators

The first is a notion of program disjunction already mentioned in aside 3.1 and repeated here:

(52) $g[\![\phi \oplus \psi]\!]^d h \iff g[\![\phi]\!]^d h$ or $g[\![\psi]\!]^d h$

$g[\![\phi \oplus \psi]\!]^+ h \iff g[\![\phi]\!]^+ h$ or $g[\![\psi]\!]^+ h$

The other is strong negation

(53) $g[\![\sim^s \phi]\!]^d h$ always

$g[\![\sim^s \phi]\!]^+ h$ iff $g = h \ \&\ \neg\, \exists k\ g[\![\phi]\!]^+ h$

Note the following properties that will be of use later:

5.1 Fact

(i) *for every ϕ, the relation $\perp_\phi$, defined by*

(54) $\perp_\phi := \sim^s \sim^s \phi \wedge \sim^s \phi$

is the universally false relation, which means that it is false for every input-output pair: $g[\![\sim^s \sim^s \phi \wedge \sim^s \phi]\!]^d h$ *always holds,* $g[\![\sim^s \sim^s \phi \wedge \sim^s \phi]\!]^+ h$ *never,*

(ii) *Any formula ϕ is completely determined by giving the two parts $\mathcal{D}\phi$ and $+\phi$:*

(55) $[\![\phi]\!]^d = [\![(\perp_\phi \wedge \mathcal{D}\phi) \oplus +\phi]\!]^d,$

$$[\![\phi]\!]^+ = [\![(\perp_\phi \wedge \mathcal{D}\phi) \oplus +\phi]\!]^+,$$

(iii) *Static negation $\sim$ can be defined in terms of other operators:*

(56) $\sim\phi := +(\perp_\phi \wedge \sim^s \sim^s \phi) \oplus + \sim^s \phi.$

(i) gives us something that we can be sure is false. Because $\perp_\phi$ is the same relation for any ϕ, I will normally omit the subscript ϕ and write $\perp$. (ii) gives a definition of any three-valued expression ϕ in terms of two expressions $\mathcal{D}\phi$ and $+\phi$ that are effectively two-valued (true or undefined). This definitions will be used in a moment to apply results of two valued dynamic logic to three valued dynamic logic. Note how this definition works. The left hand side is false if the right hand side is true, because $+\phi$ implies $\mathcal{D}\phi$, so there are three cases. ϕ is true iff $+\phi$ is true, hence the whole disjunction is true, if ϕ is false, $+\phi$ is undefined and $\mathcal{D}\phi$ true, hence $(\perp_\phi \wedge \mathcal{D}\phi)$ is false and the total disjunction is false. If ϕ is undefined, so are $+\phi$ and $\mathcal{D}\phi$ and hence the disjunction is undefined. Finally, (iii) defines $\sim$ in terms of $\sim^s$.

The definition of bisimulation is simply two copies of the standard two valued definition, because $[\![\phi]\!]^d$ and $[\![\phi]\!]^+$ are two copies of standard DPL on top of each other, together with operators ($+$ and $\mathcal{D}$) to switch between the two.

5.2 Definition (Dynamic Model and Partial Bisimulation)

A ***dynamic model*** *is a pair $<S, \{R_a \mid a \in A\}>$, with S a set of states and $\{R_a \mid a \in A\}$ a family of relations.*

Given two dynamic models *$<S, \{R_a \mid a \in A\}>$ and $<S', \{R'_a \mid a \in A\}>$, then C is a partial bisimulation iff*

- *1 if gCg', then g, g' validate the same atomic propositions,*
- *2a if gCg' and $gR^+_a h$, then there is a h' s.t. $g'R^+_a h'$ and hCh',*
- *2b if gCg' and $g'R^+_a h'$, then there is a h s.t. $gR^+_a h$ and hCh',*
- *3a if gCg' and $gR^d_a h$, then there is a h' s.t. $g'R^d_a h'$ and hCh',*

3b if gCg' and $g'R^d_a h'$, then there is a h s.t. $gR^d_a h$ and hCh',

As you can see, this does indeed say that something is a bisimulation for the three-valued relation R, iff it is a simultaneous bisimulation for the two two-valued relations R^+ and R^d. Note that there is an interesting logic internal consequence of this definition: C is a bisimulation for R iff it is a bisimulation for $+R$ and $\mathcal{D}R$. Because $+R$ is true (and defined) iff R^+ holds and $\mathcal{D}R$ is true (and defined) iff R^d holds, so calculating the bisimulation for either of these corresponds to either (2a–b) or (3a–b) of the above definition, but not the other.

I will sometimes have use for saying that C is a bisimulation of the negative extension of the formula, which means that the above applies to $[\![\phi]\!]^-$: if sCs' and $sR^-_a t$, then there is a t' s.t. $s'R^-_a t'$ and tCt', and vice versa. Note that if C is a bisimulation for both R^+ and R^-, then it is also one for $R^d = R^+ \cup R^-$, because union of relations results is a safe operation (van Benthem 1993). However, the other way around is not true, if C is a bisimulation for R^d and R^+ there is no general reason why it should be one for $R^- = R^d - R^+$.

This does mean, that the alternative possibility, saying that there is a bisimulation between one model and another when there are independent bisimulations for the true and false extensions of the relation does not result in the same set of bisimulations. I will return to this at the end of this discussion. for the moment, I will concentrate on this definition. The notion of safety is now defined as for the two valued case.

5.3 Definition (Safety for Partial Bisimulation)
An operation $\Phi(\phi_1, \ldots, \phi_n)$ is ***safe for partial bisimulation*** *if, whenever C is a partial bisimulation for the transitions $[\![\phi_1]\!], \ldots, [\![\phi_n]\!]$, then C is also a partial bisimulation for $[\![\Phi(\phi_1, \ldots, \phi_n)]\!]$.*

We can see directly that this is equivalent to the following

5.4 Lemma
(i) An operation $\Phi(\phi_1, \ldots, \phi_n)$ is ***safe for partial bisimulation*** *iff both $[\![\Phi(\phi_1, \ldots, \phi_n)]\!]^d$ and $[\![\Phi(\phi_1, \ldots, \phi_n)]\!]^+$ are safe for bisimulation (in the original way).*
(ii) An operation $\Phi(\phi_1, \ldots, \phi_n)$ is ***safe for partial bisimulation*** *iff both $[\![+\Phi(\phi_1, \ldots, \phi_n)]\!]$ and $[\![+\Phi(\phi_1, \ldots, \phi_n)]\!]$ are safe for partial bisimulation.*

Part (i) follows from the fact, that the definition of bisimulation separates the $+$ and d components throughout, and (ii) follows from $[\![+\phi]\!]^+ = [\![\phi]\!]^+$ and $[\![+\phi]\!]^d = [\![\phi]\!]^+$ and $[\![\mathcal{D}\phi]\!]^+ = [\![\phi]\!]^d$ and $[\![\mathcal{D}\phi]\!]^d = [\![\phi]\!]^d$.

The following facts hold

5.5 Fact
The operators $\sim^s$, $\wedge$, $\oplus$, $+$, $\mathcal{D}$ are safe for partial bisimulation.

For $\sim^s$, the d part is trivial, the $+$ part identical to that for $\sim$ in the two-valued case. for $\wedge$, $\oplus$ the proves are identical to those for the two-valued counterparts ($\oplus$, $\cup$) applied twice to d and $+$

components. Both $+$ and $\mathcal{D}$ project on one of the two components, for which the bisimulation holds by assumption.

As van Benthem (1993) shows, in two valued logic, the safe operators that define first order operations on the relations or exactly those operators that are defined using $\sim$, $\wedge$ and $\cup$ (union of relations) alone. For three-valued logic, we have the following:

5.6 Proposition
The operators that are safe for partial bisimulation are those that can be defined using $\sim^s$, $\wedge$, $\oplus$, $+$ *and* $\mathcal{D}$.

Before I prove this, take a moment to realize that this is exactly what you expect. The first three behave identically to the 2-valued operations $\sim$, $\wedge$ and $\cup$, whereas the other two project on either the $+$ or $\mathcal{D}$ part of the relation. This combined with the fact that the bisimulation definitions are defined independently for these two parts, makes it impossible to imagine that this is wrong. But Let us prove it anyway.

Let us call a relation $[\![\phi]\!]$ that only takes the values *true* and *undefined* ($[\![\phi]\!]^+ = [\![\phi]\!]^d$) a $2'$-valued formula. We already saw that any three-valued expression ϕ can be written using two $2'$-valued expression $+\phi$ and $\mathcal{D}\phi$, where $\phi := (\perp_\phi \wedge \mathcal{D}\phi) \oplus +\phi$. The crucial observation now are

5.7 Lemma
(a) On $2'$-valued expressions, the operators $+ \sim^s$, $\wedge$ *and* $\oplus$ *have identical effect to those of* $\sim$, $\wedge$ *and* $\cup$ *on the 2-valued expressions of* DPL.
(b) A $2'$-valued expression R is safe for partial bisimulation if $R^+ = R^d$ is safe for bisimulation.
(c) The operator $\Gamma(\phi, \psi) := (\perp \wedge \mathcal{D}\phi) \oplus +\psi$ is safe for partial bisimulation.

For (a) observe that $\wedge$ is relation composition for both $+$ and $\mathcal{D}$ parts, so it is also relational composition for the $+$ part alone, and the disjunction is the same: disjunction for both parts implies disjunction for one of the parts. $+ \sim^s \phi$, a composite operator, is true if nothing makes ϕ true, and undefined (because $\sim^s \phi$ is false) if something makes it true. Again, identical behavior to $\sim$. (b) was proved in lemma 5.4ii, and (c) can easily be checked by observing that $\sim^s$ and $\oplus$ are safe for partial bisimulation (the latter again because it is essentially two instances of DPL's $\cup$) so their composition is too.

As a consequence, to find all partially bisimulation safe operators, we only have to sum up all partially bisimulation safe $2'$-valued operators, and consider pairs of these. Then Γ can be use to define a 3-valued operator. by the original proof on 2-valued dynamics and (a) above, all $2'$-valued operators can be defined using $\sim^s$, $\wedge$ and $\oplus$ but we also need $+$ and $\mathcal{D}$ to project any ϕ on two such operators $+\phi$ and $\mathcal{D}\phi$. **Thus the main proposition is proven.**

Not all operators of the logic defined in this chapter are safe. In particular, weak negation is not a safe operator, because safety of an operator on $[\![\phi]\!]^d$ and $[\![\phi]\!]^+$ does not need to mean that the operator is safe for $[\![\phi]\!]^-$. The reason is, that $g[\![\phi]\!]^- h$ holds iff $g[\![\phi]\!]^d h$ and $\neg g[\![\phi]\!]^+ h$ hold. Now the first we know, but the second contains a negation, which causes a problem. Even if $\neg g[\![\phi]\!]^+ h$, this does not mean that no output h' exists in the prime-model. For a slightly different reason, $\cap$ and $\cup$ are also not safe, because their defined clauses, both of which have a form demanding that output is the same for both components:

(57) $[\![\phi \cap \psi]\!]^d = [\![\phi]\!]^d \cap [\![\psi]\!]^d$, $[\![\phi \cup \psi]\!]^d = [\![\phi]\!]^d \cap [\![\psi]\!]^d$

and there is no way to be sure that when the two components have the same output in one model, they still have the same in another. However, all FDPL operators are safe. We already saw that $\sim$, $\wedge$, $+$ and $\mathcal{D}$ are safe, and it is easy to check that $\neg$, is definable by:

$$\neg\phi := (\bot_\phi \wedge +\phi) \oplus (+\sim^s \phi \wedge \mathcal{D}\phi)$$

and therefore also safe.

It is still an open problem what the operators are that are safe and strict. Partially because it is not completely clear whether the notion of strict used in this chapter is the right one. For example, the following variant of conjunction, defined by Krahmer (1995) is safe:

(58) $(\phi \odot \psi) := +(\phi \wedge \psi) \oplus (\bot\phi \wedge +(\sim(\phi \wedge \psi)))$.

This conjunction is a hybrid definition. It's true part is true exactly if the normal dynamic conjunction is, but the false part holds if there is no output making the conjunction true. this makes that $-(\phi \odot \psi)$ is true for exactly the same inputs and outputs that make $\sim(\phi \wedge \psi)$ true. However, double negation disappears: $--\phi$ and ϕ are true for the same inputs and outputs. The problem is to decide whether this is to count as a strict operator. Normally an expression is either strict in a parallel or in a serial way, but this a mixed case. Note, that the natural negation that goes with this conjunction is weak dynamic negation, an operator not safe for partial bisimulation. however, this might not be a real problem, given that this conjunction is not defined with the defined/true form of definitions in mind, but the true/false form, for which it is safe. I end with some observations.

First order relations That the result really only holds for first order relation between states shows the following operator, a variation of one proposed once by Paul Dekker (p.c.):

(59) $k = g + h :\Leftrightarrow (k(x) = g(x) \;\&\; h(x) = \star) \;\&\; (k(x) = h(x) \;\&\; h(x) \neq \star))$

(60) $g[\![\phi\vec{\sqcap}\psi]\!]h$ iff $\exists k, l \;\; g[\![\phi]\!]k \;\&\; g[\![\psi]\!]l \;\&\; h = k + l$

This strange version of parallel conjunction is bisimulation safe, but uses the structure of the states, and is therefore not expressible as a first order formula over the relations. It not definable in terms of the base operators. The symmetric version of this operator, which is the one Dekker actually proposed

(61) $\phi \sqcap \psi := (\phi\vec{\sqcap}\psi) \cap (\psi\vec{\sqcap}\phi)$

is actually very interesting. It is a version of the static conjunction that is defined for a slightly larger set of conjuncts, because it allows the conjuncts to bind different variables. When For the variables that both conjuncts bind they have to agree, but the different ones are just "added" to the output. If $\phi \cap \psi$ *is* defined, it is strongly equivalent to $\phi \sqcap \psi$.

$^+/_-$-bisimulation An alternative definition of bisimulation is given by demanding that C is bisimulation for R if it is a bisimulation for R^+ and R^-. This leads to a slightly larger set of safe-operators, which includes weak negation $-$. It is not clear, what exactly the set of operators is, although, when pressured, I would conjecture that it is the same set as before, with weak negation $-\phi$ added.

The reason that it is much harder to determine what the functionally complete set of operators is, is that for partial bisimulation, the d and $+$ components are effectively independent (cf. above) but no such fact holds for the $+/-$ case.

decidability It was already mentioned for the two-valued case that the fragment of relational algebra consisting of only $\wedge, \sim$ is decidable. The 2-layers perspective (d and $+$) that was so useful for proving bisimulation properties can also be used to argue that the 3-valued fragment consisting only of $\wedge, \sim, \neg, +, \mathcal{D}$ (i.e. the FDPL operators) is also decidable, because the d and $+$ components are.

5.3 Partializing the states

The next stage of partialization proposed in this chapter has been the use of partial states, carrying a natural notion of 'refinement'. This type of dynamic modeling is more in line with that found, e.g., in Update Semantics, where the information states come with an ordering of inclusion, which is used in defining - and constraining - dynamic operators. We have considered some new atomic operations made available in this richer structure - while leaving the operational structure essentially unchanged. For relevant logical literature on the use of partial assignments in dynamic semantics, we refer to Fernando (1994a), and for some similar issues in partial modal logic with an distinguished inclusion relation over states, to Jaspers (1994).

What we *can* do is mention a number of basic properties, and see whether the operators defined above satisfies these. This will give another explanation why certain operators seem more natural than others, but will not be enough to give exactly these operators and nothing else. consisting of partial assignments was defined using two relations between states, $[\![\phi]\!]^d$ and $[\![\phi]\!]^+$. It is possible to consider these two as relations that we can we give independent meanings to. I start with a constraint that restricts attention to pairs of relations that actually define a partial relation.

A true relation has to be a defined relation.

(62) $g[\![\phi]\!]^+h \rightarrow g[\![\phi]\!]^dh$

precedes the actual considerations. It defines the two relations to together form one partial relation.

Information Increase

The first real principle expresses the intuition that every formula expresses information, and that this information is stored in the state:

No formula decreases the information.

(63) $g[\phi]^d h \rightarrow g \preceq h$

DPL random assignment does not satisfy this principle, because this version of the random assignment will in general replace the old vale of the variable by a new one. All other operators satisfy this principle. This includes the closure operator. The principle only says something about the way output states are related to input states, nothing is said about how the input states are constructed. In fact the application of every operator listed above results in something satisfying the principle as long as its argument does.

Strictness

A principle that does restrict the operators to those that doe not throw away information is the principle (schema) of *strictness*. It states that all outputs of the result of applying an operator $[\Phi(\phi)$ should in some way originate in ϕ itself. For one-place operators the definition is straightforward enough, and as we saw before, for two place operators, there are two reasonable choices. Operators produce only output states that are relevant for their arguments.

(64) $g[\Phi(\phi)]^d h \rightarrow g[\phi]^d h$ (dynamic strictness)

(65) $g[\Phi(\phi, \psi)]^d h \rightarrow \exists k \;\; g[\phi]^d k[\psi]^d h$ (sequential strictness)

(66) $g[\Phi(\phi, \psi)]^d h \rightarrow \exists k \;\; g[\phi]^d h$ and $g[\psi]^d h$ (serial strictness)

Above we saw, that the only reasonable operator not satisfying strictness is the closure operator $!\phi$. It does satisfy a weaker principle that states that though that output may not originate in the argument, it should at least be licensed by there being an output:

(67) $g[\Phi(\phi)]^d h \rightarrow (g = h \;\&\; \exists k \;\; g[\phi]^d k)$ (static strictness)

The fact that closure destroys information makes it slightly suspect as a general operator.

Logicality

An important constraint on logical operators is that they *are* logical. This means that operators should not be sensitive to non-logical information like the structure of the states and the names of the variables actually used in the argument. Formally the relevant notion here is invariance of operators under permutation of non-logical information (van Benthem (1986a), Sher (1990)). In the current case, logical operators should preserve all dynamic properties under permutations of the space of states.

The identity of a partial state is determined by two components. First, a state is defined for certain variables, and undefined for others, second, the state assigns values to the variables it is defined for.

For an operator to be logical, there are two natural constraints. One is to demand that the identity of the variables does not influence the operator. The other is that the values assigned to variables do not matter for the operator (in both cases, it will, in general, matter for the arguments of the operator, just not for the operator itself.) The first can be expressed as follows. Let in the following V-PERM be a set of permutations on the natural numbers. Let $g[\![\pi^\star(\phi_1)]\!]h$ be defined as $\pi_\star(g)[\![\phi]\!]\pi_\star(h)$ and let $\pi_\star(g)(x_i) := g(x_{\pi(i)})$. Then we demand that the following holds:

(68) $\pi^\star(\Phi(\phi_1,\ldots,\phi_n)) = \Phi(\pi^\star(\phi_1),\ldots,\pi^\star(\phi_n))$

In a similar way, we can express the other constraint. Let D-PERM be a set of isomorphisms on the domain E. Now let $g[\![\pi^\bullet(\phi_1)]\!]h$ be defined as $\pi_\bullet(g)[\![\phi]\!]\pi_\bullet(h)$ and let $\pi_\bullet(g)(x_i) := \pi(g(x_i))$..

(69) $\pi^\bullet(\Phi(\phi_1,\ldots,\phi_n)) = \Phi(\pi^\bullet(\phi_1),\ldots,\pi^\bullet(\phi_n))$

Then a logical operator is an operator Φ such that it satisfies both constraints.

The effect of V-PERM is that variables are mixed up, making it impossible for the operators to be sensitive to which variable is which, The effect of D-PERM is that entities are mixed up, making it impossible for the operators to be sensitive to which value a variable is bound to. The effect is that operators are only sensitive to the propositional content of their arguments, not to the properties of the state. Logical operators should only look at the (truth) values $g[\phi]h$, and not at the states g, h themselves. All operators defined in this chapter satisfy this principle. Note that D-PERM expresses a well-known principle of logicality, also used in generalized quantifier theory. In dynamic logics, V-PERM is added to this, because the identity of variables becomes important.

Conclusions

In this chapter, I discussed Groenendijk and Stokhof's Dynamic Predicate Logic DPL, and proposed a number of changes, all having to do with partializing part of the theory. The arguments given for these changes are rooted firmly in linguistics, but I also gave more formal arguments.

First, in section 2, I proposed to partialize the dynamic expressions, distinguishing between true, false and inappropriate transitions. This proposal was based on the analysis of the behavior of negation as an operator on processes, which is what dynamic meanings are. This resulted in a **dynamic definition of negation** ($\neg$). This negation is such that $\neg\phi$ satisfies the same truth condition as $\sim\phi$, but on top of that has a dynamic effect. Although the dynamics of this logic is partial, the truth conditions, the static content of the logic, were shown to be total (two valued).

In section 3, the extra power we get by having a partial logic was used to define additional operators, in particular an operator $+$, **for presuppositions**, which resulted in partial truth conditions for the logic. Although it is also possible to define additional atomic actions, I did not do this.

In section 4, another form of partialization for dynamics is discussed, which is proposed by various authors, including Fernando (1994a), who uses analogies from logical recursion theory for the theory of *linguistic dynamics*. Here one changes from total to **partial states, introducing a natural notion of information increase**. Although here too an extension of the possible possible operators might be possible due to the additional degree of freedom, I restricted myself here to discussing possible alternative atomic actions. This section ends with an overview of *full dynamic predicate logic*, the logical system that results from the considerations in the previous sections.

Finally, section 5 discusses the formalism from a more logical perspective. I mention a number of reasonable constraints on dynamic operators. Results from the latter theory show that the chosen operators are in fact natural operators. It is quite likely, though not completely proven yet, that the operators in FDPL are exactly the operators that are strict and safe for their natural notion of bisimulation. The arguments in this chapter show that the operators of FDPL are natural ones from both a procedural, linguistical and logical perspective.

3

Plurality and Generalized Quantifiers

Introduction

The logic defined in chapter 2 makes singular logic dynamic. This logic which I have argued improves on earlier attempts to treat dynamic semantics is, as formulated, inadequate to deal with the complex examples of plural NPs introduced in Chapter 1. Because noun phrases commonly denote groups of entities, any treatment of NPs which does not provide an understanding of plurality is inherently inadequate. In order to apply dynamic semantics to arbitrary determiners, in this chapter I define a version of plural logic in which plural objects are interpreted as sets, based on the treatment of Scha (1981) and van der Does (1992).[1] which can be made dynamic along lines which exactly parallel the process by which the singular logic developed above was made dynamic. This static plural logic is made dynamic in Chapter 4.

The current chapter consists of five sections. In section 1, I briefly recall the discussion on basic noun-phrase and plural data of section 4 of chapter 1. I discuss a number of technical properties of plural quantifier readings, closely following (van der Does (1992)), in which terms of quantifiers are treated as relations between sets.

In section 2, in order to provide a form that can be made dynamic easily in the next chapter, the formalization of the plural phenomena discussed in the first section is recast in terms of quantifiers as operators on pairs of formulas. Finally, the question what exactly contributes to the sets that are quantified over, and what are just comments, is addressed.

In section 3, I give an alternative definition for the interpretation of formulas. Traditionally a formula is interpreted relative to an *assignment*, which assigns values to the variables in that formula. In a plural logic, the values assigned will be plural objects (*sets of entities*). In this section, I propose to replace interpretation relative to an assignment of sets by an interpretation relative to *sets of assignments*. The elements of this set are assignments that assign *singular entities* to the variables. This step makes it possible to express functional and relational dependencies between elements of a plural object.

In section 4 partialization of the logic is discussed to prepare for the combination of this logic and the partial dynamic logic proposed in the previous chapter. Finally, in section 5, some

1. The formalism defined by Link (1983, 1991) interprets plurals as abstract objects (so called i-sums) on a complete Boolean Lattice. He does this to stress the similarity between plurals and mass-terms. However his logic and the logic based on 1-level sets are essentially equivalent. Some of the definitions in this chapter would get a more complicated definition when phrased in terms of Link's i-sums. Also cf. footnote 7 on page 31.

more mathematical issues are raised, and the received view on plurals is discussed somewhat further.

1 Plurality

Introduction

In this section I will define a general treatment of quantifiers over plural objects. The semi-informal discussion in section 4 of chapter 1 already introduced some notation that van der Does (1992) uses to express the different readings in a formal system. A plural predicate will be interpreted as a relation between *sets of sets of entities*. This means that, whereas in standard predicate logic the interpretation of the predicate $\mathbf{woman}$ is the set of women, in plural logic the interpretation of $\mathbf{women}$ is the set of sets of women. As a consequence, quantifiers are relations between two sets of sets. In the following I will denote, as I did in chapter 1, sets of sets as $\mathbf{X}, \mathbf{Y}, \ldots$ and elements of these as $A, B, C, \ldots$.

Before I can start discussing quantifier readings and their properties, there are two obstacles I have to get out of the way. Both have to do with the fact that plural predicate are sets of sets of entities, rather than the sets of entities in standard generalized quantifier theory. First, generalized quantifiers are relations between sets. So in order to talk about generalized quantifiers in plural logic a method has to be found to lift these quantifiers to the right level. Second, the important notion of conservativity ($Q(A, B)$ is equivalent with $Q(A, A \cap B)$) is also phrased in terms of sets of entities. Again I discuss how to solve this mismatch.

Let me stress that what I am after is to find an operator Q on sets of sets of entities that is the same as the operator Q on sets of entities. I am not after an operator that constitutes a quantification over sets *as* entities. That would result in the original generalized quantifier theory.

1.1 Maximization

The interpretation of predicates as sets of sets raises an important question. Traditional generalized quantifiers are relations between sets of entities ($Q(A, B)$). How can we define a quantifier over plural arguments ($\mathsf{Q}(\mathbf{X}, \mathbf{Y})$) which we can still recognize to be “the same” quantifier? The answer to this is to lift quantifier meanings to be relations between sets of sets. But that leaves us with another question. What lift is acceptable and what not?

To answer this question, I will first discuss a simpler one. Given a set of sets $\mathbf{X}$, what is *the* set of entities corresponding to $\mathbf{X}$? If $\mathbf{X}$ is a powerset, like the interpretation of a noun-phrase like $\mathbf{women}$, then the answer is easy: it is $\bigcup \mathbf{X}$. In fact, at first, $\bigcup \mathbf{X}$ may not seem a bad choice for any kind of set of sets. However, there is a problem: $\bigcup \mathbf{X}$ may not be itself an element of $\mathbf{X}$. If it is not, $\bigcup \mathbf{X}$ would not be an acceptable representative of the sets in $\mathbf{X}$.

The set of entities A corresponding to $\mathbf{X}$ should satisfy three constraints. First, it should be an element itself: $A \in \mathbf{X}$. Secondly, if $\bigcup \mathbf{X} \in \mathbf{X}$, A should be identical to $\bigcup \mathbf{X}$ and thirdly, there should not be a larger set B, ($A \subseteq B$), such that $B \in \mathbf{X}$, otherwise B would be an obviously better candidate.

If these necessary conditions are now also taken as sufficient conditions, then the resulting definition of sets A that represent $\mathbf{X}$ becomes:

(1) $\mathsf{M}(\mathbf{X}) = \{A \in \mathbf{X} \mid \neg \exists B \in X \;\; A \subsetneq B\}$. (Local Maximum)

Note that this does not define *the* set representing $\mathbf{X}$, there may not be a unique one. I will call $\mathsf{M}(\mathbf{X})$ the set of **(locally) maximal elements** of $\mathbf{X}$. In cases that the domain is not finite, the set $\mathbf{X}$ could conceivable not have a maximum at all. For example, the set F of all finite sets in a given, infinite, domain will not have a maximum in F, there will always be a larger one. I will return to this in the last section. In natural language examples, such exotic sets do not seem to occur.

Maximization, or exhaustive interpretation of quantifiers as it is sometimes called, has been somewhat of a popular topic in the last few years. It is used by Sher (1990), Spaan (1993), and Beghelli et al. (1995) to discuss properties of branching quantifiers and the so called cumulatives. In chapter 4, I will argue that maximization as described here is not just "a good trick" to simulate abstraction, but that in fact the discussion surrounding E-type pronouns gives us independent reasons to believe that this maximization actually occurs.

1.2 Conservativity

A quantifier Q is called conservative if $Q(A, B)$ only depends on the members of A: knowing A and $A \cap B$ is enough. Conservativity means that $Q(A, B)$ will always be equivalent with $Q(A, A \cap B)$ (cf section 3 of chapter 1). It is often assumed that all generalized quantifiers that are interpretations of natural language expressions are conservative.

This poses a problem. How can we translate this principle to plural quantifiers, which have the form $\mathsf{Q}(\mathbf{X}, \mathbf{Y})$? in his dissertation, van der Does (1992) proposes several possible lifts of the principle of conservativity . A variant of these is applicable here. In the previous section, I explained that the definition of $\mathsf{Q}(\mathbf{X}, \mathbf{Y})$ involves selecting preferred (locally maximal) members A of $\mathbf{X}$ and B of $\mathbf{Y}$. For the moment I will not make any assumptions concerning how the plural quantifier Q is defined in terms of Q, just that the definition involves instances of $Q(A, B)$ for preferred members A, B. Take any such $Q(A, B)$. According to the standard theory, this is equivalent to $Q(A, A \cap B)$. However, the last formula suffers a similar problem as the one we had in the previous section: $\bigcup \mathbf{Y}$ need not be a member of $\mathbf{Y}$ and neither does $A \cap B$ have to be an element of $\mathbf{Y}$.

This suggests the following constraint on the definition of $\mathsf{Q}(\mathbf{X}, \mathbf{Y})$. First, it involves certain (maximal) elements $A \in \mathbf{X}$, and given such an A, it will involve elements $A \cap B \in \mathbf{Y}$. By the argument given in the previous section, these $A \cap B$ should be the largest such sets. That is, we are interested in elements maximal in $(\wp(A) \cap \mathbf{Y})$. Formally, this means that he following lifted version of conservativity should hold:

(2) $\mathsf{Q}(\mathbf{X}, \mathbf{Y}) \equiv \mathsf{Q}(\mathsf{M}(\mathbf{X}), \mathsf{M}(\{\wp(A) \cap \mathbf{Y} | A \in \mathsf{M}(\mathbf{X})\}))$

Even stronger versions can be formulated. For example, quantifiers should be about entities, not about sets of entities, and in dynamic logic it is important that the choice of A in both arguments is the same:

(3) $\forall A \in \mathsf{M}(\mathbf{X})\mathsf{Q}(\{A\}, \mathbf{Y}) \equiv \mathsf{Q}(\{A\}, \mathsf{M}(\wp(A) \cap \mathbf{Y}))$

Throughout this chapter I will be assuming this principle. In section 5, a number of alternatives are listed and linked with different readings of quantifiers. In chapter 4, I will show that this form of conservativity is directly related to the dynamic argument for conservativity on page 42 [2:1.1].

1.3 Readings

A simple lifted definition of quantifiers on sets of sets can now be given:

(4) $Q^{\mathsf{M}}\mathbf{XY} := \exists A \in \mathsf{M}(\mathbf{X})\, \exists B \in \mathsf{M}(\wp(A) \cap \mathbf{Y})\ \ Q(A, B).$

This lift gives the quantifier a collective interpretation. Collective, because there has to be one set in the nuclear scope $\mathbf{Y}$, that is also part of a set in the restriction $\mathbf{X}$. This definition is slightly different from the one that van der Does (1992) gives for the collective reading. His definition[2] is similar, but does not involve maximization [1:3]:

(5) $Q^{c}(\mathbf{X}, \mathbf{Y}) := \exists A \in \mathbf{X}\, \exists B \in \wp A \cap \mathbf{Y}\ \ Q(A, B).$

In this definition the maximized case is just a particular choice. This makes some difference, but not too much. Look at example (6) again, in the collective reading (cf. the discussion about this sentence in chapter 1 [1:4.2]). If there are several other groups of women, of any cardinality, lifting pianos, I can still use the collective reading to single out one of these. The difference between van der Does' definition and mine is that he can select any four women belonging to a group of piano-lifters, independently of whether it is part of a larger group of piano-lifters. By contrast, I demand that the group really consists of four women, but if there are more such groups, any one will do[3].

Below I will argue that the reading where some subset of a group can also be selected, a reading often called the adjectival reading of the quantifier, is in fact a different quantification, essentially equivalent to an indefinite. I will argue that this involves a different way of packaging the information, and that an utterance with the maximized reading given here is not the same as an utterance that gets the adjectival interpretation. In the next chapter I discuss how maximization

2. He assumes the reasonable simplification that $\mathbf{X} = \wp(X)$ for some set, and gives the formalization:

$$Q^{c}(X, \mathbf{Y}) := \exists B \subseteq X\ (QAB\ \&\ B \in \mathbf{Y}).$$

For $\mathbf{X} = \wp(X)$ this is equivalent to the (5).

3. My definition does not yet completely achieve this goal, because it is atemporal. For example, in the case of (6), there may be a group of four women that lifted a pianos a a maximal group, but at another time joined with some more women to lift another piano. The current analysis of collectivity would mistakenly consider this last group bigger, and make the sentence come out false. So where the original reading comes out too weak, this one comes out too strong. This problem can only be solved when the temporal dimension (and for other kinds of cases probably also the spatial dimensions) is added to the formalism, either in the form of events or in the form of stages of individuals. Cf. Krifka (1990) for discussion of these kind of questions. This is a subject beyond the scope of the current investigations.

is based on observations in discourse semantics, in particular, on properties of so called E-type pronouns.

The other readings of quantifiers can now be obtained by considering variants of this scheme. Let us see how this is done using example (6):

(6) *Four women lifted a piano.*

The three readings[4] that I gave informally in section 4.2 of chapter 1, can now be repeated here in more formal terms. No attempt is made in this list to give a uniform notation.

(8) $\exists A \in \mathsf{M}(\mathbf{X})\ \exists B \in \mathsf{M}(\wp(A) \cap \mathbf{Y})\ \#B = 4$

(9) $\#\{a \mid \{a\} \in \mathbf{X}\ \&\ \{a\} \in \mathbf{Y}\} = 4$

(10) $\#\{a \in A \mid \exists A \in \mathbf{X}\ \ A \in \mathbf{Y}\} = 4$

This looks as if there is no relation between these readings whatsoever. But it is possible to make the relations between these readings clearer by introducing some primitive operators, in a similar way to van der Does (1992).

(11) $\mathbf{1}(\mathbf{X}) = \mathbf{X}$

$\alpha(\mathbf{X}) = \{\{a\} \mid \{a\} \in \mathbf{X}\}$

$\pi(\mathbf{X}) = \{\bigcup \mathbf{Y} | \mathbf{Y} \subseteq \mathbf{X}\}$

$\delta(\mathbf{X}) = \pi(\alpha(\mathbf{X}))$

Of these, $\mathbf{1}$ is the operator involved in the collective reading, α the operator for the singular reading (not normally discussed because strictly speaking, it is not a singular at all), π is the operator that causes the pseudo-distributive reading, and δ, finally, is the operator that gives rise to the distributive reading, something we can see by observing that

$$\delta(\mathbf{X}) = \wp(\bigcup \alpha(\mathbf{X})) = \{a \mid \{a\} \in \mathbf{X}\}$$

4. The fourth, the deviant neutral reading, will not be included in the rest of the discussion. For completeness, it would be

$$\#\{a \in A \mid \exists A \in \mathbf{X}\, \exists B \supseteq A\ \ B \in \mathbf{Y}\} = 4$$

and the corresponding operator, as defined in (8):

$$\nu(\mathbf{X}) = \{A \mid A \subseteq \bigcup \mathbf{X}\}$$

Note that δ amounts to pseudo-distribution over singular objects.

In terms of these operators, and using $4(A, B) := (\#(A \cap B) = 4)$, the plural quantifier readings can be rewritten as

(13) $Q^c(A, \mathbf{Y}) := \exists A \in \mathsf{M}(\mathbf{1}(\mathbf{X}))\, \exists B \in \mathsf{M}(\mathbf{1}(\wp(A) \cap \mathbf{Y}))\;\; 4(A, B)$

(14) $Q^d(A, \mathbf{Y}) := \exists A \in \mathsf{M}(\delta(\mathbf{X}))\, \exists B \in \mathsf{M}(\delta(\wp(A) \cap \mathbf{Y}))\;\; 4(A, B)$

(15) $Q^p(A, \mathbf{Y}) := \exists A \in \mathsf{M}(\pi(\mathbf{X}))\, \exists B \in \mathsf{M}(\pi(\wp(A) \cap \mathbf{Y}))\;\; 4(A, B)$

If $\mathbf{X} = \wp(A)$, we have $\mathbf{1}(\wp(A)) = \pi(\wp(A)) = \delta(\wp(A)) = A$ and then the formulas reduce to:

(16) $Q^c(A, \mathbf{Y}) := \exists B \in \mathsf{M}(\mathbf{1}(\wp(A) \cap \mathbf{Y}))\;\; 4(A, B)$

(17) $Q^d(A, \mathbf{Y}) := \exists B \in \mathsf{M}(\delta(\wp(A) \cap \mathbf{Y}))\;\; 4(A, B)$

(18) $Q^p(A, \mathbf{Y}) := \exists B \in \mathsf{M}(\pi(\wp(A) \cap \mathbf{Y}))\;\; 4(A, B)$

The operators δ, π, α and $\mathbf{1}$ form a natural group of transformations[5]:

	$\mathcal{O}_2$				
$\mathcal{O}_2 \circ \mathcal{O}_1$		$\mathbf{1}$	α	π	δ
$\mathcal{O}_1$	$\mathbf{1}$	$\mathbf{1}$	α	π	δ
	α	α	α	δ	δ
	π	π	α	π	δ
	δ	δ	α	δ	δ

Using these operators it is possible to give a general schema for quantifier readings. First of all, a quantification will involve a singular or a plural noun-phrase interpretation as its restriction. If it is singular, it is assumed that the operator α is present, if it is plural, it will be the operator $\mathbf{1}$. Then both singular and plural quantifications, are ambiguous between a collective ($\mathbf{1}$), a distributive reading (δ) or a pseudo-distributive reading (π). This leads to the following general schema of a quantification:

1.1 Definition (Plural Quantification)
Plural quantifiers have the format

---

5. $\mathbf{1} \circ \mathcal{O} = \mathcal{O} \circ \mathbf{1} = \mathcal{O}$. $\mathcal{O}_2 = \alpha$ takes the singleton sets from its argument, first adding elements together which is what δ and π do will not add any new singleton sets, also, taking the singletons after the singletons does not do anything new. so $\alpha \circ \delta = \alpha \circ \pi = \alpha \circ \alpha = \alpha$. $\mathcal{O}_2 = \pi$ closes the set under unions, so doing this again does not add anything: $\pi \circ \pi = \pi$. If we remember that function composition is associative, the others follow from the definition of δ: $\pi \circ \alpha = \delta$ (by def.), $\pi \circ \pi \circ \alpha = \pi \circ \alpha$, $\pi \circ \alpha \circ \alpha = \pi \circ \alpha$, $\pi \circ \alpha \circ \pi = \pi \circ \alpha$, $(\pi \circ \alpha) \circ (\pi \circ \alpha) = \pi \circ (\alpha \circ \pi) \circ \alpha = \pi \circ \alpha = \pi \circ \alpha$.

$$\exists A \in \mathsf{M}\,(\mathcal{O}\,(\mathbf{X}))\ \exists B \in \mathsf{M}\,(\mathcal{O}((\mathbf{num}(\wp(A) \cap \mathbf{Y})))\ Q(A, B)$$

where $\mathcal{O} \in \{\delta, \pi, \mathbf{1}\}$ *and* $\mathbf{num} \in \{\alpha, \mathbf{1}\}$.

Plural quantifications have potentially three readings (collective, pseudo-distributive and distributive) and singular quantifications have potentially two readings (singular and distributive). Often, lexical information will restrict the readings, preferring only one of these. For example, *every man walks* is a singular quantification, but it only has a distributive reading because of the meaning of *every*. Note further, that I assume that the operator $\mathcal{O}$ is the same for both the restriction (the NP) and the nuclear scope (the VP). Provided $\mathbf{X}$ is a powerset, this is not problematic, for then $\bigcup \alpha(\mathbf{X}) = \bigcup \mathbf{X}$. Also note, that the operator selecting singular or plural quantification is only present in the nuclear scope (second) argument. Although here too we could add it to both arguments without changing the interpretation, the chosen version will make more sense dynamically, it allows to identify A as the CN-anaphor, B as the (normal) discourse referent. But this is the subject of the next chapter. Following van der Does, I will sometimes speak about the different readings as different **varieties** of the quantifier.

2 Plural Predicate Logic

Introduction

The discussion so far used the language of set-theory to introduce a number of concepts concerning plural quantification. The language was based on the one used by van der Does (1992), which is the one most commonly used: an explicit logic over sets. This makes the comparison with the different versions in the literature easier. However, the above format has its limitations. In natural language, most quantifiers can be embedded inside other quantifiers, and a set-oriented formalization becomes cumbersome very soon when used to formalize the resulting dependencies. Furthermore, the more the logical language looks like standard predicate logic, the easier it is to use the methods discussed in the previous chapter to define a dynamic logic based on it. And this, after all, is our ultimate objective.

In this section I will define a logic that has the syntax of standard predicate logic with one distinctive predicate $\subseteq$, which is to be interpreted as the subset relation. Although the interpretation of this logic is obvious (standard predicate logic with variables over sets, cf. also Lønning (1987b)), I will give it explicitly to introduce the notation. I will use bold letters ($\mathbf{g}, \mathbf{h}, \ldots$) for the assignments, for the moment just to remind you that the values assigned are sets.

2.1 Definition (Plural Predicate Logic)
The syntax of PPL *is that of ordinary predicate logic with identity. The semantics of* PPL *is the standard interpretation over a model* $\langle \mathcal{D}, \mathcal{I} \rangle$*, restricted to domains with a subset structure (*$\mathcal{D} = \wp(E)$*), and a special interpretation of the predicate* $\subseteq$.

(19) $\|Px_1 \ldots x_n\|^{\mathbf{g}}$ iff $<\mathbf{g}(x_1), \ldots, \mathbf{g}(x_n)> \in \mathcal{I}(P)$
$\|x \subseteq y\|^{\mathbf{g}}$ iff $\mathbf{g}(x) \subseteq \mathbf{g}(y)$
$\|\phi \wedge \psi\|^{\mathbf{g}}$ iff $\|\phi\|^{\mathbf{g}}$ and $\|\psi\|^{\mathbf{g}}$
$\|\neg\phi\|^{\mathbf{g}}$ iff not $\|\phi\|^{\mathbf{g}}$
$\|\exists x \phi\|^{\mathbf{g}}$ iff $\exists \mathbf{k} \approx_x \mathbf{g}\ \|\phi\|^{\mathbf{k}}$

There are some auxiliary notions that will be of use later:

(20) $\mathbf{empty}(x) := \forall y(y \subseteq x \rightarrow y = x)$

(21) $\mathbf{sing}(x) := \forall y(y \subseteq x \rightarrow (\mathbf{empty}(y) \vee y = x))$

(22) $\mathbf{plur}(x) := (x = x)$

These characterize the empty set (20) and the singleton sets (21). An operator corresponding to plurals ($\mathbf{plur}$) is given to complement $\mathbf{sing}$ for singulars, but it has no content. Following a long standing plural-logic tradition, the singleton sets will denote the singulars, $\mathbf{sing}$ will correspond to α in the previous section.

Already two differences with other logics of plurality will be apparent: the empty set is a plural object, and the plural operator does not demand that the set in question has more than one member.

The exclusion of the empty set is demanded in most theories of plurals, but nowhere so extremely as in abstract boolean lattice theory of Link (1983) and work based on this. In this, plural objects are elements of an abstract complete boolean lattice, the atoms of which are the individual entities. The question where the empty set is does not even come up. Other authors do discuss the empty element (cf. an extensive discussion in (Lønning 1987b)), but conclude that it does not matter much. Below I will show that the empty set as a real, be it a rather degenerate, plural object is very useful. Using it we can make sense of the difference in behavior of collective universal quantifiers, which have existential import, and distributive universal quantifiers, which do not. In chapter 4 I will show that this also has dynamic consequences: it enables a distributive universal quantifier to succeed on an empty restriction, while introducing an empty (or failed) object.

An alternative definition for $\mathbf{plur}(x)$, similar to the one given by van der Does (1992), is to put $\mathbf{plur}(x) := \neg\mathbf{sing}(x)$. The definition of the plural operator as a tautology is based on linguistic data. One observation is that if the cardinality of something is unknown, the plural is used. As an answer to the question (23)

(23) *Where there any bugs in the room?*
a. *Yes, I found one in the lamp.*
b. $^?$*No, only one, it was in the lamp.*

the answer could very well be (23a). But if the plural would demand there to be more than one, the answer should have been phrased as (23b). Data from other languages seems to suggest that this is the norm. Even languages with a more subtle number system, like arabic, which have *duals*, *plural but less than six*, and even *plurals of duals* and *plurals of plurals* (for certain nouns) seem to be such that the larger amounts include the smaller amounts, and the restriction to the complement is done on pragmatic grounds (cf. (Ojeda 1992, Roberts 1983).

2.1 Distributives

In this logic the operators δ and π can be defined. A formula containing a variable x is true or false for different values assigned to x. Because this is a plural logic, and the values assigned to x are sets, this means that the formula is evaluated relative to a set assigned to x. Therefore we say that the formula is evaluated collectively (relative to the set x). We then define an operator that changes this[6]:

(24) $\delta_x(\phi) := \forall y \subseteq x(\mathbf{sing}(y) \rightarrow \phi[x/y])$.

A formula of the form $\delta_x(\phi)$ is true for a set assigned to x, if the formula ϕ is true for the elements of x (repackaged as singleton sets). Two standard notions of distributivity can be defined in terms of this operator as follows:

(25) A one place predicate P is distributive iff Px and $\delta_x(Px)$ always have the same truth value.

(26) The distributive interpretation $Q^\delta x(\phi, \psi)$ of a quantifier Q is $Qx(\delta_x(\phi), \delta_x(\psi))$.

Note a conceptual difference between these two notions and the notion of a distributivity operator. Distributivity for predicates has to do with a position in the argument list; distributivity for quantifiers has to do with different ways of translating a quantification in set theory. Both are syntactic, configurational, properties of formulas. The distribution operator, on the other hand, is a semantic notion, that can be defined in the language. Of course, in the presence of type theory this difference disappears, because you can then abstract over the variable and manipulate that argument.

Translated as an operator the pseudo-distribution operator is defined as follows

(27) $\pi_x(\phi) := \forall y \subseteq x(\mathbf{sing}(y) \rightarrow \exists z \subseteq x(y \subseteq z \wedge \phi[x/z]))$

This says that every element y of x is contained in some subset of x that has the required property. Note that both this and π do not demand any uniqueness of x, $\delta_x\mathbf{walk}(x)$ is a property that holds of sets of which every element walks, $\pi_x\mathbf{sing} - \mathbf{together}(x)$ is a predicate that holds of x if x consists of subsets that sing together. Whereas the sets involved in collective and distributive quantification are defined uniquely (the set itself and the singleton sets that can be made out of the elements of the set respectively), this is not the case for pseudo-distributivity; there are large number

6. Remember that $\phi[x/y]$ denotes the formula that you get if you substitute all occurrences of x in ϕ by y, where it is assumed that the substituted variable does not occur in ϕ.

of covers of any given set. To reject a set of n elements, the collective interpretation only has to check one case, the set itself, the distributive also has to check one case (but the n elements independently). The pseudo-distributive on the other hand has to check a large number of different covers, before the set can finally be rejected. This means that the complexity of calculating the pseudo-distributive interpretation is much higher than that of the collective or the distributive interpretation.

2.2 Varieties of Quantification

In this section I will repeat the discussion of sections 1.1–1.3 on quantifier readings, but now in the language of plural predicate logic. In plural predicate logic, predicates are interpreted as sets of n-tuples of sets of entities. In particular, a 2-place predicate is a relation between sets. Standard generalized quantifier theory interprets quantifiers also as relations between sets [1:3]. This means that every generalized quantifier corresponds with a two place predicate in this logic. We can use this to implement different expressions for the received view on plural quantifiers within this language.

I will define a notion of *pluralized* quantifier $\mathsf{Q}x\ (\phi, \psi)$ in terms of the 2-place "predicate" Q. This predicate will hold for two arguments (sets), whenever the generalized quantifier it represents holds between these sets. To define a quantifier $\mathsf{Q}x\ (\phi, \psi)$, we have to do three things

(i) we have to find a definition of the sets corresponding to ϕ and ψ within plural logic, then
(ii) assign these sets as values to variables x' and x and
(iii) use these variables as arguments to the quantifier-predicate $Q(x', x)$.

I proceed parallel to the previous discussion involving sets of sets.

Standard generalized quantifier theory can be paraphrased as follows. Take *the* set corresponding to the restriction and call it A, take *the* set corresponding to the nuclear scope and call it B, then, apply the quantifier to these: $Q(A, B)$. In plural logic we do almost the same. Take *a* set corresponding to the restriction and call it x', take *a* set corresponding to the nuclear scope and call it x, finally, apply the quantifier to these: $Q(x', x)$.

The maximization operator that was used in the preceding discussion to define sets corresponding to a formula can be defined in this logic as follows:

(28) $\mathsf{M}_x(\phi) := \neg\, \exists y\ (x \subsetneq y \wedge \phi[x/y]) \wedge \phi$

An expression $\mathsf{M}_x(\phi)$ is true for a given value assigned to x if ϕ is true for that value of x and if there is no larger value that can be assigned to that x which would also make ϕ true.

This completes the list of components needed to translate the definitions for the different readings of quantifiers I gave above into plural predicate logic. Let us review these.

As we have seen, expressions can be either plural or singular. I take this to be an absolute: plural expressions translate as something that contains $\mathbf{plur}$, singular expressions as something that contains $\mathbf{sing}$. Given that the number is fixed, quantifiers are in general ambiguous between a $\mathbf{1}$, a δ or a π reading. The choice may be restricted by lexical constraints and δ and π result in the same reading in the singular case. The general schema given in definition 1.1 can now be repeated in terms of this logic:

2.2 Definition (Generalized Quantifiers in PPL)
In plural predicate logic, generalized quantifiers have the format

$$\mathsf{Q}x\,(\phi,\psi) :=$$
$$\exists x, x'\,(\mathsf{M}_{x'}(\mathcal{O}_x(\phi[x/x'])) \wedge \mathsf{M}_x(\mathcal{O}_x(x \subseteq x' \wedge \mathbf{num}(x') \wedge \psi)) \wedge Q(x', x))$$

where $\mathcal{O} \in \{\delta, \pi, \mathbf{1}\}$ *and* $\mathbf{num} \in \{\mathbf{sing}, \mathbf{plur}\}$.

This is identical to definition 1.1[7]. The set A is here represented by x', the set B by x. The choice between α (resulting in only singletons) and $\mathbf{1}$ (resulting in arbitrary sets) is replaced by a choice between $\mathbf{sing}$ (singletons) and $\mathbf{plur}$ (arbitrary sets). The operators $\mathbf{1}$, δ and π have the same meaning in both definitions. Taking a locally maximal subset of $\wp(A) \cap \mathbf{Y}$ is identical to taking a locally maximal subset satisfying $x \subseteq x' \wedge \psi$. In the following discussion, I will omit $\mathbf{1}$ and $\mathbf{plur}$ if they occur in formulas, because they do not make any real contribution to the meaning.

The schema gives rise to three forms for the plural case, and three forms for the singular. However, some are equivalent[8], resulting in only four readings. For plural restrictions we have:

(29) $\mathsf{Q}x\,(\phi,\psi) :=$ (collective)
$$\exists x, x'\,(\mathsf{M}_{x'}(\phi[x/x']) \wedge \mathsf{M}_x(x \subseteq x' \wedge \psi) \wedge Q(x', x))$$

(30) $\mathsf{Q}^{\delta}x\,(\phi,\psi) :=$ (distributive)
$$\exists x, x'\,(\mathsf{M}_{x'}(\delta_{x'}(\phi[x/x'])) \wedge \mathsf{M}_x(\delta_x(x \subseteq x' \wedge \psi)) \wedge Q(x', x))$$

(31) $\mathsf{Q}^{\pi}x\,(\phi,\psi) :=$ (pseudo-distributive)
$$\exists x, x'\,(\mathsf{M}_{x'}(\pi_{x'}(\phi[x/x'])) \wedge \mathsf{M}_x(\pi_x(x \subseteq x' \wedge \psi)) \wedge Q(x', x))$$

and for singular restrictions we have:

(32) $\mathsf{Q}^{\mathsf{s}}x\,(\phi,\psi) :=$ (singular)
$$\exists x, x'(\mathsf{M}_{x'}(\phi[x/x']) \wedge \mathsf{M}_x(x \subseteq x' \wedge \mathbf{sing}(x) \wedge \psi) \wedge Q(x', x))$$

(33) $\mathsf{Q}^{\delta,\mathsf{s}}x\,(\phi,\psi) :=$ (distributive)
$$\exists x, x'(\mathsf{M}_{x'}(\delta_{x'}(\phi[x/x'])) \wedge \mathsf{M}_x(\delta_x(x \subseteq x' \wedge \mathbf{sing}(x) \wedge \psi)) \wedge Q(x', x))$$

(34) $\mathsf{Q}^{\pi,\mathsf{s}}x\,(\phi,\psi) :=$ (distributive)
$$\exists x, x'(\mathsf{M}_{x'}(\pi_{x'}(\phi[x/x'])) \wedge \mathsf{M}_x(\pi_x(x \subseteq x' \wedge \mathbf{sing}(x) \wedge \psi)) \wedge Q(x', x))$$

---

7. An alternative definition would have been to choose the form $\ldots \mathsf{M}_x(x \subseteq x' \wedge \mathcal{O}_x(\mathbf{num}(x) \wedge \psi))$. Because x and x' are sets, this would not have made a difference.

8. This follows from $\pi \circ \alpha = \delta \circ \alpha = \delta$: $\pi_x(\mathbf{sing}(x) \wedge Px) = \delta_x(\mathbf{sing}(x) \wedge Px) = \delta_x(Px)$.

It is interesting to see that the possibility to distinguish between plurals and singulars does not add much. The only really new reading is the singular, normally ignored because it is of little interest to generalized quantifier theory: almost all quantifiers would be trivially true or false if restricted to singleton sets. The other two "readings" are both equivalent to the distributive because $\delta_x(\mathbf{sing}(x) \wedge \phi)$, $\pi_x(\mathbf{sing}(x) \wedge \phi)$ and $\delta_x(\phi)$ are equivalent.

I already mentioned when discussing the set-theoretic version of plural quantifier theory, that this system is essentially equivalent to the theory given by van der Does (1992). The only difference is that in his version of the collective, maximization does not occur (cf. the discussion on page 84).

2.3 Downward Monotonic Quantifiers

In the above I made the generally accepted assumption that the restrictions of quantifiers, the nouns, are generated as the powersets of sets. This means that they always have a unique maximum: the set generating them. We can safely assume that the restriction will cause no problems. The nuclear scope, on the other hand, is something else entirely. It might have a non-unique maximum. Consequently the existential quantifier will pick out one of the possible local maxima; a plural generalized quantifier is true, if it is true for one possible nuclear scope.

This is not always what we want. It will cause problems for collective readings of MON↓ quantifiers. Take the example

(35) *At most three women gathered in the square.*

Suppose there are exactly five women, and there are two groups that gathered independently: {Sue, Mary, Fatima} and {Agnes, Zoe}. Then the existential in the definition makes this true if we would apply the quantifier, because there is one set ({Agnes, Zoe}), that makes the resulting formula true.

This points at a curious fact. Although the collective is defined by means of a maximization, as opposed to the definition in van der Does (1992), the definition is not yet maximal enough. It two sets have a subset relation, it takes the maximal one, but if there is no relation like this, both will do (provided there are no other sets in the way).

The crucial observation is that MON↓ make claims about all cases in the relevant domain. A sentence like (35) is true when all sets of gathering women satisfy the description, not just one of them. Or to put it the other way around, the sentence is equivalent with

It is not the case that three or more women gathered in the square.

This suggests, as van der Does (1994) also remarks, that MON↓ quantifiers should be interpreted as negations of MON↑ quantifiers. This means that if the quantifier Q is MON↓, the translation for the collective reading should be:

(36) $$\neg\exists x, x' \left(\mathsf{M}_{x'}(\pi_{x'}(\phi[x/x'])) \wedge \mathsf{M}_x(x \subseteq x' \wedge \psi) \wedge Q^d(x', x)\right)$$

where Q^d is the dual of Q.

This definition becomes really interesting in the dynamic case, where part of the non-uniqueness may be caused by non-unique dynamic effects. The above solution will also work in these cases, and will explain some of the observations made in Kanazawa (1993b), where choices among different dynamic readings of quantifiers are related to monotonicity properties. Section 3.2 of Chapter 5 discusses this issue.

2.4 Packaging

Maximization turns out to have further independent uses. Most notably, it allows us to express the various modes of "packaging" that occur in natural language.

All but the simplest sentences correspond to meanings that will have additional content next to the basic predication. some information in the clause will be presupposed, some will be part of the actual restriction or quantification, and yet another part may be just a non-restrictive comment. What is part of creating the set the quantification is about, and what is only predicated of that set —what goes inside the maximization operator and what outside— is determined by the whole linguistic expression, including lexical markers and intonation. In English, most of the work of structuring the information is done by prosody, but since written language lacks this, other, discourse, principles are used to reconstruct the intended prosody[9]. In other languages, like Hungarian, a lot of such structuring information *is* expressed using syntactic means, and there the written language should, at least for the dimensions expressed configurationally, give rise to less ambiguity.

Non-restrictive Phrases

First among the examples of packaging are non-restrictive phrases or clauses.

(37) *The farmers, who were not that intelligent, bought too many cows.*

has a reading were the subordinate clause does not help determine the restrictive set, but is just a comment on it. A non-restrictive component need not be full clause:

(38) *The stupid farmers did not see the cow-tax coming.*

has a similar reading; where the quantification is over farmers, and *stupid* only a comment (a non-restrictive adjective) about them. We can encode this by having the predicate $\mathbf{f}$ inside the maximization operator, but s outside of it.

(39) $\exists x', x \;\; \mathsf{M}_{x'}(\mathbf{f}(x')) \wedge \mathsf{s}(x') \wedge \ldots$

---

9. It should never be forgotten that spoken language comes first, and that prosody is an indivisible part of it. In written English, which lacks the essential intonational part of the linguistic structure, ambiguities are much more common than in spoken English. Still, even in spoken language, discourse principles (Prüst 1992, Prüst et al. 1994, van den Berg 1996a) will be needed to fully disambiguate the utterance.

Note that non-restrictive components do constitute predications on the set x', but they do not contribute to the construction of x'. First the set of farmers is introduced as a value of x', *then* of this set it is claimed that the elements are stupid. It is non-restrictive because it does not help construct the restriction.

Adjectival Quantifiers

Another well known example is given by "adjectival" uses of quantifiers. This works the other way around. Normally, when the speaker starts a story with *Three men entered the bar* she does not claim there were only three men. It just means that she has some amusing anecdote about three particular men. Adjectival uses can be formalized by quantifiers inside the maximizers:

(40) $\exists x', x\ \ \mathsf{M}_{x'}\,[\mathbf{man}(x')] \wedge \mathsf{M}_{x}\,[\mathbf{enterbar}(x) \wedge \mathbf{3}(x', x)]$

Note an interesting fact from a linguistic point of view. in this perspective, the difference between the quantificational (*exactly three*) reading and the adjectival (*at least three*) reading is based on a general mechanism, and no *hidden indefinite* is needed to explain "movement" of the quantifier to adjectival position.

In this perspective, the difference between the collective my theory and that of van der Does (1992) may seem to become really marginal. However, do not forget that I claim that such "movements" are actually the result of the original utterances being different, and that this difference can often be heard in the intonation.

Contrastive Stress

A phenomenon that falls into the same category is contrastive stress: *the RED roses are in the FRONT garden and the WHITE roses in the BACK garden*. We can deal with this by assuming that only the stressed adjectives are part of the maximization, and the non-stressed nouns are not. The non-stressed material can be considered to be presupposed, for example by there being a question *What roses grow in what garden?* In the terminology of Prüst et al. (1994), the non-stressed parts determine the *schema* that the Discourse Grammar uses to establish discourse coherence: *the X roses are in the Y garden*.

But here we are treading more and more onto the territory of the next two chapters. For the moment, I will just assume that the translation of the sentence magically results in a representation that has the right form. In chapter 5, I will then give some of the magic away, and discuss these phenomena, as well as Cumulatives and other non-standard scope phenomena, which can also be considered cases of this kind of structural variation of the quantificational phrase.

3 Independent Plural Logic

Introduction

In this section a mechanism is proposed to model the phenomenon of *dependent variables*. I will define an alternative way of interpreting formulas, relative to sets of assignments rather than assignments. This adds the possibility to not only relate sets of entities, as is done in the plural logic defined so far, but to relate the elements of such sets of entities. In the next chapter, it will turn out that the logical machinery defined here has a further dynamic motivation. There, this approach turns out to be so natural that I will start with the dynamic intuitions to argue for this framework. In this section I will restrict the possible ways in which the logic can change these assignments, and the net result will be that the logic we get is equivalent with the logic defined in the previous section. I will only give the definitions to the extent that they are needed for the proof of this equivalence. A more extensive discussion can be found in the next chapter [4:2.2–2.4].

It may seem a bit strange to define a new and very complex logical system, only to arrive at something that is equivalent with something much simpler we already had. But the surprising thing is, that when this logic is made dynamic along the same lines as the construction of FDPL_s in chapter 2, a large number of dependencies between elements of plural objects that exist in natural language are automatically introduced without extra effort. The resulting logic is much more powerful than the result of making the logic of the previous chapter dynamic, illustrating the sometimes forgotten wisdom, that theory change does not commute with predictive-equivalence of theories.

3.1 Arguments from Dynamics

I will start with some arguments from discourse semantics that illustrate why we need to be able to express dependencies between variables. Take the following example and three possible continuations of it:

(41) *Every man loves a woman.*
  a. *They prove this by giving them flowers.*
  b. *Therefore, every man gives her flowers.*
  c. *and she loves him right back.*

This illustrates that singulars can be used to refer to plurals, indirectly in the case of *a woman* (41), and directly, it seems, in the case of *her* (41b) and *she* (41c), which both refer to the set of women introduced in the antecedent. Or so it seems.

In the antecedent sentence, at least in one reading, *a woman* really *is* a singular: for every man, there is one woman. Only afterwards, there will be a set of women, just as there will be a set of men. The elements of the set of men stand in a functional relation (f) with elements of the set of women and the elements of the set of women stand in a functional relation with subsets of the set of men (f^{-1}).

Anaphors referring back to *a woman* in (41) can either be plural (41a), in which case the (plural) anaphor refers back to the set of women as a whole, or singular in the scope of an operator that replays the original distribution, giving the respective women for the men (41b), or the anaphor can just be singular (41c), in which case the women are picked up one at the time (re-distributed). Example (41c) is probably better known as the chess-piece example:

(42) *Every chess-set comes with a spare pawn. It is glued on the back of the lid.*

Exactly under what circumstances the singular can be used to refer to a set, is determined by the discourse grammar that translates the actual linguistic utterance into its logical representation. This issue goes beyond the scope of this thesis. I restrict myself here to one important observation. The singular pronoun can be used if it is anaphoric to something that itself is expressed using a singular, irrespective of the actual cardinality of the value of the corresponding discourse referent (syntactic number agreement). The important subcases are illustrated by examples (41 b–c). A singular anaphor can refer to an antecedent that is in the scope of a distributive quantifier (as in 41b) or forms itself the singular restriction of a distributive quantifier (as in 41c). In these cases that anaphor will itself be distributive.

But let us return to static logic, and leave the dynamic issues to chapters 4 and 5. The important fact that example (41) illustrates is that dependencies may exist between variables. These dependencies may relate the individual elements of one set to the individual elements of another.

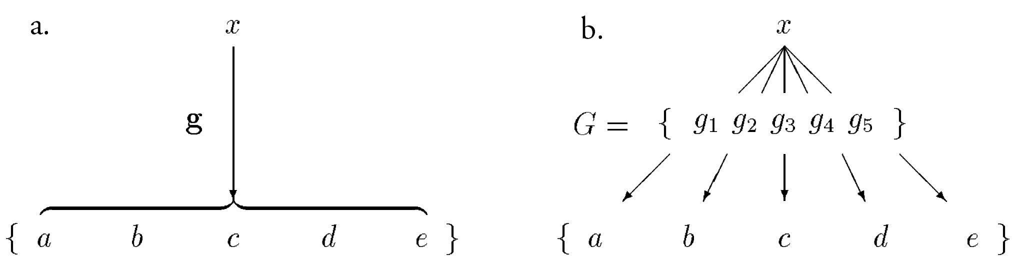
3.2 Dependencies in Static Logic

Sets of Assignments

The previous discussion suggests the following image. Elements of the value of one variable may depend on elements of values of other variables. If the elements of a variable y (say, the women in (42)) in some way are dependent on the elements of a variable x (the men in (42)), then distributing over x will also divide up the values of y.

The way dependencies will be encoded is by interpreting formulas not relative to a single assignment **g** that assigns a set to a variable, but to a set of assignments G, each of which assigns only one, singular element. From now on, I will use (43b) instead of (43a), which we used before.

(43)



Formulas are interpreted by giving essentially the same definitions I gave before, but now using the sets of assignments. I will from now on refer to sets of assignments as **states**.

First of all, let us see what the relations are between plural assignments and states Every state G corresponds to one assignment $\mathrm{Ass}(G)$ that assigns sets via the definition:

$$\mathrm{Ass}(G)(x) := \{g(x) \mid g \in G\} \text{ for every } x.$$

This is something that has to be done so often, that I will always write $G(x)$ for $\mathrm{Ass}(G)(x)$, essentially pretending that G is an assignment:

$$G(x) := \{g(x) \mid g \in G\} \text{ for every } x.$$

Conversely, the definition is not that easy. There are infinitely many states G that assign the same values to all variables as one particular plural assignment $\mathbf{g}$, where assigning the same values means that $\mathbf{g} = \mathrm{Ass}(G)$. It is however possible to come up with a distinguished element.

3.1 Definition (Canonical Representation)
*the **canonical representation** of* $\mathbf{g}$ *is a state defined by.*

(44) $\mathrm{Can}(\mathbf{g}) := \{g \mid g(x) \in \mathbf{g}(x) \text{ for every } x\} = \prod_{x \in \mathrm{VAR}} \mathbf{g}(x).$

When a state G is the canonical representation *of some state, I will call it a **canonical state**.*

The following properties can be easily seen to hold

3.2 Fact
If $g \in \mathrm{Can}(\mathbf{g})$, then
(i) For $d \in D$ we have $g[x := d] \in \mathrm{Can}(\mathbf{g}[x := D])$
(ii) For $D \subseteq E$ we have $\mathrm{Can}(\mathbf{g}[x := D]) \subseteq \mathrm{Can}(\mathbf{g}[x := E])$

There are two reason why this is called the canonical representation. First, this is the largest state G such that $\mathbf{g} = \mathrm{Ass}(G)$[10]. And secondly, it represents exactly the same information on the relationship between elements of sets as the original plural assignment (namely none). To understand this last point, I will have to explain how states can be used to encode relations between elements of sets.

Encoding Dependencies

Suppose we want to encode the relationship between the men and the women in *Every man loves a woman*. How would go about this? Let us assume that the men are $john$, $bill$, $charles$ and tom, and the women are $mary$, sue and $joan$. Suppose that the love-relationships between them are as given by (45a). Then we can represent this by a state $G = \{g, h, k, l\}$ as shown in (45b).

	a. the relationships	b. as assignments	
	john loves mary	$g(x) = john$	$g(y) = mary$
(45)	*bill loves sue*	$h(x) = bill$	$h(y) = sue$
	charles loves mary	$k(x) = charles$	$k(y) = mary$
	tom loves joan	$l(x) = tom$	$l(y) = joan$

It should be clear from this, that G does encode the relationships between the men and women. The next question is, whether it is possible to make use of this extra information. The answer to this is that in static logic not much use can be made of it, as I will show in a moment. However,

10. Suppose there were a larger set of assignments G' such that $\mathbf{g} = \mathrm{Ass}(G')$, then there has to be an assignment $g \in G' - G$. But for this assignment, $g(x) \in \mathbf{g}(x))$ for all x, hence it is in G, contrary to the assumption. Therefore the original G has to be the largest set.

I can give some idea of how these state interact with the other operators to give some interesting new properties. To be able to do that, I first define a number of notions and operations that will be useful in defining interpretations of formulas relative to these states.

A notion that is very useful when defining δ and π, is the notion of an induced subset. Given a variable x, $d \in G(x)$ and $D \subseteq G(x)$, it makes sense to ask which parts of G assign the value d or the values D to x. The following two operations answer this question[11]:

3.3 Definition (Induced Subsets)
The projection on an induced subset of G is defined by:

$$G|_{x=d} = \{g \in G \mid g(x) = d\},$$
$$G|_{x \in D} = \{g \in G \mid g(x) \in D\},$$

Given this definition, it is now possible to define a distribution operator that preserves the dependencies between variables[12]:

(46) $\|\delta_x(\phi)\|^G = \forall d \in G(x)\ \ \|\phi\|^{G|_{x=d}}$

Because this operator divides up the state in partitions induced by the elements of the value of a given variable x, all dependent variables vary corresponding to x. This can be illustrated by means of of the earlier example of women loving men. Suppose the state is as given above, repeated here in a slightly different format:

(47)

	x	y
g	*john*	*mary*
h	*bill*	*sue*
k	*charles*	*mary*
l	*tom*	*joan*

now suppose that we want to interpret ... *he admires her* ... in

Every man loves a woman and he admires her too.

Now suppose that by some discourse means that will be made clear in the next chapter, we know that this means calculating $\|\delta_x(\mathbf{admire}(x, y))\|$. Then definition (46) states that we should calculate $\|\mathbf{admire}(x, y)\|^{G|_{x=d}}$ for every $d \in G(x)$. This means that the interpretation involves the following sets

11. Of course, $G|_{x=d}$ is equivalent to $G|_{x \in \{d\}}$. The single value case occurs so much more often than the set value case, that it makes sense to give it its own, shorter, notation.

12. Let me stress that this is a different definition from the earlier one. However, it does intend to encode the same notion as before and I will prove equivalence with the earlier notion in lemma 3.10, which seems to warrant the same notation.

$$\begin{array}{ll} G|_{x=john}(x) = \{john\}, & G|_{x=john}(y) = \{mary\} \\ G|_{x=bill}(x) = \{bill\}, & G|_{x=bill}(y) = \{sue\} \\ G|_{x=charles}(x) = \{charles\}, & G|_{x=charles}(y) = \{mary\} \\ G|_{x=tom}(x) = \{tom\}, & G|_{x=tom}(y) = \{joan\} \end{array}$$

The result is that **admire** is calculated for each man-woman pair independently. The women are co-distributed with the men. That the case may not always be that simple shows another example that I repeat here:

Every man loves a woman and she loves him right back.

We can assume that by similar discourse means as before, *she* is given a distributive interpretation, and we have to calculate $\|\delta_y(\mathbf{love}(y,x))\|$. But now the correspondence is not one-to-one. The y-induced division of G does not result in singleton sets corresponding to x:

$$\begin{array}{ll} G|_{y=mary}(x) = \{john, charles\}, & G|_{y=mary}(y) = \{mary\} \\ G|_{y=sue}(x) = \{bill\}, & G|_{y=sue}(y) = \{sue\} \\ G|_{y=joan}(x) = \{bill\}, & G|_{y=joan}(y) = \{joan\} \end{array}$$

Of course, *him* brings its own distribution operator, unraveling the set, but that is another issue.

The moral from this is that distributing over one variable causes an induced pseudo-distribution of the others. This leads to the following definition of dependency between variables:

3.4 Definition (Dependency)
*A variable y is **dependent** on a variable x with respect to a state G if there are elements $d, e \in G(x)$, such that $G|_{x=d}(y) \neq G|_{x=e}(y)$.*
*In general, states G and H **express the same dependencies**, written as $G \doteq H$, iff for all variables $x_1, \ldots, x_n$, y and values $d_1, \ldots, d_n$ $G|_{x_1:=d_1} \cdots|_{x_n:=d_n}(y) = H|_{x_1:=d_1} \cdots|_{x_n:=d_n}(y)$.*

In the following section, a notion of variable binding is given that always introduces a variable x in a way that makes it *independent* of all other variables.

3.3 Independent Plural Predicate Logic

I will now define an interpretation of formulas of predicate logic relative to states that is equivalent to the plural predicate logic (PPL) defined in the earlier part of this chapter. This may seem strange, because it employs a new and really complex mechanism to do what we already had. However, as already announced at the beginning of this section, when this logic is made dynamic in the next section the correct dependency relations, like those suggested in section 3.2 will in fact fall into our lap.

Auxiliary Notions

Before we do this, there is one small problem we have to solve. In the interpretation of PPL formulas, the assignments $\mathbf{g}$ may assign the empty set to a variable. However, in the case of states G, which are sets of total assignments, every set assigned will contain at least one element. The solution to this is to make it possible for the elements of the state *not to assign* elements. In other words, we define states to be sets of partial assignments. This means that the definition of assignment to a variable has to be revised somewhat

3.5 Definition (Assignment to Set)

A set of partial assignments G assigns a set to a variable x

(48) $G(x) := \mathrm{Ass}(G)(x) := \{g(x) \mid g \in G \;\&\; g(x) \neq \star\}$.

Given a plural assignment $\mathbf{g}$*, the (minimal) canonical state is defined by*

(49) $\mathrm{Can}(\mathbf{g})(x) := \{g \mid \forall x : g(x) \in \mathbf{g}(x)$ or $(\mathbf{g}(x) = \emptyset \;\&\; g(x) = \star)\}$.

And the maximal canonical state is defined by

(50) $\mathrm{Can}^{\max}(\mathbf{g})(x) := \{g \mid \forall x : g(x) \in \mathbf{g}(x)$ or or $g(x) = \star\}$.

As in chapter 2, I use $\star$ to denote that a particular assignment does not give a value. Note that this definition proposes two versions of a canonical state. The two are equivalent in the sense that they express the same dependencies (namely none). We can now prove the following lemma.

3.6 Lemma[13]

Let $x_1, \ldots, x_n$ be some variables, G and $H \subseteq G$ some states, and H' is the state defined by: $H' := \{h[x_1 := \star]\ldots[x_n := \star] | h \in H\}$. In that case adding H' to G does not make a real difference: G and $G \cup H'$ express the same dependencies: $G \doteq G \cup H'$.

That $\mathrm{Can}(\mathbf{g}) \doteq \mathrm{Can}^{\max}(\mathbf{g})$ is an easy corollary. The two definitions for canonical state are special because $\mathrm{Can}(\mathbf{g})$ is the smallest state that assigns the same values to the variables as $\mathbf{g}$ and expresses no dependencies, and $\mathrm{Can}^{\max}(\mathbf{g})$ is the largest such state. From now on, I will exclusively use $\mathrm{Can}(\mathbf{g})$ to denote the canonical state. Given that they express the same dependencies, I could have chosen the other, but a number of definitions that I give later would have to be changed, and in general would result in more complicated expressions. However, it may be that in some cases using the maximal form would result in simpler expressions. For example, the complex mechanism needed in chapter 5, page 178, to add variables together may not be needed when the maximal version is used.

The properties of Can expressed by fact 3.2 hold also for this partialized version, as do the following:

---

13. For any $x_1, \ldots, x_n$, y and values $d_1, \ldots, d_n$: $(G \cup H')|_{x_1:=d_1} \cdots |_{x_n:=d_n}(y) =$
$G|_{x_1:=d_1} \cdots |_{x_n:=d_n}(y) \cup H'|_{x_1:=d_1} \cdots |_{x_n:=d_n}(y) = G|_{x_1:=d_1} \cdots |_{x_n:=d_n}(y)$.
The last identity because $H'|_{x_1:=d_1} \cdots |_{x_n:=d_n}(y) \subseteq G|_{x_1:=d_1} \cdots |_{x_n:=d_n}(y)$.

3.7 Fact
If $g \in \mathrm{Can}(\mathbf{g})$*, then*
(i) $g[x := \star] \in \mathrm{Can}(\mathbf{g}[x := \emptyset])$
(ii) For D *we have* $\mathrm{Can}(\mathbf{g}[x := \emptyset]) \subseteq \mathrm{Can}(\mathbf{g}[x := D])$

In PPL, variables are never dependent on each other. If I want to preserve this in this logic, I have to demand that no quantifier will ever introduce dependencies. This means in particular that the definition of $\exists x$ should be such that it only introduces states that do not make x dependent on any of the other variables. The consequence of this is that the most straightforward definition of states being the same up to a variable x does not work:

(51) $$G \approx_x H \text{ iff } \forall d \; \{g[x := d] | g \in G\} = \{h[x := d] | h \in H\}, \quad \text{iff } G, H \text{ equal upto } x$$

because an existential quantifier defined in terms of this would introduce arbitrary states, which may introduce dependencies between variables.

To illustrate this, take the state made in the following way. Let $\mathbf{g}$ be the assignment that assigns $\{a, b\}$ to x, and c to all other variables. Now consider Can($\mathbf{g}$). It consists of two assignments g_1 and g_2, such that

$$g_1(x) = a, g_1(y) = c, g_1(z) = c$$

$$g_2(x) = b, g_2(y) = c, g_2(z) = c$$

Now suppose I *did* define the existential quantifier $\exists'$ as

$$\|\exists' x\phi\|_{\mathrm{i}}^{G} \text{ iff } \exists K \approx_x G \; \|\phi\|_{\mathrm{i}}^{K}.$$

Now consider $\exists' y(x = y \wedge \phi)$, with $x = y$ defined by $x \subseteq y \wedge y \subseteq x$. This binds y to the same values as x, and interprets ϕ relative to that assignment. Now with this definition of the existential quantifier, it would be possible that the state in which the scope of the quantifier $(x = y \wedge \phi)$ is interpreted is in fact

$$g_1'(x) = a, g_1'(y) = a, g_1'(z) = c$$
$$g_2'(x) = b, g_2'(y) = b, g_2'(z) = c$$

Whereas the canonical state, in which y is independent from x, would consist of the four assignments $g_1', \ldots, g_4'$:

$$g_1'(x) = a, g_1'(y) = a, g_1'(z) = c$$
$$g_2'(x) = a, g_2'(y) = b, g_2'(z) = c$$
$$g_3'(x) = b, g_3'(y) = a, g_3'(z) = c$$
$$g_4'(x) = b, g_4'(y) = b, g_4'(z) = c$$

If the interpretation in terms of states is to exactly mirror the interpretation in terms of assignments, we better have only canonical states be involved. Let me remind you again, that although it might seem that introducing dependencies is exactly the extra power that would make a logic like this interesting, this is not my goal here.In the next chapter, this non-introduction of dependencies on a local, static level will correspond to exactly the right kind of dependencies being introduced on a more global, dynamic level. What I try to get rid of here are *spurious* dependencies that are not rooted in language, but in accidental properties.

What we need is a variant of $G \approx_x H$ saying that H is given new values (compared to G) for x, not disturbing the values assigned to other variables and not introducing any relationships with these. The following definition does exactly that.

(52) $H \Leftarrow_x G$ iff $\forall d \in H(x)\ H|_{x=d} \approx_x G\ \&\ H \approx_x G$

Note that this is an *asymmetric* relation, to express that H is made out of G by introducing a new value for x that is independent of all existing variables. It does this by demanding that for every value d of x, H is equal to G up to x, The result is that H is the largest set assignments equal to G up to x, but no constraints are put on G, nor can there be: G is assumed to be given. The second conjunct ($H \approx_x G[x := d]$) is a consequence of the first if $H(x)$ is non-empty. If $H(x)$ is empty, it makes sure that the other variables are assigned the same values with the same dependencies. This form of the definition anticipates the dynamic definition in the next chapter.

The relation $\Leftarrow_x$ can also be given in terms of a transformation $G[x := D]$ on states , for D a, possibly empty, set of entities and x a variable, as follows

(53) $G[x := D] := \{h \mid h(x) \in D\ \&\ \exists g \in G\ \ h \approx_x g\}$

(54) $G[x := \emptyset] := \{h \mid h(x) = \star\ \&\ \exists g \in G\ \ h \approx_x g\}$

(55) $H \Leftarrow_x G$ iff $\exists D\ \ H = G[x := D]$

Which illustrates how the new state H is made out of the old state G.

Note that this preserves canonicity, as the following shows:

3.8 Lemma[14] *If G is canonical, then $G[x := D]$ is canonical.*

Except for the need to built in independence explicitly, the definition of the logic is as before:

---

14. Assume G canonical, i.e. for some $\mathbf{g}$, $G = \mathrm{Can}(\mathbf{g})$. then $g \in G$ can be rewritten as $\forall y.g(y) \in \mathbf{g}(y)$. And consequently, $\exists g \in G\ \ h \approx_x g$ as $\forall y.(h(y) \in \mathbf{g}(y)$ or $y = x)$.
Let D be non-empty, then $G[x := D] = \{h \mid \forall y.h(x) \in D\ \&\ (h(y) \in \mathbf{g}(y)$ or $y = x)\}$. This is equivalent to $G[x := D] = \{h \mid \forall y.h(y) \in \mathbf{g}[\mathbf{x} := \mathbf{D}](y)\}$. Hence $G[x := D] = \mathrm{Can}(\mathbf{g}[x := D])$.
For $D = \emptyset$ we proceed analogously: $G[x := \emptyset] = \{h \mid \forall y.h(x) = \star\ \&\ (h(y) \in \mathbf{g}(y)$ or $y = x)\}$. equivalent to $G[x := \emptyset] = \{h \mid \forall y.h(y) \in \mathbf{g}[\mathbf{x} := \emptyset](y)\}$. Hence $G[x := \emptyset] = \mathrm{Can}(\mathbf{g}[x := \emptyset])$.

3.9 Definition (Independent Plural Predicate Logic)
The syntax of IPPL *is that of ordinary predicate logic with identity. The semantics of* IPPL *is the interpretation over a model* $\langle \mathcal{D}, \mathcal{I} \rangle$ *relative to sets of assignments.*

(56) $\|Px_1 \ldots x_n\|_{\mathrm{i}}^{G}$ iff $<G(x_1), \ldots, G(x_n)> \in \mathcal{I}(P)$

(57) $\|x \subseteq y\|_{\mathrm{i}}^{G}$ iff $G(x) \subseteq G(y)$

(58) $\|\phi \wedge \psi\|_{\mathrm{i}}^{G}$ iff $\|\phi\|_{\mathrm{i}}^{G}$ and $\|\psi\|_{\mathrm{i}}^{G}$

(59) $\|\neg\phi\|_{\mathrm{i}}^{G}$ iff not $\|\phi\|_{\mathrm{i}}^{G}$

(60) $\|\exists x \phi\|_{\mathrm{i}}^{G}$ iff $\exists K \Leftarrow_x G\ \|\phi\|_{\mathrm{i}}^{K}$

And the plural modifiers

(61) $\|\delta_x(\phi)\|_{\mathrm{i}}^{G}$ iff $\forall d \in G(x)\ \|\phi\|_{\mathrm{i}}^{G|_{x=d}}$

(62) $\|\pi_x(\phi)\|_{\mathrm{i}}^{G}$ iff $\forall d \in G(x)\, \exists D \subseteq G(x)\ \left[d \in D\ \&\ \|(\phi)\|_{\mathrm{i}}^{G|_{x\in D}}\right]$

(63) $\|\mathsf{M}_x(\phi)\|_{\mathrm{i}}^{G}$ iff $\|\phi\|_{\mathrm{i}}^{G}\ \&\ \neg\, \exists G' \Leftarrow_x G\ \left[G(x) \subsetneq G'(x)\ \&\ \|\phi\|_{\mathrm{i}}^{G'}\right]$

The same auxiliary notions as in PPL can be defined:

(64) $\mathbf{empty}(x) := \forall y(y \subseteq x \rightarrow y = x)$

(65) $\mathbf{sing}(x) := \forall y(y \subseteq x \rightarrow (\mathbf{empty}(y) \vee y = x))$

(66) $\mathbf{plur}(x) := (x = x)$

The operators δ, π and M are defined directly as interpretation relative to states. The reason that we cannot copy the definitions given before is that all of the earlier definitions involve quantification, and quantifiers always introduce variables that are independent of the other variables, whereas the validity of formulas in the scope of δ, π and M may crucially depend on existing dependencies. However, this is again an example of preparing for things to come in chapter 4. In this logic, there never are dependencies, as the following lemma shows (proof in section 5.5 below):

3.10 Lemma
Within IPPL*, the* PPL *definitions of* δ_x*,* π_x *and* M_x *are still valid. For any formula* ϕ *of* IPPL*, that does not contain* y *or* z*:*

$$\delta_x(\phi) \iff \forall y \subseteq x(\mathbf{sing}(y) \rightarrow \phi[x/y]).$$

$$\pi_x(\phi) \iff \forall y \subseteq x(\mathbf{sing}(y) \rightarrow \exists z \subseteq x(y \subseteq z \wedge \phi[x/z]))$$

$$\mathsf{M}_x(\phi) \iff \neg\, \exists y\ (x \subsetneq y \wedge \phi[x/y]) \wedge \phi$$

This means that we can get away with the original definitions. We now stated the whole logic. Given this lemma, we can interpret a formula ϕ either as a formula of the original logic ϕ (PPL) or as a formula of this logic (IPPL) We can now compare the two. It turns out that they are essentially equivalent, as the following proposition shows:

3.11 Proposition[15] **(Equivalence of PPL and IPPL)**
(i) For every formula ϕ and plural assignment $\mathbf{g}$ *the following holds*

$$\|\phi\|_{\mathrm{i}}^{\mathrm{Can}(\mathbf{g})} \text{ iff } \|\phi\|^{\mathbf{g}}$$

(ii) For every ϕ and canonical state G, the following holds

$$\|\phi\|_{\mathrm{i}}^{G} \text{ iff } \|\phi\|^{\mathrm{Ass}(G)}$$

(iii) For every ϕ with no free variables, and arbitrary state G, the following holds

$$\|\phi\|_{\mathrm{i}}^{G} \text{ iff } \|\phi\|^{\mathrm{Ass}(G)}$$

Part (i) and (ii) say that a formula is true in a canonical state, iff it is true in the plural assignment that generates this state. Part (iii) says that even if the formula is interpreted in a non-canonical state, the formula is true in the corresponding plural assignment, provided none of the dependencies expressed by the state is actually used.

Conclusion

The above discussion seems to give a lot of smoke for almost no fire. The logic defined turns out to be equivalent to the original one, and the only immediate advantage seems that interpretation relative to states uses standard *first order models* again. But giving up the *first order interpretation* of formulas may seem a big price for this. And predicates did not become simpler, they are still interpreted as n-place relations between sets. Also, there still is a subset structure to worry about (now as subsets caused by the states). However, as I announced before, in the next chapter, this interpretation structure will be put to good use to explain a number of different discourse phenomena, some of which I already hinted at. Because the way this plural logic will be made dynamic involves the partial dynamic logic of the previous chapter, it makes sense to see how partiality can contribute to plural logic. This is what I will do now.

---

15. The proof is by induction over the formula. (i) the base cases are easy: $\|P\vec{x}\|^{\mathbf{g}} \iff \mathbf{g}(\vec{x}) \in \mathcal{I}(P) \iff \mathrm{Can}(\mathbf{g})(\vec{x}) \in \mathcal{I}(P) \iff \|P\vec{x}\|_{\mathrm{i}}^{\mathrm{Can}(\mathbf{g})}$ (and the same for $\subseteq$).
$\|\phi \wedge \psi\|^{\mathbf{g}} \iff \|\phi\|^{\mathbf{g}}$ & $\|\psi\|^{\mathbf{g}} \iff \|\phi\|_{\mathrm{i}}^{\mathrm{Can}(\mathbf{g})}$ & $\|\psi\|_{\mathrm{i}}^{\mathrm{Can}(\mathbf{g})}$ (ind.) $\iff \|\phi \wedge \psi\|_{\mathrm{i}}^{\mathrm{Can}(\mathbf{g})}$
$\|\neg\phi\|^{\mathbf{g}} \iff$ not $\|\phi\|^{\mathbf{g}} \iff$ not $\|\phi\|_{\mathrm{i}}^{\mathrm{Can}(\mathbf{g})}$ (ind.) $\iff \|\neg\phi\|_{\mathrm{i}}^{\mathrm{Can}(\mathbf{g})}$.
$\|\exists x\phi\|^{\mathbf{g}} \iff \exists D\ \|\phi\|^{\mathbf{g}[x:=D]} \iff \exists D\ \|\phi\|_{\mathrm{i}}^{\mathrm{Can}(\mathbf{g}[x:=D])} \iff \exists D\ \|\phi\|_{\mathrm{i}}^{\mathrm{Can}(\mathbf{g})[x:=D]}$
$\iff \exists K \Leftarrow_x \mathrm{Can}(\mathbf{g})\ \|\phi\|_{\mathrm{i}}^{K} \iff \|\exists x\phi\|_{\mathrm{i}}^{\mathrm{Can}(\mathbf{g})}$.
(ii) This is a direct corollary of (i). Because G is canonical ($\mathrm{Can}(G) = \mathbf{g}$), we have $\|\phi\|_{\mathrm{i}}^{\mathrm{Can}(\mathbf{g})}$ iff $\|\phi\|^{\mathrm{Ass}(\mathrm{Can}(\mathbf{g}))}$ and Ass is a left inverse of Can: $\|\phi\|_{\mathrm{i}}^{\mathrm{Can}(\mathbf{g})}$ iff $\|\phi\|^{\mathbf{g}}$.
(iii) This is proved in section 5.5.

4 Partial Plural Logic

Introduction

In chapter 2, three ways of partializing logic where discussed. Partializing the dynamic effect, partializing the truth values and partializing the states. In a static logic, only the last two apply, and in this section I will discuss both of them. Although partializations of states have already been touched upon to arrive at the equivalence, there is much more to them than just being helpful in defining the empty set.

Different partializations of quantifier logic were discussed by van Eijck (1991b) to implement uncertainty or to formalize vagueness. Given that quantifiers are relations between sets, it is important to study the logic that you get when set membership is not always known. All of these are examples of three-valued logics where the third value represents the underspecified element [1:2.5 *p*.25].

The partializations in chapter 2 serve a different purpose. They are used to exclude irrelevant information, to exclude "noise". In this section I will extend that treatment to plural logic. In the first part of this section, I will show how partiality can be used to further formalize the special nature of the restriction of a quantifier. In the second part, partialization is used to implement a powerful notion of subset in the logic. No new technical insights are presented here, these are standard extensions of predicate logic that are extensively discussed in the literature. I will restrict myself to some observations that will be relevant for the next chapter.

4.1 Partial Logic

In chapter 1, I discussed a strict-partial logic extension (SPL) of standard predicate logic. For PPL, we can copy this directly, because it is nothing but standard predicate logic over distinguished (plural) domains. The logic SPPL, defined in the previous section is similar, but here we have to worry about how to partialize the modifiers δ_x, π_x an M_x, which in this logic get an independent definition. Before I do this, I will first discuss why partialization of the logic makes independent sense for plural quantifier logics.

The restriction of a quantifier determines the domain of quantification. In van van Benthem's (1986a) words, it "sets the stage" for the quantification. The notion of conservativity in generalized quantifier theory is always argued for by referring to the special character of the quantifier restriction. To interpret $Q(A, B)$, we only have to look at that part of B that is also in A, hence the equivalence of $Q(A, B)$ and $Q(A, A \cap B)$.

In chapter 1 I discussed the strict partial logic SPL for which $\{\neg, +, \wedge\}$ is a functionally complete set of operators. The Presupposition operator $+\phi$ also turns out to be of good use to Quantifier theory to express the special nature of the quantifier restriction.

If we write $\|\phi\|_d^G$ to express that ϕ *is defined in* G and $\|\phi\|_+^G$ to express that ϕ *is true in* G, the definition of strict-partial plural logic, including the presupposition operator, looks as follows[16]:

16. Cf. chapter 1, section 2.5, where it is explained why for this kind of partial logic, it makes more sense to define the **defined** and **true** parts than to define **true** and **false** parts. Also compare chapter 2, section 5.2 to see the computational

4.1 Definition (Strict-partial Plural Predicate Logic)
The syntax of SPPL *is that of* PPL. *The semantics of* SPPL *is the interpretation over a model* $\langle \mathcal{D}, \mathcal{I} \rangle$ *relative to sets of assignments.*

(a) $\|Px_1 \ldots x_n\|_d^{\mathbf{g}}$ always
$\|Px_1 \ldots x_n\|_+^{\mathbf{g}}$ iff $<\mathbf{g}(x_1) \ldots \mathbf{g}(x_n)> \in \mathcal{I}(P)$

(b) $\|x \subseteq y\|_d^{\mathbf{g}}$ always $\quad \|x \subseteq y\|_+^{\mathbf{g}}$ iff $\mathbf{g}(x) \subseteq \mathbf{g}(y)$

(c) $\|\neg\phi\|_d^{\mathbf{g}}$ iff $\|\phi\|_d^{\mathbf{g}}$ $\quad \|\neg\phi\|_d^{\mathbf{g}}$ iff $\|\phi\|_d^{\mathbf{g}}$ and not $\|\phi\|_+^{\mathbf{g}}$

(d) $\| + \phi\|_d^{\mathbf{g}}$ iff $\|\phi\|_+^{\mathbf{g}}$ $\quad \| + \phi\|_+^{\mathbf{g}}$ iff $\|\phi\|_+^{\mathbf{g}}$

(e) $\|\phi \wedge \psi\|_d^{\mathbf{g}}$ iff $\|\phi\|_d^{\mathbf{g}}$ and $\|\psi\|_d^{G}$ $\quad \|\phi \wedge \psi\|_+^{\mathbf{g}}$ iff $\|\phi\|_+^{\mathbf{g}}$ and $\|\psi\|_+^{\mathbf{g}}$

(f) $\|\exists x\phi\|_d^{G}$ iff $\exists \mathbf{h} \approx_x \mathbf{g}\ \|\phi\|_d^{\mathbf{h}}$ $\quad \|\exists x\phi\|_+^{\mathbf{g}}$ iff $\exists \mathbf{h} \approx_x \mathbf{g}\ \|\phi\|_+^{\mathbf{h}}$

The presupposition operator can be used to express the intuition about quantifiers that whether true or false, the quantification is about its restriction. Furthermore, the value corresponding to the nuclear scope should be a subset of the value corresponding to the restriction, whether or not it is under a negation. This too can be implemented using this operator. For example, the sentence

(67) *Four women entered the bar*

is about women whether true or not. We can implement this by applying $+$ to the restriction of the quantifier.

$$\exists x', x\ \ \mathsf{M}_{x'}(+\mathbf{w}(x')) \wedge \mathsf{M}_x(+x \subseteq x' \wedge \mathbf{e}(x)) \wedge \mathbf{4}(x', x).$$

This says, as it did before, that x' will have the largest set satisfying $\mathbf{w}(x')$ as its value. However, in

$$\neg\, \exists x', x\ \ \mathsf{M}_{x'}(+\mathbf{w}(x')) \wedge \mathsf{M}_x(+x \subseteq x' \wedge \mathbf{e}(x)) \wedge \mathbf{4}(x', x)$$

x' will still have a satisfying $\mathbf{w}(x')$ as its value. It is immune for negation. This becomes important in the dynamic case, where the quantifier will introduce discourse referents into the context, and it is crucial that the referent introduced is of an appropriate kind (in this case: women). In a similar way, $+x \subseteq x'$ ensures that the set x is a subset of x', and that only the content part of the nuclear scope ($\mathbf{e}(x)$) is sensitive to the negation.

Adding Dependencies

If we want to partialize IPPL, we can start with definition 4.1. Only its definition of the existential quantifier has to be changed to allow for interpretation relative to states, and the change is minimal:

---

background of this intuition.

4.2 Definition (Strict-partial Independent Plural Predicate Logic)
The syntax of SIPPL *is that of* IPPL. *The semantics of* SIPPL *is the interpretation over a model* $\langle \mathcal{D}, \mathcal{I} \rangle$ *relative to sets of assignments.*

(a) $\|Px_1 \dots x_n\|_d^G$ always
$\|Px_1 \dots x_n\|_+^G$ iff $<G(x_1) \dots G(x_n)> \in \mathcal{I}(P)$

(b) $\|x \subseteq y\|_d^G$ always $\quad \|x \subseteq y\|_+^G$ iff $G(x) \subseteq G(y)$

(c) $\|\neg\phi\|_d^G$ iff $\|\phi\|_d^G$ $\quad \|\neg\phi\|_d^G$ iff $\|\phi\|_d^G$ and not $\|\phi\|_+^G$

(d) $\|+\phi\|_d^G$ iff $\|\phi\|_+^G$ $\quad \|+\phi\|_+^G$ iff $\|\phi\|_+^G$

(e) $\|\phi \wedge \psi\|_d^G$ iff $\|\phi\|_d^G$ and $\|\psi\|_d^G$ $\quad \|\phi \wedge \psi\|_+^G$ iff $\|\phi\|_+^G$ and $\|\psi\|_+^G$

(f) $\|\exists x\phi\|_d^G$ iff $\exists K \Leftarrow_x G \ \|\phi\|_d^K$ $\quad \|\exists x\phi\|_+^G$ iff $\exists K \Leftarrow_x G \ \|\phi\|_+^K$

In IPPL the modifiers are no longer defined in terms of the other logical operators, so these too have to be given a definition. The following are relatively conservative extensions to the partial domain of the distribution and pseudo-distribution. Following the example of conjunction and existential quantification, we define them by using the total-definition for both components.

(g) $\|\delta_x(\phi)\|_d^G$ iff $\forall d \in G(x) \ \|\phi\|_d^{G|_{x=d}}$
$\|\delta_x(\phi)\|_+^G$ iff $\forall d \in G(x) \ \|\phi\|_+^{G|_{x=d}}$

(h) $\|\pi_x(\phi)\|_d^G$ iff $\forall d \in G(x)\, \exists D \subseteq G(x) \ \left[d \in D \ \& \ \|\phi\|_d^{G|_{x \in D}}\right]$
$\|\pi_x(\phi)\|_+^G$ iff $tre\forall d \in G(x)\, \exists D \subseteq G(x) \ \left[d \in D \ \& \ \|\phi\|_+^{G|_{x \in D}}\right]$

Maximization cannot be given this simple form. The following is **wrong**:

(68) $\|\mathsf{M}'_x(\phi)\|_d^G$ iff $\|\phi\|_d^G \ \& \ \neg\, \exists G' \approx_x G \ (G(x) \subsetneq G'(x) \ \& \ \|\phi\|_d^{G'})$
$\|\mathsf{M}'_x(\phi)\|_+^G$ iff $\|\phi\|_+^G \ \& \ \neg\, \exists G' \approx_x G \ (G(x) \subsetneq G'(x) \ \& \ \|\phi\|_+^{G'})$

because the states for which the formula is defined and for which it is true will in general be disjoint, which constitutes an unacceptable situation. Any true formula should be a defined formula by definition. A definition that at least satisfies *this* is given by:

(69) $\|\mathsf{M}'_x(\phi)\|_d^G$ iff $\|\phi\|_d^G$
$\|\mathsf{M}'_x(\phi)\|_+^G$ iff $\|\phi\|_+^G \ \& \ \neg\, \exists G' \approx_x G \ (G(x) \subsetneq G'(x) \ \& \ \|\phi\|_+^{G'})$

Where it is only demanded that the argument of M_x is defined, to preserve strictness of the logic, but nothing more. A set satisfying $\neg\mathsf{M}_x(Px)$ may very well be a set satisfying Px, provided it is not the largest such set. For me, this goes a bit against the spirit of what $\mathsf{M}(Px)$ is supposed to be. If it is the closest we can come in the language to *the set satisfying* P, then the negation should be the complement of that set. If $\mathcal{I}(P) = A$ and the universe is E, then the negation of $\mathsf{M}(Px)$ should be $E - A$. That is also what makes the most sense linguistically, as we will see in chapter 5; so this is what I will define. Because here the positive and negative parts are the ones that are directly arrived at, and the defined part is just the disjunction of these, I will give the definition in terms of $+$ and $-$ rather than $+$ and d.

(i) $\|\mathsf{M}_x(\phi)\|_+^G$ iff $\neg\,\exists G' \approx_x G\ (G(x) \subsetneq G'(x)\ \&\ \|\phi\|_+^G)$

$$\|\mathsf{M}_x(\phi)\|_-^G \text{ iff } \neg\,\exists G' \approx_x G\ \Big(G(x) \subsetneq G'(x)\ \&\ \|\phi\|_-^G\ \&\ \neg\,\exists G'' \approx_x G\ (G''(x) \subset G'(x)\ \&\ \|\phi\|_+^{G'}\Big)$$

The definition of M_x may look a bit complicated, but is in fact very intuitive. The maximized version of a formula is true for a set A assigned to x by G if that formula is not true for some larger G', that assigns a larger set to x. So much for the part that is not changed. The falsity clause should result in the complement of the maximal set A. If something satisfies the formula, the maximal set for which the formula does not hold will in general be the whole universe: a combination of some elements (or subsets) for which the formula holds and some for which it does not hold. So what we do instead is look for the largest set B such that this set does not contain parts for which the formula holds. The definition states that all larger sets will contain a subset for which the formula is true. The following fact gives some more formal content to the reason why this definition makes sense.

4.3 Fact[17]
If ψ is defined for all (singular) values of x, ϕ is $\delta_x(\psi)$ and G, H are such that $\|\mathsf{M}_x(\phi)\|_+^G$ and $\|\mathsf{M}_x(\phi)\|_-^H$, then, for E the set of entities in the model
(i) $G(x) \cup H(x) = E$ and
(ii) if $G \approx_x H$ then $G(x) \cap H(x) = \emptyset$.

This finishes the discussion of partiality of the truth-values for this chapter. But there is a second kind of partiality that can be applied to the logic: partiality of the assignments. This is the next subject.

---

17. The proof is simple. Observe that if ψ is defined for all singleton sets assigned to x, ϕ is defined on all sets.
(i) Suppose $G(x) \cup H(x) \subsetneq E$, then there is an entity $d \in E$ that is not in the union. By assumption, $\|\phi\|_+^{G[x:=d]}$ or $\|\phi\|_-^{G[x:=d]}$ is defined. Suppose $\|\phi\|_+^{G[x:=d]}$, then by the fact that ϕ is distributive: $\|\phi\|_+^{G \cup G[x:=d]}$, hence G was not maximal in x, contradicting the assumption. So it has to be that $\|\phi\|_-^{G[x:=d]}$, but this leads to $\|\phi\|_-^{G \cup G[x:=d]}$ by the same argument. So the original assumption was incorrect, and $G(x) \cup H(x) = E$ after all.
(ii) If the intersection is non-empty, there is an object d both in G and H. Then $\|\phi\|^{+G[x:=d]} \iff \|\phi\|^{+H[x:=d]}$ (because $G \approx_x H$) so there is a subset of H that makes the formula true, contradicting the fact that $\|\mathsf{M}_x(\delta_x(\phi))\|_-^H$ implies that no subset makes ϕ true.

4.2 Partial Assignments

In definition 3.5, I introduced partial assignments to make it possible to have a state assign the empty set to a variable. There is a second, completely independent reason, why partial assignments should be introduced, and that is to give a better treatment of subsets.

In the section on dependencies, one thing is missing. If we introduce a variable as the subset of a set, then this new variable will have no relationship with the old one. However, it is possible to define a mechanism that preserves all dependencies that exist when a subset is taken.

If states are sets of partial assignments, it is possible to define a notion of pointwise subset. This tests whether a variable y is a subset of a variable x, and whether the two variables agree on the relations their values have with other variables, in as far as the values of x and y overlap.

(70) $\|y \subseteq^{\bullet} x\|^G$ iff $\forall g \in G \;\; g(x) = g(y)$ or $g(y) = \star$.

We can illustrate this with a simple example. Suppose that in a given situation, there are four people in the room, two men and two women, and each of them has a dog. The state that gives x the people as value, y only the women and z the dogs could look like this:

people	*women*	*dogs*
$g(x) =$ john	$g(y) = \star$	$g(z) =$ fido
$h(x) =$ marlies	$h(y) =$ marlies	$h(z) =$ bruno
$k(x) =$ rens	$k(y) = \star$	$k(z) =$ spot
$m(x) =$ cathy	$m(y) =$ cathy	$m(z) =$ boris

This is a notion of subset $y \subseteq x$ that is stronger than just demanding $G(y) \subseteq G(x)$. It demands that the dependencies of the variable expressing the set x are preserved into the variable y.

Note that for the definition of the distribution operator: $\|\delta_x(\phi)\|^G$ iff $\|\phi\|^{G|_{x=d}}$ for every $d \in D$, the quantification is only over defined values d. This means that the assignments that assign an undefined value are essentially ignored in the distribution. It is important to realize that this also ignores all the values that these assignments assign to other variables, which need not be undefined.

In the next chapter, I will show that in a dynamic logic the IPPL definitions do in fact give rise to exactly the kind of dependencies described here between the discourse referents.

5 Logical Issues

Introduction

I will now, as in the preceding chapter, discuss some more formal considerations related to the subject matter of this chapter.

First, a little detour is in order. In 5.1 I discuss some of van der Does's theoretical insights, to make it easier to compare what I am doing here with his work, and give some background to the definitions. Then I return to the main line of the chapter, and discuss some of the properties of the generalized quantifiers as defined in this chapter. In section 5.4 I turn my attention to the dependency structures that turned to be surprisingly independent, as shown by the equivalence proof 3.11 that I finish here, proving that the new interpretation scheme results in the same logic as the original plural logic.

5.1 Van der Does, 1992

To a large extent, the theory of generalized quantification (GQT) was developed under the assumption that the objects in the domain of quantification are indivisible units (atoms). But as is well-known the ontology of natural language allows for much more diversity, as it involves the description of groups and sums; events, processes, and states; facts and situations; intervals, times, and cases; and so forth, and so further. An important question is therefore: how to adapt GQT to this diversity? In section 3 of chapter 1, the basic theory was discussed, so let us look at how we can move to plurals from these.

Structured Domains

Once we have adopted the global view of a quantifier as a functor, it is quite simple to allow for different kinds of *structured domains*; just restrict the functor to the relevant ones. For example, Lewisian cases are tuples of entities over a domain E. So, the quantifier $\mathbf{Q}^n$, which quantifies over n-ary cases can be obtained by setting (Westerståhl 1989):

$$\mathbf{Q}^n_E RS \iff \mathbf{Q}_{E^n} RS$$

Case quantification, i.e., quantification over arbitrary cases, arises in the limit:

$$\mathbf{Q}^\omega_E = \bigcup_{n\in\omega} \mathbf{Q}^n_E$$

Along similar lines we get some base material to study plural quantification. For instance, when quantifying over *sums* (= sets of objects) we may define the relation among sets of sets $\mathbf{Q}^s$ by:

$$\mathbf{Q}^s_E \mathbf{XY} \iff \mathbf{Q}_{\wp(E)} \mathbf{XY}$$

Once this step is taken, we may consider quantifiers $\mathbf{Q}^{s:n}$ over *sums of depth* n by setting:

$$\mathbf{Q}^{s:n}_E \mathcal{XY} \iff \mathbf{Q}_{\wp^n(E)} \mathcal{XY}$$

where $\wp^0(X) = X$ and $\wp^{n+1}(X) := \wp(\wp^n(X))$. In the limit we find $\mathbf{Q}^g$ which quantifies over groups, i.e., sums of finite depth:

$$\mathbf{Q}^g_E \mathcal{GG}' \iff \exists n \in \omega : \mathbf{Q}^{s:n}_E \mathcal{GG}'$$

Of course, restricting the functors to special domains in the way indicated comes with considering special variants of the constraints CONS, EXT, ISOM, ..., and of the special properties MON, CONT, ... We shall consider some of these variants for quantifiers $\mathbf{Q}^s$ ($= \mathbf{Q}^{s:1}$) over sums in Section 5.1. But first we give some further suggestions concerning readings.

Collective readings

There are many more possibilities for collectives than the one reading considered in the text. The basic intuition behind a collective reading is that part of the common noun satisfies the verb phrase as a unit; cf. (70).

(71) a. *Most of the people joined forces*
b. *Four committees had a joint lunch*

Observe that (71.a) has a distributive noun, $\mathbf{people} \subseteq \mathbf{at}(E)$, which pertains to singletons only. Sentence (71.b) has a collective noun, $\mathbf{committee} \subseteq \mathbf{n}-\mathbf{at}(E) := \wp(E) - \mathbf{at}(E)$, pertaining to non-singletons. a case which we did not considered in this chapter. It is also worth noticing that here collectivity is enforced by modification in the object NP.

Depending on their main subject, sentences like those in (71) may describe the relevant noun elements in an indefinite, a universal, a descriptive or even a referential way. This leads to the four proposals for collective readings listed in (72).

(72)
existential	$\exists\mathbf{Z} \subseteq \mathbf{X}[\mathbf{Y}(\bigcup\mathbf{Z}) \wedge \mathbf{Q}^s_E\mathbf{X}\mathbf{Z}]$
universal	$\forall\mathbf{Z} \subseteq \mathbf{X}[\mathbf{Y}(\bigcup\mathbf{Z}) \rightarrow \mathbf{Q}^s_E\mathbf{X}\mathbf{Z}]$
descriptive	$\exists!\mathbf{Z} \subseteq \mathbf{X}[\mathbf{Y}(\bigcup\mathbf{Z}) \wedge \mathbf{Q}^s_E\mathbf{X}\mathbf{Z}]$
referential	$\mathbf{C} \subseteq \mathbf{X} \wedge \mathbf{Y}(\bigcup\mathbf{C}) \wedge \mathbf{Q}^s_E\mathbf{X}\mathbf{C}$

Here, **C** is a contextually given set of sums, much as defined by Westerståhl (1984). Moreover, there are maximized and minimized versions of these proposals, which are obtained as before by applying the operators MIN and MAX to the verbal part.

$$\mathbf{min}(\mathbf{X}) = \lambda Y.\text{`}Y\text{ is inclusion minimal in }\mathbf{X}\text{'}$$
$$\mathbf{max}(\mathbf{X}) = \lambda Y.\text{`}Y\text{ is inclusion maximal in }\mathbf{X}\text{'}$$

All in all, this gives eight possible candidates for the collective reading! Still, life isn't that simple, for each of these readings can be used to generate distributive, pseudo-distributive, and neutral readings by modifying the verbal argument using δ, π, or ν, respectively. How to choose among all these possibilities? We shall propose some criteria now.

Choice criteria

One way to choose among the many possible formalizations is to check one's semantical intuitions concerning sentences with plural NPs. This is a subtle affair, which quickly leaves the impression that the data can almost be stretched as we please. For this reason it is more fruitful to engage in a logical investigation. This should show more clearly what the proposals are up to if we were to accept them. It also gives the following criteria for choosing. A lifting of a quantifier is acceptable if:

1. it preserves important constraints and properties;
2. it validates reasonable inferences.

In the following we shall have a look at each of the criteria in turn. We limit ourselves to quantification over sums.

Constraints for sums

The first thing we shall look for are variants of the general constraints for quantifiers $\mathbf{Q}^s$; i.e., relations between sets of sets. Since we only consider domains of form $\wp(E)$ there is not much room to play in case of EXT. But for CONS and ISOM the situation is quite different.

Conservativity As to CONS we can stay close to the notions discerned in (van der Does 1992). Besides the standard (73a), which $\mathbf{Q}^s$ inherits from $\mathbf{Q}$, there are the variants (73b-e) that may be satisfied by different liftings to relations among sets of sets:

(73) a. $\mathbf{Q}^s\mathbf{XY} \iff \mathbf{Q}^s\mathbf{XY} \cap \mathbf{X}$
b. $\mathbf{Q}^s\mathbf{XY} \iff \mathbf{Q}^s\mathbf{XY} \cap \mathrm{AT}(\mathbf{X})$
c. $\mathbf{Q}^s\mathbf{XY} \iff \mathbf{Q}^s\mathbf{XY} \cap \wp(\bigcup \mathbf{X})$
d. $\mathbf{Q}^s\mathbf{XY} \iff \mathbf{Q}^s\mathbf{X}\mathbf{max}(\mathbf{Y} \cap \mathbf{X})$
e. $\mathbf{Q}^s\mathbf{XY} \iff \mathbf{Q}^s\mathbf{XY} \sqcap \mathbf{X}$

Here we used the following operations:

$$\mathrm{AT}(\mathbf{X}) \iff \{\{d\} : \{d\} \subseteq \mathbf{X}\}$$
$$\mathbf{Y} \sqcap \mathbf{X} \iff \bigcup_{X \in \mathbf{X}} \mathbf{Y} \cap \wp(X)$$

The liftings introduced in this chapter satisfy: For the collective: (73e), for the distributive (73a) and (73b) and for the pseudo-distributive (73c).

Isomorphy There seem to be at least two variants of ISOM depending on the nature of the permutations we allow. On the one hand we my demand closure under arbitrary permutations $\Pi : \wp(E) \to \wp(E)$. Or we could restrict ourselves to permutations $\pi : E \to E$ lifted to $\pi^* : \wp(E) \to \wp(E)$ by $\pi^*(X) := \{\pi(X) : X \in \mathbf{X}\}$ with $\pi(X) := \{\pi(d) : d \in X\}$. As it happens, the first variant demands closure under too many permutations, and is hence too restrictive. The second version is the most reasonable one.

Preservation of properties

A natural criterion to choose among the many liftings is to prefer those that induce natural higher-order variants of lower level properties. For example, a left monotone quantifier as a relation among sets should induce a relation among sets of sets which is also somehow left monotone. In this respect $\mathbf{Q}^s$ behaves nicely. It preserves all the relevant properties, since it is the same quantifier as the lower level one but restricted to a special kind of domain. For our maximized lifting the situation is different. It preserves the version of ISOM with lifted permutations, and it has a notion of conservativity built in but it seems to destroy most of the other properties. E.g, monotonicity does not seem to make sense in this area, and similarly for the other notions. It remains to be done to look for alternative properties, which are more suitable for the maximized setting.

Logical properties

The lifted quantifiers on top of a standard plural logic (cf. Lønning (1995)) will induce quantifier logics, for which all familiar questions can be asked. Which ones are (in)complete? And which decidable? Which quantifier formulas preserve which model constructions, and can they be characterized up to logical equivalence? Etc. etc. ...

As in Van der Does 1992, we can also ask for the logical relationships among the liftings, and whether we can characterize the lifted quantifier in the higher type. The case of the first question will be discussed in section 5.2.

As to the second question, let us concentrate on a slightly simplified lifting, where the first argument is the power of a set (recall that this corresponds to the assumption that the noun is distributive).

$$\mathrm{M}(\mathbf{Q})_E\wp(X)\mathbf{Y} \iff \exists Z[Z \in \mathbf{max}(\mathbf{Y} \cap \wp(X)) \wedge \mathbf{Q}_E XZ]$$

This lifting satisfies the notion of conservativity in (73a,c). It also induces a form of maximized monotonicity:

$$\mathrm{M}(\mathbf{Q})_E\wp(X)\mathbf{Y} \;\; \& \; \mathbf{max}(\mathbf{Y} \cap \wp(X)) \subseteq \mathbf{max}(\mathbf{Z} \cap \wp(X)) \rightarrow \mathrm{M}(\mathbf{Q})_E\wp(X)\mathbf{Z}$$

In fact we have a form of maximized continuity:

$$\mathrm{M}(\mathbf{Q})_E\wp(X)\mathbf{Y} \iff \exists Z \in \mathbf{max}(\mathbf{Z} \cap \wp(X))\mathrm{M}(\mathbf{Q})_E\wp(X)\{Z\}$$

As for the lift C *simpliciter*, there is the simple observation: the MAX-continuous quantifiers are precisely those quantifiers in type $(ett)(ett)t$ which are of the form $\mathrm{M}(\mathbf{Q})$ for some $\mathbf{Q}$ in type $(et)(et)t$.

This ends the digression into the world of van der Does.

5.2 Formal Properties

Properties of maximization

The maximization operator as defined here has a number of properties that may not agree with everyone, especially where the collective modes of quantification are concerned. One point that may raise an eyebrow is the fact that the maximum need not be unique. I will have something to say about this in chapter 5. Here I just want to observe that uniqueness can of course always be demanded afterwards as a pragmatic constraint, and can then be seen to guide interpretation: *it has to be unique, so the speaker means one specific one of the possible ones.* The non-uniqueness will then encode the possibilities that the hearer can still choose from.

The definition of quantification given above are most natural on finite domains. On infinite domains, curious properties may arise. Take $\mathbb{N}_{<\omega} := \{V \mid \exists n.\#V = n\}$, the set of all finite sets of natural numbers. This does not have a maximum, there is always a larger set, and the set of natural numbers, the external maximum of $\mathbb{N}_{<\omega}$, is not a member of $\mathbb{N}_{<\omega}$. Of course, the fact that the collective of all elements of finite sets is not itself a finite set is not really surprising. It is hard to imagine how a natural language expression, even over an infinite domain, may look, that has exactly the required properties to have this effect. Let me remind you: this is a theory about quantification over individuals within a language capable of talking about sets, it is not a theory of quantifications over sets. The latter is a different subject. For example, even the predicate **gather** expresses a property about entities, although it is a property of entities doing something together (also cf. Verkuyl (1994)). Think what the above mathematical example would mean in that perspective. What property of individuals might this set express. It is not the property *being an element of a finite set*, because that property happily admits for infinite sets in its extension. It also cannot characterizes those individuals that are members of some finite set, but not an infinite set. Because the set of all those entities is itself infinite: shades of the Russell paradox, if you ask me. This set then expresses a property of sets, that cannot be readily expressed as a property of the individuals. Note that if we take the most advanced version of the maximization operator, the partial definition, then it can be seen that $\mathsf{M}_x(\mathbb{N}_{<\omega}(x))$ is never true, and false for x the empty set and no other set. This does seem reasonable behavior to me. If there is no maximum, it gives up. This may not be what is most acceptable in mathematics. But before you start thinking about changing the definition, first think about the simpler question raised above: might the case you are thinking of not better be treated using quantifiers over sets? And even if you have a reason to not want to do this, you should note that the problem only arises for the collective case, which is not the most natural variety for mathematical quantifiers. The maximum of (pseudo-) distributives will always exist and be unique, and these are the "really quantificational" expressions.

5.3 Logical Properties

In this discussion, I will limit attention to plural cases distributive, pseudo-distributive and collective ($\mathcal{O} = \delta, \pi, \mathbf{1}$):

(74) $\mathbf{Q}^{\mathcal{O}}(\mathbf{X}, \mathbf{Y}) \iff \exists A \in \mathcal{O}(\mathbf{X}), A \in \mathcal{O}(\mathbf{Y})\ \ Q(A, B).$

The following properties, some of which were mentioned before, can be easily seen to hold

5.1 Fact

- *No entity gets lost: if* $d \in \bigcup \mathbf{X}$ *then there is a* $A \in \mathsf{M}(\mathbf{X})$ *such that* $d \in A$.
- $\mathsf{M}(\pi(\mathbf{X}))$ *and* $\mathsf{M}(\delta(\mathbf{X}))$ *are singleton sets.*
- $B \in \mathsf{M}(\pi(\mathbf{X})) = \{\bigcup \mathbf{X}\}$.
- *If* $A \in \mathsf{M}(\mathbf{X})$ *and* $B \in \mathsf{M}(\pi(\mathbf{X}))$ *then* $A \subseteq B$.
- *If* $A \in \mathsf{M}(\delta(\mathbf{X}))$ *and* $B \in \mathsf{M}(\pi(\mathbf{X}))$ *then* $A \subseteq B$.

Using these properties, it is not difficult to prove the following lemma:

5.2 Lemma[18]

if a plural generalized quantifier $\mathbf{Q}$ *is a variety of a generalized quantifier* Q*, then they have the same monotonicity properties (expressed as implications between quantificational expressions).*

Furthermore, if $\mathbf{X}$ is distributive, i.e. $\mathbf{X} = \wp(A)$, then $\mathsf{M}(\mathbf{X}) = \mathsf{M}(\delta(\mathbf{X})) = \mathsf{M}(\pi(\mathbf{X})) = A$. This is something I will assume the NP to satisfy. This means that the difference between readings resides in the VP. It also means that the fact that the modifier is always the same for NP and VP may seem more like a fashion statement than anything else. However, the dynamic case will prove otherwise.

If the quantifier is MON$\uparrow$, (74) gives

(75) collective $\rightarrow$ pseudo-distributive $\leftarrow$ distributive

and if the quantifier is MON$\downarrow$, the "sign" of these relations swaps around:

(76) collective $\leftarrow$ pseudo-distributive $\rightarrow$ distributive.

These facts are not surprising. In van der Does (1992) (75) is already mentioned: our distributive is exactly his D_1, the pseudo-distributive is almost identical to N_2 and the collective is even more selective than his C_2^a. The relation (76) is even a bit stronger then the one that van der Does gives, because of the maximization by the collective.

5.4 Collective Dependency Models

In the second part of this chapter, we have motivated the use of sets of assignments as a new kind of 'collective' semantic states, encoding various useful dependencies between variables. Actually, such sets suggest various natural mathematical notions of dependency. Here is one example (out of several). We say that variable y functionally depends on the set of variables X if, whenever the values of X are fixed, so is that of y . Put differently, given any assignment g in G, changing its y-value inside G automatically involves changing one of the X-values. It is easy to see that this notion of dependence may be described using a set of structural rules very much like those governing classical inference. This dependence relation is monotonic in its X argument, and it satisfies the Cut Rule. For instance, if y depends on X and z depends on U, while v depends on $\{y, z\}$, then v depends on $X \cup U$.

By contrast, the standard semantics of collectives and plurals uses assignments mapping variables to sets of objects (cf. the survey of Lønning (1995) and the first part of this chapter). Our current semantics is richer. In set-theoretic terms, from a repeated power $(2^D)^V$, we go to 2^{D^V}.

---

18. For π and δ the proof is simple. In both cases, the maximum is unique. In the distributive case, $A = \bigcup(\mathbf{at}(\mathbf{X}))$ and $B = \bigcup(\mathbf{at}(\mathbf{X} \cap \mathbf{Y}))$, in the pseudo-distributive case, $A = \bigcup(\mathbf{X})$ and $B = \bigcup((\mathbf{Y} \cap \wp \bigcup \mathbf{X}))$. It is now easy to see that if Q is $\uparrow$MON, then so is $\mathbf{Q}$, and the same for other monotonicity properties: Suppose Q is $\uparrow$MON, and suppose $\mathbf{X}' \subseteq \mathbf{X}$, hence $\bigcup \delta(\mathbf{X}') \subseteq \bigcup \delta(\mathbf{X})$. Then $\mathbf{Q}(\delta(\mathbf{X}), \delta(\mathbf{Y})) \iff Q(\bigcup \delta(\mathbf{X}), \bigcup \delta(\mathbf{Y}))$ (def.) $\Longrightarrow$ $Q(\bigcup \delta(\mathbf{X}'), \bigcup \delta(\mathbf{Y}))$ (by $\uparrow$MON) $\iff \mathbf{Q}(\delta(\mathbf{X}), \delta(\mathbf{Y}))$ (def.). The other cases are similar.
For the collective readings, the proof is only a little more complicated. We have to take into account that their may not be a unique maximum. Consider Q $\uparrow$MON. By definition $\mathbf{Q}(\mathbf{X}, \mathbf{Y})$ holds if for some $A \in \mathsf{M}(\mathbf{X}), B \in \mathsf{M}(\mathbf{Y} \sqcap \mathbf{X})$: $Q(A, B)$. Now suppose $\mathbf{X} \subseteq \mathbf{X}'$. Now either A is also maximal in $A \in \mathsf{M}(\mathbf{X}')$ or it is not. If it is, we trivially have (cf. page 112) $\mathbf{Q}(\mathbf{X}, \mathbf{Y} \sqcup \mathbf{X}) \iff \mathbf{Q}(\mathbf{X}', \mathbf{Y} \sqcup \mathbf{X})$. If A is not maximal, there has to be a $A' \in \mathsf{M}(\mathbf{X}')$ such that $A \subseteq A'$. Then for that A', $Q(A', B)$ by $\uparrow$MON of Q.
The other cases are again similar.

Generally speaking, the latter will be much larger in size — which reflects our greater freedom for encoding dependencies between objects assigned. Even so, as we have seen, there are natural connections between these two domains. The following maps were defined to send plural assignments $\mathbf{g}$ to sets $\mathrm{Can}(\mathbf{g})$ of individual assignments and vice versa:

$$\mathrm{Can}(\mathbf{g}) = \{h \mid \forall x \;\; h(x) \in \mathbf{g}(x)\}$$

$$\mathrm{Ass}(G) = \lambda x\{g(x) \mid g \in G\}.$$

The map Can delivers special 'full' sets of assignments. It is 1-1, unlike the map Ass, which is a left-inverse of Can:

$$\mathrm{Ass}(\mathrm{Can}(\mathbf{g})) = \mathbf{g}$$

One can design various formal languages over these collective dependency models. For instance, in our chapter so far, we found a number of operators which satisfy the scheme

$M, G \models \mathcal{O}_x\phi$ *iff for some* H *such that* $R(G, H)$, $M, H \models \phi$.

or its dual ($\mathcal{R}_x = \neg\mathcal{O}_x\neg$) version

$M, G \models \mathcal{R}_x\phi$ *iff for every* H *such that* $R(G, H)$, $M, H \models \phi$.

In the first place, this is the existential quantifier (which inspired this scheme to begin with)

$M, G \models \exists x\phi$ *iff for some* D *such that* $H = G[x := D]$, $M, H \models \phi$.

Distribution and pseudo-distribution have this (dual) form:

distribution $M, G \models \delta_x\phi$ *iff for every* H *such that there is a* $d \in G(x)$ *and* $H = G|_{x=d}$, $M, H \models \phi$.

pseudo-partition $M, G \models \pi_x\phi$ *iff for every* $d \in G(x)$ *there is a* $D \subseteq G(x)$ *such that* $H = G|_{x \in D}$, $M, H \models \phi$.

Note that distribution has an obvious dual ∇_x, defined by $\nabla_x := \neg\delta_x\neg$:

$M, G \models \nabla_x\phi$ *iff for some* $d \in G(x)$ *such that* $H = G|_{x=d}$, $M, H \models \phi$.

Expressing a notion of guilty by association: a set satisfies $\nabla_x\phi$ if one member satisfies the ϕ. Pseudo-distribution does not have a dual that is that natural, although we can of course define the dual $\amalg_x := \neg\pi_x\neg$.

Our earlier equivalence proposition, which I will prove in a moment (section 5.5), says that a language with just these quantificational operators sees no difference between the two levels of plural representation.

There are a number of general questions that arise when such richer interpretation models are introduced. The first question might concern a worry about the complexity. Can we find the normal first-order quantification back in this theory? An important partial answer to this is given in the equivalence theorem. For all sentences (formulas with no free variables) $M, G \models \phi$ iff $M, \mathrm{Ass}(G) \models \phi$ (cf. section 5.5 below), because the dependencies are erased by the quantifiers before they can be used.

Generalized Assignment Models

Another way of looking at our plural models links them up with current research on *generalized assignment models* for predicate logic (cf. Németi 1993). These models interpret first-order languages in the following format:

$$M, G, g \models \phi \qquad \text{(singular state } g \text{ verifies } \phi \text{ in 'plural context' } G\text{)},$$

where G is a set of *available assignments* or *relevant computational states*. In line with the general program set up in this chapter, one could use these assignment sets G themselves as indices of evaluation:

$$M, G \models \phi \qquad \text{(plural state } G \text{ itself verifies formula } \phi\text{)}$$

where formulas ϕ may involve new logical operators, exploiting the richer structure of collective states.

In particular, the linguistic analysis of this chapter suggests at least two kinds of existential quantification, both reflecting natural transition relations over states:

> *$G \models \exists_{\text{coll}} x \phi$ iff ϕ is true at some G' with $G \approx_x G'$, i.e., G, G' have the same assignments up to values for the variable x,*
>
> *$G \models \exists_{\text{ind}} x \phi$ iff ϕ is true at some G' with $G \approx^1_x G'$, i.e., G, G' have the same assignments up to values for the variable x, but all x-values in G' are set to one object.*

The resulting modal logic encodes a theory of interaction between individual and collective quantification. It can be explored via the standard technique of *modal frame correspondences*, with axioms reflecting structural properties of and connections between the above two types of accessibility relation, say, $R_{\text{coll},x}$ and $R_{\text{ind},x}$ (for all variables x). Here are a few illustrations.

Some Examples of Plural Frame Correspondences

Once we take the modal perspective, we can go all the way and inspect what kind of accessibility relations correspond to different kinds of quantifiers.

a. *All $R_{\text{coll},x}$ are equivalence relations. Hence, each quantifier $\exists_{\text{coll}} x$ satisfies S5.*

b. *The $R_{\text{ind},x}$ are not equivalence relations. They are transitive, but not reflexive or symmetric. Thus, we have that if $\exists_{\text{ind}} x(\exists_{\text{ind}} x \phi)$, then also $\exists_{\text{ind}} x \phi$, but if ϕ, it need not be the case that $\exists_{\text{ind}} x \phi$, as ϕ might be true as a collective assertion, but not for any individual value of x.*

c. *There are also interactions between $R_{\text{coll},x}$ and $R_{\text{ind},x}$. Each $R_{\text{ind},x}$ is contained in $R_{\text{coll},x}$, and therefore if $\exists_{\text{ind}} x \phi$ holds, then $\exists_{\text{ind}} x \phi$ does also hold (truth for an individual is a boundary case of collective truth). The converse fails, of course, there is no need why if something holds for an arbitrary set, it will also hold for an individual.*

d. *Next, $R_{\text{coll},x}$ followed by $R_{\text{ind},x}$ reduces to an $R_{\text{ind},x}$ step. Next, $R_{\text{ind},x}$ followed by $R_{\text{coll},x}$ reduces to an $R_{\text{coll},x}$ step. The same is true with universal quantifiers: the last occurrence always counts.*

e. *Interactions between relations with different variable index occur, too. Their corresponding logical quantifier laws include various forms of Permutation:*
$\exists_{\text{coll}}x\exists_{\text{coll}}y\phi$ *iff* $\exists_{\text{coll}}y\exists_{\text{coll}}x\phi$, $\exists_{\text{coll}}x\exists_{\text{ind}}y\phi$ *iff* $\exists_{\text{ind}}y\exists_{\text{coll}}x\phi$, *etc.*

f. *and forms of Confluence:*
if $\exists_{\text{coll}}x\forall_{\text{ind}}y\phi$ *then* $\forall_{\text{ind}}y\exists_{\text{coll}}x\phi$ *and if* $\exists_{\text{ind}}x\forall_{\text{ind}}y\phi$ *then* $\forall_{\text{ind}}y\exists_{\text{ind}}x\phi$.

In this connection, an important modeling option must be mentioned which affects the complexity of our plural logic. We can set up a dynamic logic with transitions over all possible assignments. This will recreate something of at least the complexity of standard first-order logic at our plural level. But the general philosophy of generalized assignments models is that not all mathematically possible assignments need be available. A similar strategy might be followed with respect to 'available' sets of assignments in our plural models. With the latter degree of freedom, the complexity of our plural quantifier logic might be drastically reduced.

Going the other way, in the logic actually discussed in this chapter only really special, canonical sets of assignments where introduced by quantifiers. This need not be the case, and more arbitrary sets, as briefly pointed at in this section, might be introduced. The reason that I did not do this is that all complexity of arbitrary sets will be introduced in the dynamic version, and *caused* by the dynamics. This will constitute a way of introducing "arbitrary" sets of assignments in a controlled way, rather than out of the blue, given some hope that we will be able to keep the complexity under control.

Finally, there are also connections between our approach and the more general 'dependency models' of Alechina 1995, which we must forego here.

5.5 Equivalence of IPPL and PPL (remainder of proof)

Let me end with the proofs of 3.10 and proposition 3.11iii that I still owe you.

First, a simple but useful lemma

5.3 Lemma
if G is canonical, then for any $D \subseteq G(x)$, and $y \neq x$

(i) $G|_{x\in D}(y) = G(y)$ for $y \neq x$
(ii) $G[x := d](y) = G|_{x\in D}(y)$
(iii) $\mathrm{Ass}(G|_{x\in D}) = \mathrm{Ass}(G)[x := D]$

for any $D \subseteq \mathbf{g}(x)$

(iv) $\mathrm{Can}(\mathbf{g})|_{x\in D} = \mathrm{Can}(\mathbf{g}[x := D])$

proof (i) Follows from the fact that G is a canonical representation.
(ii) Because G is canonical: $G|_{x=d}(z) \approx_x G|_{x=e}(z)$ for any $z \neq x$, d and e. So we only need to prove $G[y := d](z) = G(z)$. But this is obvious, changing y will not change z.
(iii) For every y, the left hand side is equal to $\mathrm{Ass}(\{g \in G \mid g(x) \in D\})(y)$, which is equivalent to $\{g(y) \mid g \in G \,\&\, g(x) \in D\}$; the right-hand side as $\mathrm{Ass}(G)[x := D](y)$, which is equivalent to $\bigcup_{d \in D} \{g(y) \mid g \in G \,\&\, g(x) = D\}$. If $y \neq x$, the left-hand side is equivalent to $\{g(y) \mid g \in G\}$, by (i), and the right-hand side $\bigcup_{d \in D} \{g(y) \mid g \in G\} = \{g(y) \mid g \in G\}$, because for an assignment, changing one variable does not change another. This proves (ii).
(iv) $g \in \mathrm{Can}(\mathbf{g})|_{x \in D} \iff g \in \mathrm{Can}(\mathbf{g})$ and $g(x) \in D$ (def. $|_{x \in D}$) $\iff g(y) \in \mathbf{g}$ and $g(x) \in D$ (def. Can), $\iff g(y) \in \mathbf{g}$ and $g(x) \in (G) \cap D \iff g(y) \in \mathbf{g}$ and $g(x) \in (G)[x := D] \iff g \in \mathrm{Can}(\mathbf{g}[x := D])$. **[end proof]**

We are now ready to prove lemma 3.10, repeated here.

5.4 Lemma

Within IPPL, *the* PPL *definitions of* δ_x, π_x *and* M_x *are still valid: For any formula* ϕ *of* IPPL, *that does not contain* y *or* z *and* G *canonical:*

(a) $\quad \|\delta_x(\phi)\|^G \text{ iff } \|\forall y \subseteq x(\mathbf{sing}(y) \to \phi[x/y])\|^G,$

(b) $\quad \|\pi_x(\phi)\|^G \text{ iff } \|\forall y \subseteq x(\mathbf{sing}(y) \to \exists z \subseteq x(y \subseteq z \wedge \phi[x/z]))\|^G,$

(c) $\quad \|\mathsf{M}_x(\phi)\|^G \text{ iff } \|\neg \exists y\,(x \subsetneq y \wedge \phi[x/y]) \wedge \phi\|^G$

proof (a) $\|\forall y \subseteq x(\mathbf{sing}(y) \to \phi[x/y])\|^G \iff \forall D \subseteq G(x) \|(\mathbf{sing}(y) \to \phi[x/y])\|^{G[y:=D]}$ $\iff \forall D \subseteq G(x)(\|\mathbf{sing}(y)\|^{G[y:=D]} \Longrightarrow \|\phi[x/y]\|^{G[y=D]}) \iff \forall d \subseteq G(x)\|\phi[x/y]\|^{G[y:=d]} \iff$ $\forall d \subseteq G(x)\|\phi[x/y]\|^{G|_{y=d}}$ (lemma 5.3) $\iff \forall d \subseteq G(x)\|\phi\|^{G|_{x=d}}$ (rename y back to x) $\iff \|\delta_x(\phi[x/y])\|^G$
(b) $\|\forall y \subseteq x(\mathbf{sing}(y) \to \exists z \subseteq x(y \subseteq z \wedge \phi[x/z]))\|^G$
$\iff \forall d \subseteq G(x)\|\exists z \subseteq x(y \subseteq z \wedge \phi[x/z])\|^{G[y:=d]}$ (as for (a)) $\iff \forall d \subseteq G(x)\exists D \ni d\|\phi[x/z]\|^{G[y:=d][z:=D]}$ (again, as for (a)) $\iff \forall d \subseteq G(x)\exists D \ni d\|\phi[x/z]\|^{G[z:=D]}$ (assumption, y not mentioned in ϕ) $\iff$ $\forall d \subseteq G(x)\exists D \ni d\|\phi\|^{G|_{x=D}}$ (again, as for (a)) $\iff \|\pi_x(\phi)\|^G$
(c) $\|\neg \exists y\,(x \subsetneq y \wedge \phi[x/y]) \wedge \phi\|^G \iff \neg \exists D\,(\|x \subsetneq y\|^{G[y:=D]} \,\&\, \|\phi[x/y]\|^{G[y:=D]}) \,\&\, \|\phi\|^G \iff$ $\neg \exists D \supsetneq G(x).\|\phi[x/y]\|^{G[y:=D]} \,\&\, \|\phi\|^G \iff \neg \exists D \supsetneq G(x).\|\phi\|^{G[x:=D]} \,\&\, \|\phi\|^G$ (rename variable) $\iff$ $\|\mathsf{M}_x(\phi)\|^G$ **[end proof]**

And finally we turn to proposition 3.11iii. The complexity of this proof lies in the fact, that G is allowed to be any state, not just a canonical one. the condition that no variables are free insures that all dependencies encoded in G are destroyed (by the quantifiers binding the variables) before they can do any harm. What I will prove instead is the following lemma. The proof then follows from proposition 3.11ii.

5.5 Lemma

For every ϕ *with no free variables, and state* G*, the following holds*

$$\|\phi\|_{\mathrm{i}}^{G} \text{ iff } \|\phi\|_{\mathrm{i}}^{\mathrm{Can}(\mathrm{Ass}(G))}$$

proof For the basic cases, we have $G(x) = \mathrm{Can}(\mathrm{Ass}(G))(x)$. Hence $G(x) \in \mathcal{I}(P)$
$\iff \mathrm{Can}(\mathrm{Ass}(G))(x) \in \mathcal{I}(P)$.
For conjunction and negation, this is a simple induction.
$\| \exists x\ \phi\|_{\mathrm{i}}^{G} \iff \exists D\ \|\phi\|_{\mathrm{i}}^{G[x:=D]} \iff \exists D\ \|\phi\|_{\mathrm{i}}^{\mathrm{Can}(\mathrm{Ass}(G[x:=D]))} \iff \exists D\ \|\phi\|_{\mathrm{i}}^{\mathrm{Can}(\mathrm{Ass}(G))[x:=D]}$
$\iff \| \exists x\ \phi\|_{\mathrm{i}}^{\mathrm{Can}(\mathrm{Ass}(G))}$.
you may wonder why the condition that ϕ does not contain free variables is not used anywhere. This is a subtle point: If it does contain free variables, the above still holds, but it is no longer possible to reduce δ and π away. It is there that the problem arises: 3.11a assumes that the state G is canonical. [**end proof**]

This finishes the proofs.

Conclusions

In the first part of this chapter I gave a variant of the received theory of plurals. The only real difference with the standard formalization of this theory is the proposal that a form of maximization is involved. This makes a difference for the collective reading, but as the discussion in section 2.4 shows, the original meaning can also be expressed in this logic. I also maintained that MON↓-quantifiers should be defined as negations of their (MON↑) duals, in accordance with earlier proposals (van der Does 1992). This explains the feeling that they quantify over cases, and will, in chapter 5, serve to explain certain dynamic properties that MON↓-quantifiers have.

There are several reasons why an interpretation of quantifiers using **maximization is to be preferred**, but most of these will only be clear in the next chapters. The most important reason has to do with the distinction between E-type pronouns and "normal" bound variables in the standard theory. When maximizations are involved, the difference between these two can be explained by their position alone, inside or outside the maximization operator, dispensing with the need for two different notions. A variable will automatically be "normal" when in the scope of the quantifier in the traditional sense, and automatically be E-type when dynamically bound. In fact, it goes further then this. Exceptions as discussed in section 2.4 illustrate this, although this will only come into its own in the dynamic case.

The second part of this chapter is somewhat more adventurous. I discussed a number of issues that will only come to full bloom in the next chapter. I discussed a mechanism which can encode relations between elements of different plural objects, that consisted of the interpretation of formulas relative to sets of assignments. This work is related to current work by Alechina and van Benthem (1993), Alechina (1995a), Alechina and van Lambalgen (1995), Andréka et al. (1994), amongst others. but the present system is much richer, and can be used to express much more subtle **dependencies between variables**. The discussion in this thesis seems to give additional linguistic evidence for their mathematical considerations. However, I defined these interpretations in such a way that the resulting logic is equivalent with the original one. In this thesis, all dependencies will be caused by dynamic phenomena.

I then discussed two ways of partializing plural logic, parallel to the partializations in chapter 2. These partial logics differ from thAt defined by van Eijck (1991b) because that is mainly concerned with partial information, whereas I am interested in full information from which irrelevant contributions are removed using a third truth value.

I ended the chapter with a few technical issues. I showed that the distribution operators can be seen as particular kinds of modal operators.

4

Dynamic Plural Logic

Introduction

In this chapter, dynamic predicate logic, partial logic, plurals and generalized quantifiers are combined to form one logical system. The combination of dynamic and partial logic was defined in chapter 2; the combination of plural logic, generalized quantifiers and partial logic was defined in chapter 3. It is not necessary to have read either chapter 2 or 3 to understand this chapter, although certain design decisions will only be clear in the light of the discussions in these chapters. On the few occasions where the definition in this chapter is not exactly the same as the corresponding definition in the previous chapters, I will point this out and explain the reason.

This chapter consists of five sections. The first two sections define the formalism. Section 1 repeats the important parts of the findings in chapters 2 and 3, and section 2 discusses the particular nature of the states that this logic is interpreted in. Section 3 combines everything defined so far. It gives a formal definition of the logic and discusses its most important properties. Section 4 defines how quantifiers can be expressed in this logic. As in the previous two chapters, I end with a section 5 discussing the relation of the defined logic with other work and some more formal properties. Then follows the conclusion of this chapter, which among other things contains a map of the logical landscape, relating all the logics defined in this and the previous chapters.

1 Dynamic Semantics for Plural

Introduction

In chapter 1, it was said that to define a dynamic logic, three elements have to be specified, (1) what the states are the logic is about, (2) what basic (atomic) relations between states there are, and (3) what the operators between the relations are. In the next section, I will discuss what the states of the logic are, in this section I will discuss the major conclusions that the previous two chapters make about basic relations and operators. Section 3 then combines the two and defines the logic.

According to the first chapter, the resulting logic combines dynamic predicate logic, partial logic, generalized quantifier theory and plural logic. Let us see how these are combined.

1.1 Relations

An important moral, drawn in chapter 2, is that the atomic relations are partial relations between states, whatever the states are. Without repeating all the arguments, this can be seen by looking at tests. Tests are relations between states that do not change that state. A tests holds in a state rather than between states. Tests behave as loops: they end in the same state as they started out in. Take a simple predicate Px, if it holds in a state g (for DPL, states are assignments: $g(x) \in \mathcal{I}(P)$), then there exists a true transition:

(1) $g \xrightarrow[\oplus]{Px} g$

The point where partiality suggests itself is when the predicate does not hold in the state. In that case the predicate is still interpreted *in* that state. Interpreted as a relation, if the input g is not equivalent to the output h, DPL makes the relation be false, whereas the interpretation as a process that loops on a state would suggest that $R(g, h)$ should be undefined. Only true or false on the diagonal: $R(g, g)$ is true if the test holds in g and false if it does not holds in g. The relations, then, are partial relations. The underlying natural logic of dynamic logic is three-valued, and, given the nature of this third value as *inappropriate*, it is a strict three-valued logic.

In dynamic logic, there is a difference between the truth-value of the relation corresponding to a formula, and the truth conditions of that formula. For DPL, the truth conditions are defined by saying that a formula is true in an input g, if there is an output for g that makes that formula true. The formula is false if there is no such output. For three valued logic this has to be redefined. The chosen redefinition is to say that a formula is true for an input if there is an output making it true, and defined for an input if there is an output for which it is defined. The falsity of a formula is defined in terms of *defined but not true*.

(2) $\|\phi\|_d^g$ iff $\exists h \; g[\![\phi]\!]^d h$
$\|\phi\|_+^g$ iff $\exists h \; g[\![\phi]\!]^+ h$
$\|\phi\|_-^g$ iff $\|\phi\|^d{}_g$ but not $\|\phi\|^+{}_g$

Interestingly enough, the fact that the underlying logic is three-valued does not necessarily mean that the truth-conditions need be three-valued. In particular, if we restrict ourselves to the logical constants of DPL then the truth-conditions will be two-valued [2:lemmas 2.2 and 2.15], and in fact equal to standard DPL. However, the use of three-valued truth-conditions is warranted by, amongst other, the presupposition operator.

1.2 Generalized Quantifiers and Plurals

As illustrated in chapter 1 [1:1], all determiners are capable of picking up referents and introducing new referents. In general, determiners translate as generalized quantifiers, and these have to be linked into dynamic logic to give them the right anaphor+antecedent character. In Kamp and Reyle (1993), the method used is, roughly, as follows. A quantification is a relation between two formulas $Qx(\phi, \psi)$ (called a duplex-condition). The set introduced by this quantification is the set of all d satisfying $\phi \wedge \psi$, and this set is assigned independently of the quantification, which is closed as in DPL [2:1.1]. In a format approximated by the informal notation (not intended to be a formal account of the definition in Kamp and Reyle's book):

(3) $!Qx(\phi, \psi) \wedge X := \lambda x.(\phi \wedge \psi)$

In dynamic logic, this effect can be achieved more directly, by using an alternative formulation of generalized quantifiers [3:2.2 *p*.90]. Rather than defining a new notion of generalized quantifier as a relation between formulas, the starting point is the traditional generalized quantifier Q(A,B) as a relation between sets. The important observation is, that in plural logic, the elements of the domain are plural objects (sets of entities), and predicates are relations between such objects, which is exactly what quantifiers are. Generalized quantifiers are possible meanings of predicates. This can be used to give an alternative form for the definition for the meaning of determiners. A relation $Q(\phi, \psi)$ can be defined by first finding sets satisfying ϕ and ψ, and then demanding that the relation Q holds between these sets. This is of course the meaning that this relation always has. What is new in this formulation is that this can be written in the plural logic itself. It amounts to introducing variables x and x', assuming that ϕ is a formula over x' and ψ a formula over x. And more important for us, this reformulation can be made dynamic by merely reinterpreting the logical constants (though below I will do slightly more than that).

To simplify things, assume that ϕ and ψ are distributive formulas, that is, formulas that are uniquely defined by sets of entities: there is a set A_ϕ such that ϕ holds for x' exactly if the value of x' is a subset of A_ϕ, and similarly for ψ and x. Let $\mathsf{M}'_x(\phi)$ be the formula true exactly if x' is this set (the elements of x' are the entities satisfying ϕ), and $\mathsf{M}_x(x \subseteq x' \wedge \psi)$ similar for $x \subseteq x' \wedge \psi$, the largest subset of x' satisfying ψ. Then it should be clear that the following expression merely rewrites the classic meta-language definition of the generalized quantifier, and effectively *is* the meaning of that generalized quantifier.

(4) $\exists x', x \;\; \mathsf{M}_{x'}(\phi) \wedge \mathsf{M}_x(x \subseteq x' \wedge \psi) \wedge Q(x', x)$

That this can be made dynamic should be obvious. The actual quantifiers are straightforward existential quantifiers, so no new dynamic mechanism need be postulated.

Conclusion

The above sketches the route I will take from here. I will define a plural dynamic predicate logic by making the plural/generalized quantifier theory dynamic using partial dynamic predicate logic. All separate decisions on how to define the components, and how to combine separate parts have been discussed in chapters 2 and 3. What is a bit surprising, maybe, is how easy the combination is, and how powerful the result. In some cases a slightly different format for the formulas will be used, but only if the profit, simplifying the definitions, is big enough.

But before I define this logic, I have to make one final step: define what states the formula are interpreted in.

2 States for Dependent Plurals

Introduction

The above gives the basic relations and logical operators of the logic. It also gives some idea of how the states look: a state has to assign a set to variables to be able to use them for the plural logic. But the exact details of the latter are still unclear. I will now give a number of examples, and use these to argue for a particular form of the states.

2.1 The Nature of States

The states have to be chosen to be able to express all the values assigned to discourse referents, and all the relations between them that occur in language. Most of the choices made in defining plural dynamic logic can be decided on by studying a simple and classic example as an antecedent sentence followed by some not so classic anaphoric clauses:

(5) *Every*^{1} *man loves a*^{2} *woman.*

a. *They*$_1$ *bring them*$_2$ *them*$_2$ *flowers to prove this.*

b. *He*$_1$ *brings her*$_2$ *flowers to prove this.*

c. *Every*$_1$ *old man brings her*$_2$ *a present to prove this.*

d. *The*$_1$ *old men bring them*$_2$ *presents to prove this.*

e. *And they*$_2$ *love them*$_1$ *right back.*

f. *And she*$_2$ *loves him*$_1$ *right back.*

g. *Yesterday, several dogs dug holes in my garden.*

The dynamics should satisfy the following requirements:

a. In (5.a) the plural pronouns *they* and *them* should pick up the sets introduced by their antecedents, despite the fact that these antecedents are syntactically singulars. These sets are then used in the predication, which is ambiguous between a collective or distributive interpretation.

b. In (5.b) the singular pronoun *he* also picks up a set but distributes over it, effectively giving a new distributive universal quantifier meaning something like *every one of the men*. Also, the second pronoun (*her*), picks up exactly the woman loved by every particular men. I.e. there is a functional relationship that is created by the distribution in the antecedent which can be picked up by the anaphor.

c. Moreover, when distributing over a subset, the induced distribution should only be about a subset of the dependent variable: The old men give flowers to their girl-friends.

c′. Another thing that this example suggest is that quantifiers naturally pick up referents of earlier quantifiers (as if they are anaphors), and then introduce a new referent that is a subset of these. The context sets that Kanazawa (1993b) introduced as context-determined quantifier restrictions can be identified with anaphoric links.

d. In fact, even if no distribution takes place, a subset induces subsets for all dependent variables: again, the old men give flowers to their own girl-friends.

e. Furthermore, the order of the anaphors need not be the same. If the subject in the second sentence picks up a referent that is the object of the first sentence, the antecedents that the object stand in some relation with are also available, and become, if you like, more salient.

f. Furthermore, such cases can invert the functional relationship. Given the usual assumption that the subject has scope over the object, this means that the man loved by every woman is selected as a function of the women.

g. A last, different kind of point is that if the next sentence has nothing to do with the previous sentence, the objects mentioned in it should also have no relation with the objects in the previous sentence.

These observations lead to the following list of requirements:

(i) States should assign plural objects to discourse referents and

(ii) States should be able to express relationships between members of one plural object and members of another.

(iii) If we assign a subset of one variable to another, the relationships between members mentioned in (ii) should be preserved.

(iv) States should only express relationships between members if these relationships are explicitly introduced in the discourse.

To these I add a last requirement that has a more formal character:

(v) States should only give values to variables that are actually introduced.

We can make this more concrete by looking at an actual example of a situation in which the above holds. Assume that the men are $bill$, $john$, $harry$ and $charles$, the women are $mary$, $silvia$, ann and $joan$.

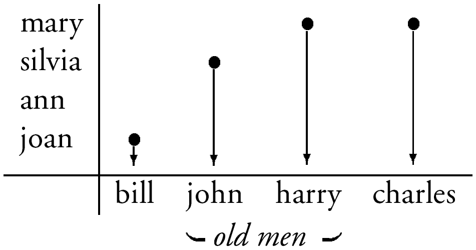
	bill	john	harry	charles
mary			•	•
silvia		•		
ann				
joan	•			

old men

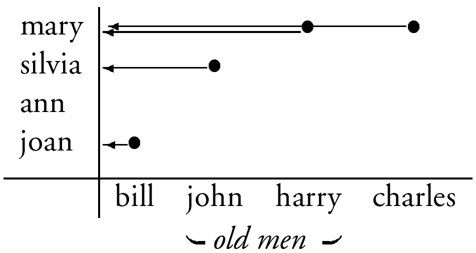
This describes a situation in which each of the men has a relationship with a woman: $bill$ with $mary$, $john$ with ann, $harry$ with $joan$ and $charles$ also with $joan$. Furthermore, $silvia$ does not have a relationship with a man at the moment.

This diagram can be considered a faithful representation of the state. The horizontal (x) axis represents the anaphora denoted with a subscript $_1$ in example (5), the vertical (y) axis corresponds with the anaphora denoted with a $_2$.

In (5a) the interpretation of the pronoun *they*$_1$ is the result of projecting the diagram on the x-axis, This projection results in the set $\{bill, john, harry, charles\}$:

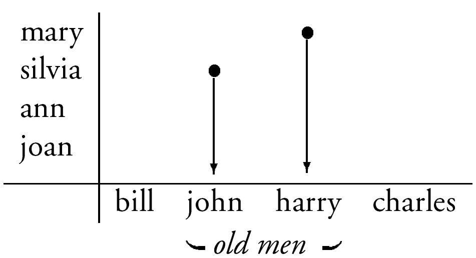


The pronoun *them*$_2$ gets an interpretation by projecting on the y-axis, resulting in the set of women that are loved by these men: $\{mary, ann, joan\}$:

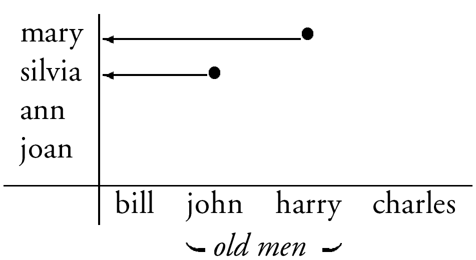


Example (5b) is essentially similar, except that the singular forces the arrows in the previous diagram to be traversed one at a time. First *he*$_1$ gets value $bill$, with corresponding value $mary$ for *her*$_2$, then value $john$ for *he*$_1$ and ann for *her*$_2$, then $harry$ and $joan$, and finally $charles$ and $joan$.

The next two examples (5c–d), show how subsets preserve relations. First the x-axis is restricted to the relevant (old) men $\{john, harry\}$. This creates a sub-diagram:



and then the women that contribute to the sub-diagram are selected ($\{ann, joan\}$):



The examples (5e–f) are essentially equivalent, in as far as the states are concerned, to the examples (5a–b). The difference is that first the projection on the vertical axis is taken, and then the projection on the horizontal axis.

I will now give a formal definition of states, and then in the next section, I give the formal definition of dynamic plural predicate logic.

2.2 A Formal Definition of States

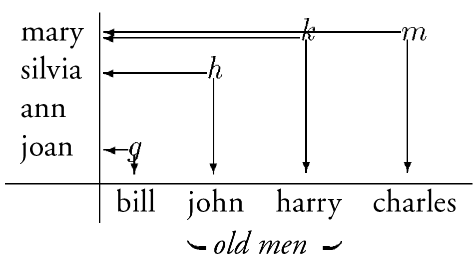
This is as far as informal arguments can bring us. I will now give the formal definition of Dynamic Plural Predicate Logic (DPPL).

I start with a formal definition of states, which will exactly mirror the diagrams used above to explain internal relations between sets of entities (plural objects). Note that this partly repeats material from section 3.2 of chapter 3, but constitutes a more extensive discussion of that material. I restrict attention to what is needed for the interpretation of natural language. Following the definition of the states I define the dynamic logic that is interpreted as transformations on these states in the next section.

It turns out, that the formal objects that can express everything we need are particular sets of partial assignments.

2.1 Definition (States)
A ***finite partial assignment*** *is a partial assignment that assigns the value undefined ($\star$) to all but a finite number of variables. A* ***(finite) state*** *is a set of finite partial assignments such that the set of variables given a value by any of its elements is finite.*

These states satisfy all of the above requirements, as can be seen as follows. Take a diagram as given above. It need not have two dimensions (variables) of course, but the finiteness of the state will ensure that the number of dimensions is finite. Any • is a partial assignment, the values of which are the projections on the different axes. This can be illustrated using the diagram used above:



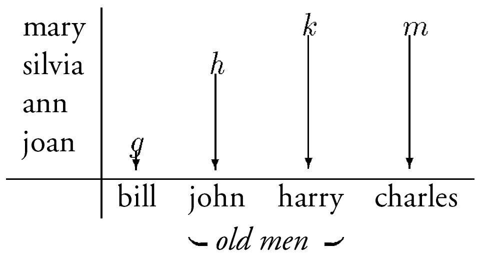
The diagram is represented by the state $G = \{g, h, k, m\}$, with $g(x) = bill, g(y) = mary$, $h(x) = john, h(y) = ann$, $k(x) = harry, k(y) = joan$ and $m(x) = charles, m(y) = joan$.

The discussion above suggests a number of manipulations on these diagrams: in particular, projection on an axis and restriction to a sub-diagram induced by a subset of the values on one of the axes (taking the values of an axis one at the time is of course an extreme case of restriction to sub-diagrams). These operations can be defined on sets of partial assignments as follows.
First projection on a variable:

2.2 Definition (Assigning a Value to a Variable)
The variable x is assigned a value by G corresponding to

$$G(x) = \{g(x) \mid g \in G \;\&\; g(x) \neq \star\}.$$

This corresponds with the diagram



This notation is deliberately the same as the notation of an assignment giving a value to a variable. I make sure to sustain the illusion that states are really just complicated assignments. The condition that only a finite number of variables is defined can now be paraphrased as: $G(x) = \emptyset$ for all but a finite number of variables x.

The operation to define is restriction to sub-diagrams, relative to projection on a subset of an axis.

2.3 Definition (Restrictions of States)
The restriction of a state G relative to a variable x to one of the elements $d \in G(x)$, to subsets $D \subseteq G(x)$, to complements or to the undefined value is given by:

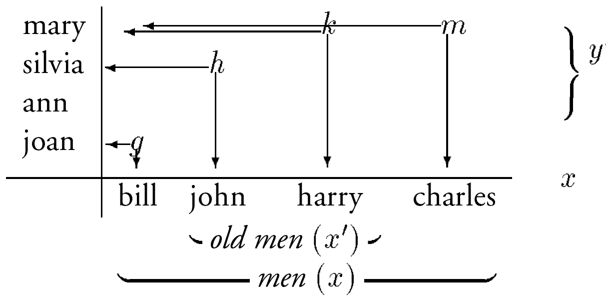
$$\begin{aligned}
G|_{x=d} &= \{g \mid g \in G \;\&\; g(x) = d\} \\
G|_{x\in D} &= \{g \mid g \in G \;\&\; g(x) \in D\} \\
G|_{x\notin D} &= \{g \mid g \in G \;\&\; g(x) \notin D\} \\
G|_{x=\star} &= \{g \mid g \in G \;\&\; g(x) = \star\}
\end{aligned}$$

There are two things to note here. First, the restriction to one element is a special case of restriction to a subset ($G|_{x=d} = G|_{x\in\{d\}}$). Restriction to an element is so common that it deserves its own notation. Second, I defined also a restriction to the undefined element. We have not see reasons to want this, but in the next definition, it will become clear that the undefined value in fact place a pivotal role. It does not only serve to indicate that a variable is not yet bound, but also makes a positive contribution by making it possible that subsets (as the case of the old men) can preserve the internal relations that the sets they are a subset of (the men as a whole) have.

2.3 Properties Holding on States

The above explanation of subsets in terms of diagrams, although correct, misses an important point, and that is that the term corresponding to the subset in real language will introduce its own discourse referent. A four dimensional diagram, displaying the existence of four discourse referents (

(x) the men, (y) the women, (x') the old men and (y') the women that the old men love) can of course not be printed on the page. Fortunately, because the subsets preserve all the relations that the supersets have, we do not need to do this, we can still put this in a two-dimensional diagram.



This form of the diagram raises two questions. The first is, what the values are that g and m assign to x', and the second is how we can define the notion of subset ($x' \subseteq x$) in our more formal language of states (as sets of partial assignments) so that the relations between elements of the values of x and x' with the elements of other values are preserved in exactly the way that this diagram suggests.

The answer to the first question is easy. Given that $g(x)$ and $m(x)$ do not have old men as values, $g(x')$ and $m(x')$ should *not have a value at all*, they should be undefined (have the "value" $\star$).

The answer to the second question is given by a definition which amounts to the following operation on the diagram: (i) Take the elements on the x-axis that we want. (ii) Look at the assignments over these elements and set for these assignments the x' value the same as the x value. (iii) Set the x' value for all other assignments to the undefined value $\star$.

2.4 Definition (Sub-Variable)
The notion of a ***(dependency preserving) subset****,* $x' \subseteq x$ *in a state* G *is defined as follows:*

$$\|x' \subseteq x\|^G \text{ iff } \exists D \subseteq G(x)\ (\forall g \in G|_{x \in D}\ (g(x') = g(x)) \wedge \forall g \in G|_{x \notin D}\ (g(x') = \star))$$

If $x' \subseteq x$ holds in G, $G(x') \subseteq G(x)$. There is some subset D of values on the x axis such that: every partial assignment over a value in that subset ($g(x) \in D$) assigns that same value to x': $g(x') = g(x)$, and every partial assignment that is not over a value in that subset ($g(x) \notin D$) assigns the undefined value to x': $g(x') = \star$. In the next section, I will discuss how such subsets are created, and this will lead to a simpler formulation.

It is important that all partial assignments over a certain value x, give a consistent value to x', either the same as to x or the undefined value. This makes sure that all relations are preserved and not partly lost.

A last concept that will turn out to be very useful is a relation that expresses that H assigns a larger value to x than G in a relevant way.

Aside 4.1 Free sub-domain variable

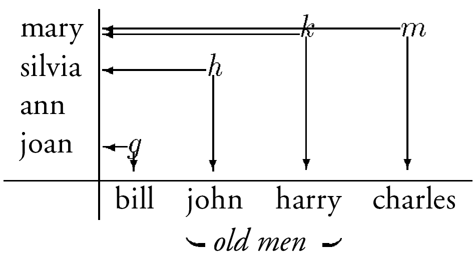
Every now and then, we may have use for a subset relation that does not preserve the dependencies, if only to compare it with the "real" subset that does preserve them. This relation is called the **free subset**. It is defined in the obvious way:

2.5 Definition (Free Subset)
The notion of a free subset, $x' \subseteq^F x$ in a state G is defined as follows:

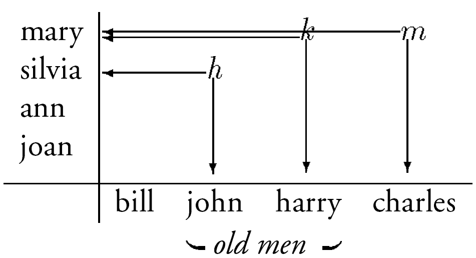
$$\|x' \subseteq^F x\|^G \text{ iff } G(x') \subseteq G(x)$$

This demands that G assigns x' a subset of x, but without any other constraints.

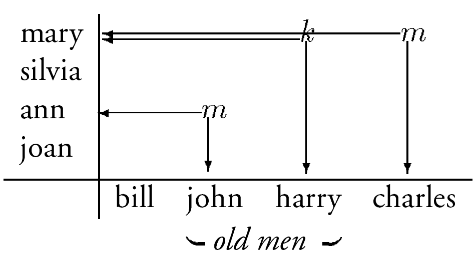
Suppose that the state G is as before:



And now suppose that H were defined by



How do we express that H assigns less men to x than G? It might seem that it is enough to demand that $H(x) \subsetneq G(x)$, but this is not a reasonable definition. Take the following diagram:



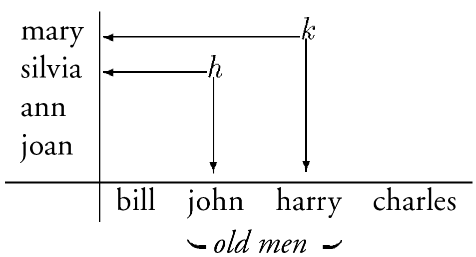
This too assigns less values to x, but it also differs in other respects.

I will define $H <_x G$ to mean that H can be made out of G by removing some of the elements in such a way, that $H(x)$ is a proper subset of $G(x)$. The formal definition looks like this:

2.6 Definition (Ordering relative to Variable)
The relation $H <_x G$, which expresses that the state G is said to assign more values to x than H, is defined by

$$H <_x G \text{ iff } H(x) \subsetneq G(x) \,\&\, H \subseteq G$$

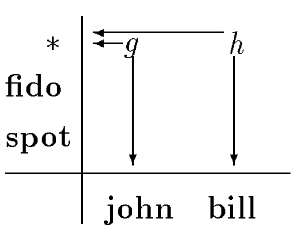
This definition consists of two parts. First of all, $H(x) \subsetneq G(x)$ demands that $G(x)$ is actually a proper superset of $H(x)$: all elements of $H(x)$ should be in $G(x)$, and there should at least be one element more in $G(x)$. The second condition ($H \subseteq G$) demands that by assigning more values to x, no other values have been changed, and only more (different) values are added. Note that G can be larger than H in two ways. First of all, the set assigns more values. But second, there may also be assignments added, that assign a value already in $H(x)$, but which assign other values to other variables: these extra partial assignments add extra dependencies. The relation $H <_x G$ is an ordering exclusively of the projection on the x axis. all other values are preserved from the smaller (H) into the larger (G), but not inspected. For example, the following also describes a sub-diagram H of G:



This last fact will be very useful later when the effects of maximizing one variable on other variables is investigated (in particular, what are usually called the weak and strong readings of quantifiers). It also saves us from something called the proportion problem. All manipulations on states defined below will be relative to projections on an axis, and never involve tuples of variables and their values.

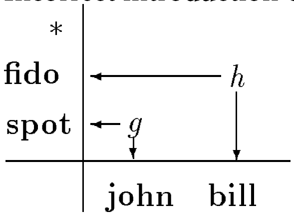
2.4 Operations on States

The above diagrams and corresponding states G are snapshots of the situation taken at one particular moment. The next step is to define the principal dynamics of states: the introduction of a new variable and the assignment of a value to it. The most important notion is that of the independent introduction of a variable and its associated values. To illustrate how a variable is introduced, let us look at a situation where only the variable x is defined, and its values are $john$ and $bill$. Particular, the variable y is not yet defined, it is assigned the value undefined. Suppose G is equivalent to

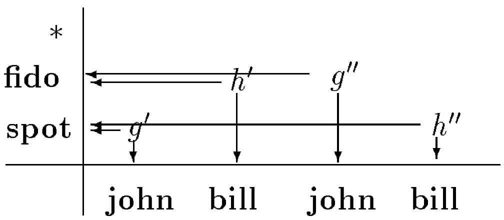


Suppose that we now want to assign y the values *fido* and *spot*. How would a state H look, that is made by taking G, and replacing assignments $g \in G$ by assignments $g' \approx g$ such that $g(y')$ is one of the dogs for every g'? We cannot just assign these values in an arbitrary way, because that might create spurious relationships we do not want. For example, the following would be incorrect:

Incorrect introduction of variables



What we have to do is to assure ourselves that no such relationships are created. We construct such a diagram by taking as many copies of G as there are elements in the value we want to assign to y, and then assign a different value to y. The new state H is given by



In one copy, g and h are replaced by g' and h' assigning *spot* to y, in the other copy, g and h are replaced by g'' and h'' assigning *fido* to y. In that case, there will be no relation between the men and the dogs. Restriction to a subset $x' \subseteq x$ will not introduce a relation with dogs, e.g. $H_{x \in \{bill\}} = \{fido, spot\}$. Formally, this can be expressed as follows (as usual, $g[x := d]$ is the assignment $h = g[x := d]$ such that $g \approx_x h$ and $h(x) = d$):

2.7 Definition (Variable introduction)
Given a state G that does not assign a value to the variable x ($G(x) = \emptyset$), then the state $G[x := D]$ is a state that contains all information that G has, and assigns D as a value to x ($G[x := D](x) = D$) in such a way that x is independent of any variable that is assigned a value by G.

$$G[x := D] = \{g[x := d] | g \in G \;\&\; d \in D\}$$

In case D is singleton, I write

$$G[x := d] = \{g[x := d] | g \in G\}$$

and using this, we can see that $G[x := D] = \bigcup_{d \in D} G[x := d]$.

This leads to the following notion, which defines the parallel of $g \approx_x h$ for dependent plural states:

Aside 4.2 Overview of Dynamic Logic Notation

$G[\![\phi]\!]^d H$	The relation ϕ is defined for G and H.
$G[\![\phi]\!]^+ H$	The relation ϕ holds for G and H.
$G[\![\phi]\!]^d H$	The relation ϕ is defined but does not hold for G and H.
$\|\phi\|_d^G$	$\exists H\ \ G[\![\phi]\!]^d H$, ϕ is defined for G
$\|\phi\|_+^G$	$\exists H\ \ G[\![\phi]\!]^+ H$, ϕ is true for G
$\|\phi\|_-^G$	$\exists H\ \ G[\![\phi]\!]^d H$ but not $\exists H\ \ G[\![\phi]\!]^+ H$, ϕ is false for G

2.8 Definition (x**-descendant of a state**)
A state is said to be an x-descendant of G, written as $H \Leftarrow_x G$, if H is constructed from G by adding an assignment to x to some value:

$$H \Leftarrow_x G \text{ iff } \exists D\ \ H = G[x := D]$$

Note that this is an asymmetric relation, x is independent in H, but in G, the old value $G(x)$ may need not have been.
This concludes the discussion of states. Now everything is in place to define the logic.

3 Dynamic Plural Predicate Logic

Introduction

I will now proceed to define a dynamic predicate logic over the states defined in the previous sections, following the informal investigation into the nature of dynamic formulas given in the first section of this chapter. As in chapter 2, I define when the relation corresponding to a formula ϕ is defined for an input G and an output H, and when the relation is true for this input and output. If it is defined, I write $G[\![\phi]\!]^d H$, if it is true I write $G[\![\phi]\!]^+ H$. I will also use the notation $G[\![\phi]\!]^- H$ to denote that the relation is defined for G and H, but is not true between them.

I will give the definition in two parts. First I give the core logic: the operators that are insensitive to the singular/plural distinction. This is done by combining dynamic and partial logic in a more or less straightforward way. The only thing we have to be careful about is the definition of random assignments, because they make reference to the form of the actual states. Next I define the operators that are specific for our choice of states and implement plural properties.

3.1 Core Logic

Predicates Predicates are tests: they that do not change the state. They are defined if the input is equivalent to the output, and true if the predicate holds in the state:

$$G[\![Px_1 \dots x_n]\!]^d H \text{ iff } G = H,$$
$$G[\![Px_1 \dots x_n]\!]^+ H \text{ iff } G = H \text{ and } \langle G(x), \dots, G(y)\rangle \in \mathcal{I}(P).$$

This is different from the definition in chapter 2, because there is never an undefined value assigned to a variable: at worst, the empty set is assigned (when all *elements* of the state are undefined).

Conjunction The conjunction is interpreted as relational composition. It is defined if the components can be combined in a *defined* way, and true if they can be combined in a *true* way:

$$G[\![\phi \wedge \psi]\!]^d H \text{ iff } \exists K \;\; G[\![\phi]\!]^d K \;\text{ and }\; K[\![\psi]\!]^d H,$$
$$G[\![\phi \wedge \psi]\!]^+ H \text{ iff } \exists K \;\; G[\![\phi]\!]^+ K \;\text{ and }\; K[\![\psi]\!]^+ H.$$

It is possible to use an argument of the same form as lemma 2.6 to establish that K is unique if variables are only bound once.

Negation Negation takes some more effort. It should complement both truth-conditions and the dynamic relation, while preserving the strictness of the logic. The following operator is the operator that does exactly this [2:2.4]. Note that the + and - definitions are similar to those in standard DPL, only the output assignments differ.

(6)
$$G[\![\neg\phi]\!]^d H \;\text{ if }\; G[\![\phi]\!]^d H \;\&\; (G[\![\phi]\!]^+ H \;\text{ or }\; \neg\,\exists K \;\; G[\![\phi]\!]^+ K),$$
$$G[\![\neg\phi]\!]^+ H \;\text{ if }\; G[\![\phi]\!]^d H \;\&\; \neg\,\exists K \;\; G[\![\phi]\!]^+ K,$$
$$G[\![\neg\phi]\!]^- H \;\text{ if }\; G[\![\phi]\!]^+ H.$$

Weak Negation An auxiliary operator that sometimes comes of use is a weaker form of negation that simply complements the relation:

(7)
$$G[\![-\phi]\!]^d H \;\text{ if }\; G[\![\phi]\!]^d H,$$
$$G[\![-\phi]\!]^+ H \;\text{ if }\; G[\![\phi]\!]^- H,$$
$$G[\![-\phi]\!]^- H \;\text{ if }\; G[\![\phi]\!]^+ H.$$

Presupposition The presupposition operator is defined exactly if its argument is true [1:2.5, 2:3.2]:

(8)
$$G[\![+\phi]\!]^d H \;\text{ if }\; G[\![\phi]\!]^+ H,$$
$$G[\![+\phi]\!]^+ H \;\text{ if }\; G[\![\phi]\!]^+ H,$$

Random Assignments Simple random assignment tests whether a variable is new and, if so, introduces it. I use $\Leftarrow_x$ in place of $\approx_x$, for the reasons discussed in section 2.4.

$$G[\![\varepsilon_x]\!]^d H \;\text{ iff }\; H \Leftarrow_x G \;\&\; G(x) = \emptyset$$
$$G[\![\varepsilon_x]\!]^+ H \;\text{ iff }\; G[\![\varepsilon_x]\!]^d H$$

This completes the system that you get by simply combining dynamic predicate logic and partial logic. In chapter 2 additional operators where defined that exist on purely logical grounds but do not seem to be needed for linguistics, at least not for the fragment I am interested in here.

3.2 Additional Random Assignments

The above does not look at the specific choice of states. The states we chose, sets of partial assignments, can express relations between the substructures of the values of variables, but none of the operators is able to introduce such structures. If the initial state is the "everywhere-undefined" state, consisting only of the assignment that is undefined for every variable, then none of the above operators can introduce a dependency. The states will behave as if they are individual assignments that assign plural objects (sets); they will not behave as sets of assignments in an interesting way [3:3.3].

I will now introduce a random assignment that introduces a new variable relative to an old one, and preserves its dependencies (cf. definition 2.4). In particular, the new variable will depend on the old one, breaking the independence of all variables.

Subset Assignments Subset assignment introduces a subset x of an earlier variable y, and preserves all dependencies that y has.

$$G[\![\varepsilon_{x\subseteq y}]\!]^d H \text{ iff } G(x) = \emptyset \;\&\; \exists D \subseteq G(y) \;\; H = \left[\bigcup_{d\in D} G|_{y=d}[x := d] \cup G|_{y\notin D}\right]$$

$$G[\![\varepsilon_{x\subseteq y}]\!]^+ H \text{ iff } G[\![\varepsilon_{x\subseteq y}]\!]^d H$$

This introduces x on a subset D of the values that y takes. On D, x takes the same values as y ($G|_{y=d}[x := d](x) = d$), outside D, x stays undefined ($G|_{y\notin D}(x) = \emptyset$). Note that we need to add $G|_{y\notin D}$, because it contains information on the values of y outside D, and the relations those values have with other variables. Without it, the set that y corresponds to would be damaged, as would be the set assigned to any variable that is dependent on y: x and y would be not only be equal to each other, but also have exactly the same relationships with other variables. This is not what we want.

Relational Assignments An even more general notion is sometimes needed to deal with linguistic examples. Sometimes, a term is introduced by having a relation with an earlier introduced variable. For example, in

(9) *Some children played in the park. The mothers were sitting on the bench.*

(10) *Three cars entered the village. The drivers looked tired.*

the interpretation of *the mother* will introduce discourse referents that are in fact the mothers of the children introduced in the earlier sentence. The case of the drivers is similar. They are related to the earlier mentioned cars.

Establishing what the relations are that exist between such terms goes beyond the scope of the current research; it is definitely part of world knowledge and not part of the internal structure of language. But it is possible to capture the logical content of this relational dependency. This leads to a third notion of random assignment: the introduction of a variable x that is known to have a dependency relation R with y, and hence with all variables that y has a dependency relation with, but not with more variables. This, what I will refer to as relational assignment, introduces variables "as independent as possible given the explicitly mentioned dependencies".

This notion of random assignment is defined relative to a relation $E(a,b)$ between entities (Note: not between sets). Now define $E_R(b) := \{a \mid R(a,b)\}$, the set of entities that b has the R relation with. This relation expresses the known-relation that exists between the antecedent variable y and the newly introduced variable x. Typical examples, related to the above examples (9–10) are $\mathbf{mother-of}$ or $\mathbf{driver-of}$. In a sense, it is this relation that encodes the outside, world-knowledge, side of the dependency.

$$G[\![\varepsilon_{xRy}]\!]^d H \ \text{ iff } G(x) = \emptyset \ \& \ H = \bigcup_{d \in G(y)} G|_{y=d}[x := E_R(d)]$$
$$G[\![\varepsilon_{xRy}]\!]^+ H \ \text{ iff } G[\![\varepsilon_{xRy}]\!]^d H$$

In general, the word *of* as in *the mothers of the children*, *the driver of the car*, *the owner of the apartment*, etc. flags the existence of such a relation, although finding the actual relation $E(d,e)$ or function $E_R(d)$ can only be done using world knowledge; this falls outside the current research topic.

We can now introduce a last notion of random assignment, that introduces a referent x that depends in some underdetermined way on an antecedent referent y:

$$G[\![\varepsilon_{x\mathbf{of}y}]\!]^d H \ \text{ iff } \exists R \ G[\![\varepsilon_{xRy}]\!]^d H$$
$$G[\![\varepsilon_{x\mathbf{of}y}]\!]^+ H \ \text{ iff } G[\![\varepsilon_{x\mathbf{of}y}]\!]^d H$$

In section 5, I will show that the different random assignments defined here are closely related. All can be defined in terms of ε_x and the still to be defined distribution operator δ_x.

3.3 Plural Operators

Singular For distinguishing singulars a distinguished predicate is defined. Although just another predicate, it plays such a prominent role in the definition of quantifiers that it deserve separate mention.

$$G[\![\mathbf{S}(x)]\!]^d H \quad \text{iff} \quad G = H$$
$$G[\![\mathbf{S}(x)]\!]^+ H \quad \text{iff} \quad G = H \ \& \ G(x) \ \text{ a singleton.}$$

$$\mathbf{sing}(x) := +\mathbf{S}(x)$$

The predicate $\mathbf{sing}(x)$ is true (and defined) exactly if x contains one element.

Maximization In discourse, maximal sets have a special position. Quantifier seem always to introduce a maximal value (cf. Evans 1980, and the discussions in chapters 1 and 3 on maximization):

(11) *John has some sheep. Harry shaves them.*

the pronoun *them* refers to all of the sheep. Also if a subset of the discourse referent is introduced, as in

(12) *Some children entered. The boys were carrying a balloon.*

the expressions *the boys* selects all the boys in the set of all children that entered, not just some of them. This shows that the definition of a maximization operator is an important ingredient of quantifier logic.

Before defining the maximization operator that I will actually use, I will first define a simpler one to make it easier to explain the reasons for certain design decisions. The definition is as follows ($<_x$ is defined in definition 2.6):

3.1 Definition (Maximization Operator, first attempt)
The operator $\mathsf{M}^{\bullet}_x$, that maximizes the output of its argument relative to the variable x is defined as follows:

$$
\begin{array}{lll}
G[\![\mathsf{M}^{\bullet}_x(\phi)]\!]^{+}H & \text{iff} & G[\![\phi]\!]^{+}H \;\&\; \neg\,\exists K\ (H <_x K \;\&\; G[\![\phi]\!]^{+}K) \\
G[\![\mathsf{M}^{\bullet}_x(\phi)]\!]^{-}H & \text{iff} & G[\![\phi]\!]^{-}H \;\&\; \neg\,\exists K\ ((H <_x K \;\&\; G[\![\phi]\!]^{-}K)\ \& \\
 & & \qquad\qquad \neg\,\exists L\ (L <_x K \;\&\; G[\![\phi]\!]^{+}K)) \\
G[\![\mathsf{M}^{\bullet}_x(\phi)]\!]^{d}H & \text{iff} & G[\![\mathsf{M}^{\bullet}_x(\phi)]\!]^{+}H \ \text{ or } \ G[\![\mathsf{M}^{\bullet}_x(\phi)]\!]^{-}H
\end{array}
$$

Note that this is one of those operators that can be defined more easily by defining true and false transitions, rather than true and existing transitions. Note the rather complex definition of the (-) case, explained in (iv) below. This operator maximizes the output H for a given variable x, given a fixed input G.

Note the following

(i) Although H is demanded to be the largest *for* x, nothing is said about other variables, be they dependent on x or not. Although it is defined in terms of the states, only the values assigned to x are used to select what output states are acceptable and what are not. As we shall see later, because all variables are introduced with their own maximization operator, this does not have any consequences.

(ii) The output is maximized for one particular input. This means that if x is already fixed on input, the output will assign an identical value to x as the input; in that case the application of the operator is vacuous. Any real application will be in situations where x is introduced inside the scope of $\mathsf{M}^{\bullet}_x$. in fact, the combination $\mathsf{M}^{\bullet}_x(\varepsilon_x \ldots$ can be understood as meaning: *the maximal x such that ....* **This is different from the definition in chapter 3**, but cf. definition 3.2 on page 141.

(iii) You may wonder why the definition uses $\neg\,\exists K\ (H <_x K \;\&\; G[\![\phi]\!]^{+}K)$ rather than the stronger $\forall K\ G[\![\phi]\!]^{+}K \rightarrow K(x) \subseteq H(x)$. To understand this, consider why there might be different output states for a given input state. The only reason for different outputs are random assignments inside ϕ adding variables to the input with different values. In the case of x this is what we expect (the previous point) but other variables might be added too. Suppose, to take the favorite example, x is assigned a set of farmers, and y the donkeys that the elements of x own. Then making (the value of) x larger will automatically add extra values for y too. This is handled correctly by both the definition used and the stronger form. However, the extra variable y introduced need not be dependent on x, and in that case we need to be careful. Suppose that y can take the values $\{a\}$ if x is $\{c\}$ and y is $\{b\}$ if $\{c, d\}$. Then we do not want to lose the value $\{a\}$ just because x is maximized. It stays a matter of further research whether such cases ever really occur.

(iv) The negative clause is maximized too. This is done to give the right behavior of quantifiers under negation, in particular in the case of downward monotonic quantifiers. The definition of the negative part may look somewhat complicated, but it is in fact exactly what one wants [3:4.1:. *p*.108] Take a simple predicate, defined using a set B by $\mathcal{I}(P) = \{A \mid A \subset B\}$, then $\mathsf{M}^{\bullet}_{x}(Px)$ is true exactly if x is assigned the value B, and false exactly if x is assigned the complement $\overline{B}$. If the extra clause, demanding that no subset makes the formula true, is omitted, the largest set making it false would be the universe: $B \cup \overline{B}$, which is not what we want. (also [3:4.1 *p*.108])

(v) The maximization need not be unique, there might be another maximal output H', which assigns other values to x. Provided there is no common, even more maximal state that is a possible output, the two are equally possible. *Maximality and uniqueness are not the same thing*, although they are often confused.

Jumping ahead a bit (I did not yet define how quantifiers are translated, and this is not the maximization operator I am going to use later) a (hugely) simplified interpretation of example (11) is given by

$$(13) \quad \mathsf{M}^{\bullet}_{x}(\varepsilon_x \wedge \mathbf{sheep-of-j}(x)) \wedge \mathbf{shaved-by-b}(x)$$

The first conjunct introduces x with a value that is the largest set of sheep such that john owns them. And the second conjunct then picks this value up.

Let us see how this works in detail, starting with some state G that does not assign a value to x. And assume that the set of the sheep that john owns is A. then $G[\![\varepsilon_x \wedge \mathbf{sheep-of-j}(x)]\!]K$ holds for any $G \Rightarrow_x K$ such that $K(x) \subseteq A$.

The maximization operator selects from these only the maximal set. This means the output of $G[\![\mathsf{M}^{\bullet}_{x}(\varepsilon_x \wedge \mathbf{sheep-of-j}(x))]\!]H$ is that $G \Rightarrow_x H$ such that $H(x) = A$. In this particular case, the output is unique: $H = G[x := A]$. The second conjunct is then interpreted in the state H. The result is that the expression is true if Bill shaves all the sheep that John owns.

Example (12) can be treated in a similarly oversimplified way:

$$(14) \quad \mathsf{M}^{\bullet}_{x}(\varepsilon_x \wedge \mathbf{child}(x) \wedge \mathbf{enter}(x)) \wedge (\mathsf{M}^{\bullet}_{y}(\varepsilon_{y\subseteq x} \wedge \mathbf{boy}(y)) \wedge \mathbf{balloon}(y))$$

The translation of the first sentence (the first conjunct) introduces x as the set of all children (C), in a similar way as in the previous example. The second conjunct gets this state $H = G[x := C]$ as input. $H[\![\varepsilon_{y\subseteq x}]\!]K$ is true for any output K, such that $K(y)$ is a subset of $H(x)$, and preserves all dependencies (not relevant here, of course). Subsequently, $K\llbracket \mathbf{boy}(y) \rrbracket K$ insures that $K(x)$ is a set of boys. The output of $H\llbracket \mathsf{M}^\bullet_y(\varepsilon_{y \subseteq x} \wedge \mathbf{boy}(y))\rrbracket L$ is a state L such that $L(y)$ are all the boys in the set $L(x) = H(x)$. Of this set, it is predicated that (all) the $L(y)$ were carrying balloons. The result is that the expression is true if some children are such that they entered, and all the boys amongst them carried balloons.

It turns out, that it has certain practical and theoretical advantages of having the random assignment outside of the scope of the maximization operator. Although essentially equivalent, it simplifies notation a lot, in particular in the case of non-default information packaging. An added advantage is that it makes the format of the quantifiers more similar to those of static quantifiers in chapter 3.

3.2 Definition (Maximization Operator)
Let $D = G(y)$ and

(15) $H \Leftarrow_{x \subseteq y} G$ iff $H \Leftarrow_{x \subseteq y} G \ \&\ H(x) \subseteq G(y)$

The operator M_x, that maximizes the input and output of its argument relative to the variable x is defined as follows:

$$G\llbracket \mathsf{M}_{x \subseteq y}(\phi)\rrbracket^+ H \quad \text{iff}$$
$$G\llbracket \phi \rrbracket^+ H \ \&\ \neg\, \exists G', H'\ (G' \Leftarrow_{x \subseteq y} G \ \&\ H <_x H' \ \&\ G'\llbracket \phi \rrbracket^+ H')$$
$$G\llbracket \mathsf{M}_x(\phi)\rrbracket^- H \quad \text{iff}$$
$$G\llbracket \phi \rrbracket^- H \ \&\ \neg\, \exists G', H'\ [(G' \Leftarrow_{x \subseteq y} G \ \&\ H <_x H' \ \&\ G'\llbracket \phi \rrbracket^+ H')$$
$$\&\ \neg\, \exists K, L\ (K \Leftarrow_{x \subseteq y} G' \ \&\ L <_x H' \ \&\ K\llbracket \phi \rrbracket^+ L)]$$
$$G\llbracket \mathsf{M}_{x \subseteq y}(\phi)\rrbracket^d H \quad \text{iff} \quad G\llbracket \mathsf{M}_{x \subseteq y}(\phi)\rrbracket^+ H \ \text{ or } \ G\llbracket \mathsf{M}_{x \subseteq y}(\phi)\rrbracket^- H$$

3.3 Lemma[1] (Relation $\mathsf{M}^\bullet_x$ and M_x)
For every x and ϕ:

(16) $\llbracket \mathsf{M}^\bullet_x(\varepsilon_{x \subseteq y} \wedge \phi)\rrbracket^+ = \llbracket \varepsilon_{x \subseteq y} \wedge \mathsf{M}_{x \subseteq y}(\phi)\rrbracket^+$

$\llbracket \mathsf{M}^\bullet_x(\varepsilon_{x \subseteq y} \wedge \phi)\rrbracket^- = \llbracket \varepsilon_{x \subseteq y} \wedge \mathsf{M}_{x \subseteq y}(\phi)\rrbracket^-$

---

1. For the + case we reason as follows: $G\llbracket \mathsf{M}^\bullet_x(\varepsilon_{x \subseteq y} \wedge \phi)\rrbracket^+ H$
$\Longleftrightarrow G\llbracket \varepsilon_{x \subseteq y} \wedge \phi \rrbracket^+ H \ \&\ \neg\, \exists H'\ (H <_x H' \ \&\ G\llbracket \varepsilon_{x \subseteq y} \wedge \phi \rrbracket^+ H')$
$\Longleftrightarrow \exists G'' \Leftarrow^*_x G\ G''\llbracket \phi \rrbracket^+ H \ \&\ \neg\, \exists H'\ (H <_x H' \ \&\ \exists G' \Leftarrow^*_x G\ G'\llbracket \phi \rrbracket^+ H')$
$\Longleftrightarrow \exists G'' \Leftarrow^*_x G\ G''\llbracket \phi \rrbracket^+ H \ \&\ \neg\, \exists G', H'\ (G' \Leftarrow^*_x G \ \&\ H <_x H' \ \&\ G'\llbracket \phi \rrbracket^+ H')$
$\Longleftrightarrow G\llbracket \varepsilon_{x \subseteq y} \wedge \mathsf{M}_{x \subseteq y}(\phi)\rrbracket^+ H$.
$G\llbracket \mathsf{M}^\bullet_x(\varepsilon_{x \subseteq y} \wedge \phi)\rrbracket^- H$
$\Longleftrightarrow G\llbracket \varepsilon_{x \subseteq y} \wedge \phi \rrbracket^+ H \ \&\ \neg\, \exists H'\ (H <_x H' \ \&\ G\llbracket \varepsilon_{x \subseteq y} \wedge \phi \rrbracket^- H') \ \&\ \neg\, \exists L\ (L <_x H' \ \&\ G\llbracket \varepsilon_{x \subseteq y} \wedge \phi \rrbracket^+ H')$
$\Longleftrightarrow \exists G'' \Leftarrow^*_x G\ G''\llbracket \phi \rrbracket^+ H \ \&\ \neg\, \exists H', G'\ (G' \Leftarrow^*_x G \ \&\ H <_x H' \ \&\ G'\llbracket \phi \rrbracket^- H') \ \&\ \neg\, \exists L, K\ (K \Leftarrow^*_x G' \ \&\ L <_x H' \ \&\ K\llbracket \phi \rrbracket^+ H')$ (twice as for + case) $\Longleftrightarrow G\llbracket \varepsilon_{x \subseteq y} \wedge \mathsf{M}_{x \subseteq y}(\phi)\rrbracket^- H$.

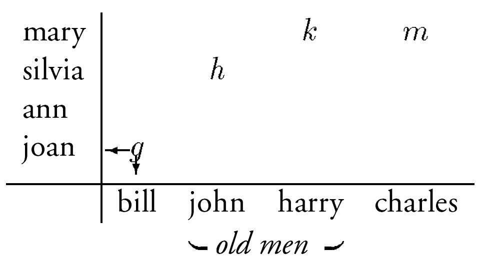
I will always use patterns of the form $\varepsilon_{x\subseteq y} \wedge \mathsf{M}_{x\subseteq y}(\phi)$. This can be read as *the set x in y satisfying ϕ* although it might not always be unique.

The discussion of quantifiers below will give ample opportunity to discuss the details of the workings of the maximization operator.

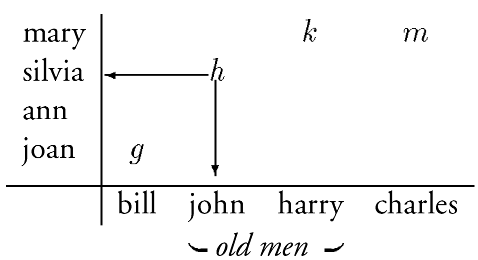
Distribution All the above operators left the states alone, and only influenced the way formulas dealt with input and output states. However, some of the examples (5) above need another mechanism. *every man loves a woman* may introduce a set of men and a related set of women, the predication is over each man at a time. In a similar way, the subject of *He*$_1$ *brings her*$_2$ *flowers to prove this* picks up the discourse referent introduced by *Every*$_1$ *man*, but again predicates over the each man.

In terms of the diagrams, this was explained as taking the sub-diagrams over each men

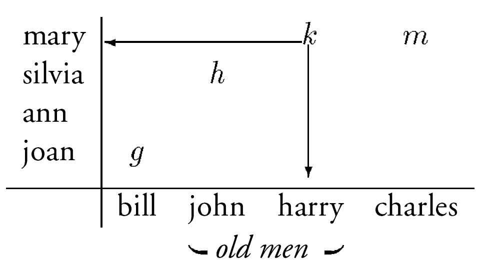
First the predication is applied to Bill,



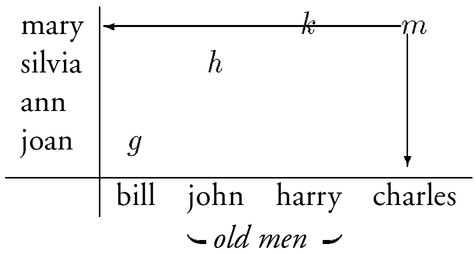
then it is applied to John,



Then Harry



and finally charles



This is only the static picture. In *every man loves a woman*, not only is the state split up in this way to get each man separately, but for each man, the variable y is introduced for the corresponding assignments, and assigned a value that is a woman. The output of the interpretation of this sentence, which will be the combination of all these assignments, will assign the set of these women to the variable y. So if a distribution operator is defined, care should be taken that both input and output are given the right attention.

Incorrect first attempt A first, but incorrect, guess of how to formalize the above story about diagrams in terms of input and output states is something like this. An input state is split into parts corresponding to the different values involved, and then the formula being distributed over is interpreted in these parts one at the time. Afterwards (when leaving the scope of the operator "at the other end") the resulting output pieces are "glued" together again, taking care to assure that in case the outputs are not unique, we take one output per sub-diagram. This leads to the following formulation:

$$G[\![\delta'_x(\phi)]\!]^+ H \quad \text{iff} \quad \forall d \in G(x) \;\; G|_{x=d}[\![\phi]\!]^+ H|_{x=d}$$

Which says that $[\![\delta'_x(\phi)]\!]^+$ holds for a state that assigns some value D to x if $[\![\phi]\!]^+$ holds for the subsets of G that assign individual entities $d \in D$ to x. It is a bit like a loop in a computer program:

Start with $H = \emptyset$.
Do the following for every element d of $G(x)$
Let $G|_{x=d}$ be the subset of G that assigns d to x.
Apply ϕ to $G|_{x=d}$. Let H_d be a possible output.
Add H_d to H $(H = H + G_d)$.
end of loop.
Any H that can be made this way is a possible output of $\delta'_x(\phi)$.

Essentially, δ'_x is a parallel conjunction indexed by the values of x.

$$G \left\{ \begin{array}{lcl} G|_{x=\text{john}} & -\phi \rightarrow & H|_{x=\text{john}} \\ \ldots & \ldots & \ldots \\ G|_{x=\text{charles}} & -\phi \rightarrow & H|_{x=\text{charles}} \end{array} \right\} H$$

This captures the meaning of distribution quite nicely, except for one thing: not all elements need assign a value to x. And, to extend the programming metaphor a bit, the program ϕ will crash on such an input, whatever happens on the other inputs.

Take the case where x is assigned the men as value and $x' \subseteq x$ the old men (as in examples (5c–d). in that case the elements of G $(= \{g, h, k, l\})$ assign the following values to x and x':

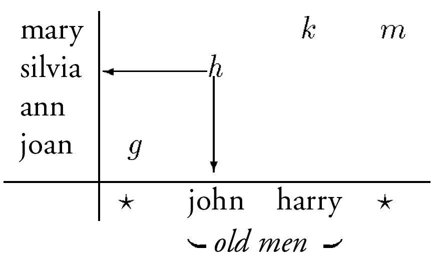
(17)

	men x	*old men* x'
h	bill	$\star$
g	john	john
k	harry	harry
m	charles	$\star$

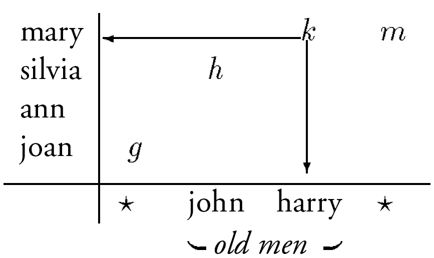
If we distribute over x', we expect only to consider the case of John and the case of Harry.

However, if the above definition is applied the distribution associated with x' will involve the following three cases (now using the diagram with x' rather that x along the horizontal axis)

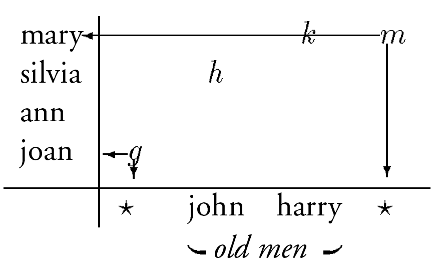
The predication is over John,



then Harry



and finally,the predication is over the undefined element



But this cannot be correct. The undefined value is not one of the old men quantified over. It is a flag to mark non-involvedness of that particular assignment, excluding that particular part of the state from being considered in the quantification. The distribution operator has to deal with the undefined value in a special way. It should ignore it rather than treat it the same as other values.

Of course, the undefined part of the state cannot be ignored. It cannot just be thrown away. In the above state, g takes care of assigning $bill$ to x and $joan$ to y, and h assigns $charles$ to x and $mary$ to y. If that part of the state would be thrown away, the values assigned to x and y would get damaged, a very unwanted side-effect.

A Successful Second Attempt The undefined part of the state has to be preserved for later, but ϕ not applied to it. This is exactly how we do it: we send the undefined part of the input directly to the output:

$$G \left\{ \begin{array}{lcl} G|_{x'=\text{john}} & -\phi \rightarrow & H|_{x'=\text{john}} \\ G|_{x'=\text{harry}} & -\phi \rightarrow & H|_{x'=\text{harry}} \\ G|_{x'=\star} & \longrightarrow & H|_{x'=\star} \end{array} \right\} H$$

This leads to the following definition:

3.4 Definition (Distribution operator)
The distribution $\delta_x(\phi)$ of a formula ϕ over a variable x is defined by:

(18) $$G[\![\delta_x(\phi)]\!]^d H \quad \text{iff} \quad \forall d \in G(x) \;\; G|_{x=d}[\![\phi]\!]^d H|_{x=d} \wedge G|_{x=\star} = H|_{x=\star},$$
$$G[\![\delta_x(\phi)]\!]^+ H \quad \text{iff} \quad \forall d \in G(x) \;\; G|_{x=d}[\![\phi]\!]^+ H|_{x=d} \wedge G|_{x=\star} = H|_{x=\star}.$$

There is a transition corresponding to $\delta_x(\phi)$ from G to H if there are ϕ-transitions from the parts of G that assign one particular value to x to parts of H that assign that same value to x, and if the parts of G and H that assign the undefined value are identical. If all the components of the resulting sum transition are true transition, then that sum-transition is a true transition of $\delta_x(\phi)$.

A New Source of Dependencies An important contribution of the distribution operator is that it is the source for new dependencies between variables. The three definition of random assignments above have in common that they introduce variables that are independent of other variables. Sure, $\varepsilon_{x\subseteq y}$ and ε_{xEy} introduce variables x that have the same dependencies as y, but for $\varepsilon_{x\subseteq y}$ to introduce dependencies, there first have to be some.

In the case of the distribution operator, variables can be introduced in the scope of the operator. In that case dependencies arise, when locally the random assignment introduces an independent variable, and then the different, distributed, output states are re-attached to form the sum-output in which there are dependencies. For example, take *Every man loves a woman* again, assume the following simplified translation (in which the translation of the universal is a bit simplistic, and I do not demand that there is only one woman).

$$\mathsf{M}_x(\varepsilon_x \wedge \mathbf{man}(x)) \wedge \delta_x(\mathsf{M}_y(\varepsilon_y \wedge \mathbf{woman}(y) \wedge \mathbf{love}(y))$$

This will assign the girl-friend of any x to y:

$$G \left\{ \begin{array}{lll} G|_{x'=\text{john}} & -\varepsilon_y \wedge \mathbf{wl}(y) \rightarrow & H|_{x'=\text{john}} \\ \dots & \dots & \dots \\ G|_{x'=\text{charles}} & -\varepsilon_y \wedge \mathbf{wl}(y) \rightarrow & H|_{x'=\text{charles}} \\ G|_{x'=\star} & \xrightarrow{\qquad\qquad\qquad} & H|_{x'=\star} \end{array} \right\} H$$

First x is introduced as set of men. Then for every one of these, a woman is introduced. Although the definition of ε_y demands that the introduction of the values is independent, the input is one of the distributed subsets, so it can only demand independence for each of the man independently. The part $\mathsf{M}_y(\varepsilon_y \wedge \dots$ is only interpreted for sub-states that assign exactly one entity, and consequently independence of other variables can only be demanded in this sub-state.

Variables introduced inside a distribution over a variable x will be (functionally) dependent on that variable x, and all elements introduced for particular values of x will have the same dependencies with variables that (that values of) x has.

Furthermore, because a formula ϕ that $\delta_x(\phi)$ distributes over is never applied to the part of a state that does not assign any value to the distributed variable x, no values for y will be introduced in that part of the state (this is a special case of inheriting the dependency relations).

Note that if later the variable x is distributed over again, this will also give rise to an induced distribution over y. This applies to cases like (5b–c), repeated her

(19) *Every*[1] *man loves a*[2] *woman.*

(b) *He*$_1$ *brings her*$_2$ *flowers to prove this.*

(c) *Every*$_1$ *old man brings her*$_2$ *flowers to prove this.*

In both cases, a distribution over men x_1 takes place (in the case of (c) the distribution is over a subset) and the set of women x_2 is also distributed over because it depends on x_1. Therefore, her_2 picks up the right dependent element.

Pseudo-distribution A related operator that can be defined is the pseudo-distribution operator. The pseudo-distribution $\pi_x\phi$ over a formula is true if ϕ is true for some cover of x. Rather than dividing an input state up in sub-states which assign exactly one value, the state is divided up into parts that assign subsets (the result should still be the whole set, that is why it is demanded that the subsets form a cover).

The definition looks much more complex, mainly because there is more than one possible cover, whereas there is only one way of dividing up a set into its elements (or singletons thereof):

3.5 Definition (Pseudo-distribution operator)
The pseudo-distribution $\pi_x(\phi)$ of a formula ϕ over a variable x is defined by:

$$G[\![\pi_x(\phi)]\!]^d H \text{ iff}$$
$$\quad \forall d \in G(x)\ \exists D \subseteq G(x)\ \left[d \in D \wedge G|_{x=D}[\![\phi]\!]^d H|_{x=D}\right] \wedge G|_{x=\star} = H|_{x=\star},$$
$$G[\![\pi_x(\phi)]\!]^+ H \text{ iff}$$
$$\quad \forall d \in G(x)\ \exists D \subseteq G(x)\ \left[d \in D \wedge G|_{x=D}[\![\phi]\!]^+ H|_{x=D}\right] \wedge G|_{x=\star} = H|_{x=\star}.$$

This definition is not so much different from that of δ_x. This similarity goes further even than it may seem: π can in fact be defined in terms of δ.

The relation between π and δ The definition of δ looks much simpler than π, because δ_x splits up the state in one unique way, and $\delta_x(\phi)$ holds if ϕ holds on *the* (singletons of) the elements, whereas π_x quantifies over all possible covers, and $\pi(\phi)$ holds if ϕ holds on *some* cover. Somewhat surprising however, it turns out that π can in fact be defined in terms of δ_x, at the cost of one extra variable:

3.6 Fact
The following is true for y not bound and ϕ not containing xy:

(20) $\pi_x\phi = \delta_x(\varepsilon_y \wedge \mathbf{sing}(y)) \wedge \delta_y(\phi)$.

We can see this as follows. Suppose $G[\![\delta_x(\varepsilon_y \wedge \mathbf{sing}(y)) \wedge \delta_y(\phi)]\!]^+H$. then $\exists K\ (G[\![\delta_x(\varepsilon_y \wedge \mathbf{sing}(y))]\!]K$ & $K[\![\delta_y(\phi)]\!]^+H)$. This means that for every $d \in G(x)$, $G|_{x=d}[\![\varepsilon_y \wedge \mathbf{sing}(y)]\!]K|_{x=d}$, where $K|_{x=d}(y) = \{e_d\}$ for some entity e_d (subscript to denote the functional dependency on d). this means that $K(y) = \{e_d \mid d \in K(x)\}$. Note that e_d need not be different for different d. This means that $K|_{y=e_d}(x)$ are subsets of of $K(x)$ that need not be singletons. Every set of subsets of $K(x)$ can be encoded in this way. The values of y can be considered names for these subsets. Distributing over y is the same as pseudo-distributing over x.

This will always work. If ϕ is true for some cover of x, it will also be true for some cover of x that has at most[2] as many elements as x.

Of course, there is the question whether this is more than just a neat trick. It does seem to introduce a spurious variable. It would be *much* better if we would have independent confirmation of the existence of such "subset-names". And in fact we have. It has been suggested in the literature that (pseudo-) distributive quantifiers are projected from an event quantifier corresponding to the VP (Schein 1993, Verkuyl 1993, 1994). For example, take the case

---

2. Given a cover **s** of $G(x)$. Say the elements are d_1 to d_nl then make a new cover **t** as follows. Take an element of **s** that contains d_1. Take this element as first element of **t**. Now proceed by taking the d_i with the lowest index that is not yet in some set in **t**, and add to **t** a set of **s** that contains d_i. At worst the elements chosen from **s** will be singleton sets, so at worst the cardinality of **t** is that of $G(x)$. Furthermore, because ϕ holds for all elements of **s**, it will also holds for all elements of **t**$\subseteq$**s**.. Consequently, the domain will always have enough elements to perform the trick (just take the elements of $G(x)$)

sixty-thousand people gathered on town squares.

In this case, there are a number of gathering-events, 3700 here, 8000 there, etc., these are summed up, and of the sum it is then claimed that the cardinality is six-thousand. Assume, for the moment, that the VP has as part of its interpretation a quantifier over events, and assume furthermore, that this quantifier is interpreted distributively, then the translation is something like.

$$\begin{aligned}&\varepsilon_{x'\subseteq y} \wedge \varepsilon_{x\subseteq x'}\\ &\qquad \wedge (+\mathsf{M}_{x'\subseteq y}\delta_{x'}(\mathbf{p}(x'))] \wedge \mathsf{M}_{x\subseteq x'}[\delta_x(\varepsilon_e \wedge \mathbf{sing}(e)) \wedge \delta_e(\mathbf{g}(e,x))] \wedge 60000(x',x))\end{aligned}$$

which does indeed give the right behavior.

This does open up a whole new bag of questions. Traditionally, events are considered to be mass-terms, if you add some together, you have a new (sum) event. And the sum event may or not be of the same type. This makes the notion of distributivity not directly applicable. There is more than one way of proceeding from here. One thing you might try is to develop a notion of "distribution of a particular type". But I would like to suggest a different approach. Why not leave the events abstract? There does not seem a quantificational need to make the events more than plain indices. There seems to be a need for enough different indices, but what really seems to make the interactions of quantifiers "tick" are the interdependencies of variables. An event, in this view, is just some particular cluster of dependencies, named by one of these abstract indices.

Collective The above two operators give special consideration to parts of a state that assign the undefined value to the variable in question. One might wonder, whether a similar consideration should not still hold, even if no (pseudo-) distribution is applied. What brings this question up, are examples like (5e), repeated here

(5) *Every*^{1} *man loves a*^{2} *woman.*
   (e) *The*$_1$ *old men bring them*$_2$ *flowers to prove this.*

This case is similar to the case discussed on page 146, but there it involves the distribution operator, where here the interpretation is collective, normally associated with the absence of an operator. This discussion suggests that there has to be a collection operator, just as there are a distributive and pseudo-distributive operators. It is defined quite simply by:

$$\begin{aligned}&G[\![\kappa_x(\phi)]\!]^d H \ \text{ iff } \ \exists D = G(x) \ \ G|_{x=D}[\![\phi]\!]^d H|_{x=D} \ \&\ G|_{x=\star} = H|_{x=\star}\,,\\ &G[\![\kappa_x(\phi)]\!]^+ H \ \text{ iff } \ \exists D = G(x) \ \ G|_{x=D}[\![\phi]\!]^+ H|_{x=D} \ \&\ G|_{x=\star} = H|_{x=\star}\,,\end{aligned}$$

Note that $\delta_x(\phi)$ is equal to $\kappa_x(\delta'_x(\phi))$, and a similar fact holds for π_x. Linguistically, collectives imply the existence of a non-empty value for the referent. So for the actual collectivity operator, I choose to built this in:

$$\begin{aligned}&G[\![\gamma_x(\phi)]\!]^d H \ \text{ iff } \ G[\![\kappa_x(\phi)]\!]^d H \ \&\ G(x) \neq \emptyset,\\ &G[\![\gamma_x(\phi)]\!]^+ H \ \text{ iff } \ G[\![\kappa_x(\phi)]\!]^d H \ \&\ G(x) \neq \emptyset,\end{aligned}$$

4 Quantification

Introduction

We now have everything in place to define quantification in natural language.

First I discuss the basic format of quantificational expressions. Then I discuss the problems relating to quantifiers like *none*, *at most four* and *few* (MON$\downarrow$). Finally I discuss some variations on this theme, that relate to cumulatives, non-restrictive clauses, and the cases that are often analyzed as scope-inversions, though I will argue that they are not.

4.1 Dynamic Quantifiers

Standard generalized quantifier theory can be paraphrased as follows. Take *the* set corresponding to the restriction and call it A, take *the* set corresponding to the nuclear scope and call it B, then, apply the quantifier to these: $Q(A, B)$. In plural logic we do almost the same: take *a* set corresponding to the restriction and call it x', take *a* set corresponding to the nuclear scope and call it x, finally, apply the quantifier to these: $Q(x', x)$. The only difference between plural logic and the standard theory is that in the logic the sets need no longer be unique. However, this non-uniqueness will only occur in typical plural-logic examples like collective quantification and in the case of singulars. We get *a* set corresponding to the formula by applying the maximization operator, which explains the possibility of non-uniqueness; the maximization does not in general produce a unique maximum.

The general definition of quantifiers in plural logic has the following form

4.1 Definition
The definition of a dynamic quantification $\mathsf{Q}^{\mathsf{y}} x\ (\phi, \psi)$ *relative to a context set expressed by a discourse referent* y *is*

$$\mathsf{Q}^{y} x\ (\phi, \psi) := \varepsilon_{x' \subseteq y} \wedge \varepsilon_{x \subseteq x'} \wedge +\mathsf{M}_{x' \subseteq y}(\phi[x/x']) \wedge \mathsf{M}_{x \subseteq x'}(\psi) \wedge Q(x', x)$$

The generally accepted intuition that the first argument is special and in some way determines what the "universe" is that the quantification is interpreted in (in the words of van Benthem (1986a): it sets the stage for the quantification) is encoded in this expression in two different places, to restrict the quantifier in two different ways. First by the *presupposition operator (+)* and secondly by having the set corresponding to the nuclear scope (x) be introduced as a (dependent) subset of the set x' corresponding to the restriction.

The *presupposition operator (+)* is used to enforce that whatever happens, the restriction is satisfied. Even under a negation, the discourse referents introduced by a quantifier will still satisfy the restriction $\ldots \phi \ldots$ because it can only be true never be false. This illustrated by examples like *It is not true that every woman*^{1} *went to the meeting on thursday. Some of them*$_1$ *went on wednesday.* The quantification is still about women, even under negation

Conservativity was built in by demanding that the set introduced by the nuclear scope is always a subset of the set introduced by the restriction ($x \subseteq x'$). The need for this has to do with yet another problem that plurality poses, which is, that when we claim that a set did something, we do not want this to be true because another larger set did it. I.e., we do not want *Three men gathered in the park* to be true, when in fact it was a group of three men and two women.

A similar example constitutes a further argument for having maximizations in the definition of quantifiers. Take the situation that there is exactly one group of four women walking in the park. In that case we would not say that the collective reading of *three women walk in the park* is true. There might be a set of three women that walks in the park, but that is not maximal.

In the definition, quantification is done over a domain that is fixed by an earlier referent y. In generalized quantifier theory, Westerståhl (1985) introduced the notion of a context set, a set relative to which all quantification is done. A simple definition of such a notion in standard theory is given by $Q^C(A,B) := Q(A \cap C, B \cap C)$ (which, assuming conservativity, equals $Q(A \cap C, B)$). What is left open in the discussion of context sets is the question where context sets come from. In dynamic logic, all linkage is done using variable values (discourse referents), and context sets are simply identified with some particular variable. In what follows, I will in general leave out the context set if no explicit one is mentioned in the context, and write ε_x instead of the more correct $\varepsilon_{x \subseteq y}$. Because in real interpretations, all random assignments will be restricted either by a context set or by a variable introduced in a quantifier restriction, every occurrence of an expression of the form ε_x can be thought of as being of the form $\varepsilon_{x \subseteq z}$, for some not previously mentioned, later to be filled in, context variable z.

If both maxima are unique, definition 4.1 results in the standard case. It will be obvious that then the result is exactly the same as for the standard theory of generalized quantifiers. In fact, the standard theory forms an identifiable subclass of this theory; they are exactly the distributive quantifiers:

(21) $\mathsf{Q}^\delta x \subseteq y\ (\phi, \psi) := \varepsilon_{x' \subseteq y} \wedge \varepsilon_{x \subseteq x'} \wedge +\mathsf{M}_{x' \subseteq y}(\delta_{x'}(\phi[x/x'])) \wedge \mathsf{M}_{x \subseteq x'}(\delta_x(\psi)) \wedge \mathsf{Q}(x', x)$

The distribution operator "forces" any formula to behave as a function over entities rather than sets.

It seems generally accepted, that the restrictions of quantifiers, the CNs, are sets, i.e. always have a unique maximum. Combined with the fact that the whole quantifier restriction is embedded under a presupposition operator, we can safely assume that the restriction will cause no problems. Whatever the position of the quantifier, the variable introduced by the restriction will be bound to *the* set corresponding to the restriction.

4.2 Fact[3] (Folklore about Restriction)
The restriction of a Quantifier corresponds to a unique maximal set.

---

3. It is difficult to see what would constitute a proof of this. Checking all noun-phrases in the world is a bit problematic, and what the form of a more formal proof would be is also not clear. However, Following the example of my old thermo-dynamics syllabus, which claimed that the impossibility of a perpetuum mobile was "proved" by hundreds of years of trying to build one and failing to do so, I would like to point out that linguistics never came up with a case violating this principle. At least as far as I know, and I have been looking.

Note that the output state of the restriction is not unique, only what it assigns to this particular variable is. Random assignments inside the restriction over other variables may still introduce non-uniqueness in the output state.

The nuclear scope, on the other hand, is something entirely different. It might be non-unique. Consequently the existential quantifier will pick out one of the possible local maxima. A plural generalized quantifier is true, when it is true for *one possible value of its nuclear scope*. This will cause problems for MON$\downarrow$ quantifiers. Take the following example

(22) *No women gathered in the square.*

Suppose there are exactly two groups gathering: {Sue, Mary, Fatima} and {Bill, John}. Then the existential in the definition makes this true if we would apply the rule to the quantifier No, because there is one set, that of the men, that makes the resulting formula true. This is of course not what this sentence is supposed to mean. Rather, it seems to mean that there is not one case of women gathering in the square, MON$\downarrow$ quantifiers implicitly quantify over cases.

The solution to this problem is to define MON$\downarrow$ quantifiers as negations of their dual counterparts, MON$\uparrow$ quantifiers. This solution, which effectively translates *no women...* as *it is not the case that for some women...*, incorporates a quantification over cases because dynamic negation quantifies over possible outputs. I will discuss this, and a curious connection that this has with the work on quantifier readings by Kanazawa (1993b), further in the next chapter where the quantifier theory developed in this chapter is applied to language, and most suitable translations of different example sentences and the consequences for the phenomena illustrated by these examples are discussed. What is left for this section is to define forms for the usual ambiguities between readings of quantifiers correspond to specific subclasses of the schema's given above.

4.2 Readings of Quantifiers

After discussing a large number of alternatives, van der Does (1992) argues for three readings of quantifiers. These are the collective reading, called C_2, the distributive reading, called D_1^a, and the pseudo-distributive reading (which he calls the neutral reading), called N_2. The origin of these names lies in his way of distinguishing different formula schema's and need not concern us. I only mention them so you know which formula of van der Does (1992) corresponds to which formula(s) of the logic of this chapter.

These readings can best be understood by looking at a simple example (for details, see chapter 1 and chapter 3).

(23) *Four women lifted a piano.*

The three readings are (C_2) a group, consisting of four women, lifted a piano together (resulting in one piano-lifting event); (D_1^a) everyone of the members of a group of four women lifted a piano (resulting in four piano-lifting events), (N_2) every member of a group of four women was involved in lifting a piano, and all the piano's lifted thus are lifted by these women alone (resulting in between one and fifteen piano-lifting events; this reading does really occur in language, cf. the last paragraph of this section). Note that C_2 is true if there is one such set of women, there might be other sets, whereas D_1^a and N_2 express that this set is every women that could be in such a relation either to this one piano lifted (D_1^a, the piano lifting event did not involve more people) or to piano-lifting in general (N_2, no other women were involved in liftings of some piano). I won't go into this further, it would lead us to far astray from the main discussion, but chapter 3 has a lot to say about this.

In dynamic logic, the readings are the result of plugging in the appropriate operator. Again, the quantification is assumed to be over a domain fixed by a context set constituted by the discourse referent y. There are three plural readings

(24) $$\varepsilon_{x'\subseteq y} \wedge \varepsilon_{x\subseteq x'} \wedge +\mathsf{M}_{x'\subseteq y}(\gamma_{x'}(\phi[x/x'])) \wedge \mathsf{M}_{x\subseteq x'}(\gamma_x(\psi)) \wedge Q(x', x)$$ (Coll.)

(25) $$\varepsilon_{x'\subseteq y} \wedge \varepsilon_{x\subseteq x'} \wedge +\mathsf{M}_{x'\subseteq y}(\delta_{x'}(\phi[x/x'])) \wedge \mathsf{M}_{x\subseteq x'}(\delta_x(\psi)) \wedge Q(x', x)$$ (Distr.)

(26) $$\varepsilon_{x'\subseteq y} \wedge \varepsilon_{x\subseteq x'} \wedge +\mathsf{M}_{x'\subseteq y}(\pi_{x'}(\phi[x/x'])) \wedge \mathsf{M}_{x\subseteq x'}(\pi_x(\psi)) \wedge Q(x', x)$$ (Pseud.)

and in case the quantifier restriction is formed by a singular noun-phrase, we get another two readings

(27) $$\varepsilon_{x'\subseteq y} \wedge \varepsilon_{x\subseteq x'}$$ (Sing.)
$$\wedge +\mathsf{M}_{x'\subseteq y}(\gamma_{x'}(\phi[x/x'])) \wedge \mathsf{M}_{x\subseteq x'}(\gamma_x(\mathbf{sing}(x) \wedge \psi)) \wedge Q(x', x)$$

(28) $$\varepsilon_{x'\subseteq y} \wedge \varepsilon_{x\subseteq x'}$$ (Sing. Distr.)
$$\wedge +\mathsf{M}_{x'\subseteq y}(\delta_{x'}(\phi[x/x'])) \wedge \mathsf{M}_{x\subseteq x'}(\delta_x(\mathbf{sing}(x) \wedge \psi)) \wedge Q(x', x)$$

(29) $$\varepsilon_{x'\subseteq y} \wedge \varepsilon_{x\subseteq x'}$$ (Sing. Distr.)
$$\wedge +\mathsf{M}_{x'\subseteq y}(\pi_{x'}(\phi[x/x'])) \wedge \mathsf{M}_{x\subseteq x'}(\pi_x(\mathbf{sing}(x) \wedge \psi)) \wedge Q(x', x)$$

Note that these six forms lead to four readings. Only one is new: the real singular (Sing.), only mentioned in passing by van der Does because it is not interesting from the perspective of generalized quantifier theory. The other two singular readings are equivalent to each other and to Q^δ, because $\delta_x(\mathbf{sing}(x) \wedge \phi)$, $\pi_x(\mathbf{sing}(x) \wedge \phi)$ and $\delta_x(\phi)$ are equivalent [3:fn. 8].

One other thing to note is that there is a subtle difference between the definition that van der Does gives of collective and the one given above. Although the above definition of collective quantification does allow for other groups to exist next to the one selected, it does demand that the selected set is maximal. Van der Does does not demand maximality, his definition would correspond to

$$\varepsilon_{x'\subseteq y} \wedge \varepsilon_{x\subseteq x'} \wedge (+(\phi[x/x']) \wedge (\psi) \wedge Q(x', x))$$

In particular, in the case there are exactly two piano liftings by women, one by a group of two women and one by a group of four women, both his and my interpretation of the collective reading of example (23) will be true, but in the case that there is one group of two women and one group of five women, his interpretation will be true, but my interpretation will be false.

In the next chapter, I will show how the non-maximal reading can be achieved in this logic by manipulation of the information structure. I will also argue, that this is the better way to deal with the phenomenon.

5 Logical Issues

I will now turn to a small number of formal issues. A large number of logical topics are raised by the preceding chapter. I will restrict myself here to a few essentials. the first of these is the question, what operators can be defined in terms of other ones.

5.1 Definability

Random Assignments

All random assignments discussed can be defined in terms of the simplest, independent version and the distribution operator.

5.1 Proposition (Random Assignments)
The different forms of random assignment can be defined as follows:

(30) $\varepsilon_{x \subseteq y} := \delta_y(\varepsilon_x \wedge \delta_x(x = y))$

(31) $\varepsilon_{xRy} := \delta_y(\varepsilon_x \wedge \mathsf{M}_x(Ryx))$

(32) $\varepsilon_{x\mathbf{of}y} := \delta_y(\varepsilon_x)$

proof The d and $+$ parts are identical, so we only have to prove the d part. I will also ignore the condition that the variable is new, because this shared by all random assignments.
First, consider $G[\![\delta_y(\varepsilon_x \wedge \delta_x(x = y))]\!]^d H$ This is true if for every $d \in G(y)$, $G|_{y=d}[\![\varepsilon_x \wedge \delta_x(x = y)]\!]^d H|_{y=d}$. This introduces x, and then demands that every element of x is equal to d. So either x is d or x is $\star$. Hence either $H|_{y=d} = G|_{y=d}[x := d]$ or $H|_{y=d} = G|_{y=d}$. Let D be the subset of $G(y)$ for which x does get a value. Then we have $\forall d \in D\ H|_{y=d} = G|_{y=d}[x := d]$ and $H|_{y \neq D} = G|_{y \neq D}$. This last conjunction is the definition of $\varepsilon_{x \subseteq y}$.
Now consider $G[\![\delta_y(\varepsilon_x \wedge \mathsf{M}_x(Ryx))]\!]^d H$.
Let $E_R(d)$ be the set of entities such that $G[y = d][x = E_R(d)][\![\mathsf{M}_x(Ryx)]\!]G[y = e][x = E_R(e)]$. Then we reason by a similar argument to the previous case. for every $d \in G(y)$, a value of x is introduced, which has to satisfy $\mathsf{M}_x(Ryx)$. This gives $H|_{y=d} = E_R(d)$, which is the definition of ε_{xRy}.
For the last case, x is introduced in the same way as before, but no constraints are put on it. This essentially means that it is the previous, relational case with an arbitrary relation. Which is how $G[\![\varepsilon_{x\mathbf{of}y}]\!]H$ is defined. [**end proof**]

Varieties

The operators used to define the different varieties can also all be defined in terms of the distribution operator and ε_x, at the cost of one extra variable.

5.2 Definition (Variety Operators)
The operators π_x and κ_x can be defined as follows, where y is fresh:

(33) $\pi_x(\phi) := \delta_x(\varepsilon_y \wedge \mathbf{sing}(y)) \wedge \delta_y(\phi)$

(34) $\kappa_x(\phi) := \delta_x(\varepsilon_y) \wedge \mathbf{sing}(y) \wedge \delta_y(\phi)$

As suggested before (cf. the discussion following fact 3.6), this extra variable may not be auxiliary at all, but in fact correspond with something real: the event introduced by the VP. Note that the extra existence constraint in γ is not easy to build in. Interestingly enough, this format suggests another operator, that does not demand singularity of the extra (event) quantifier:

5.3 Definition (Cover-distribution)
The operator σ_x, intending to demand that its argument is true on an arbitrary cover, is defined ,where y is fresh:

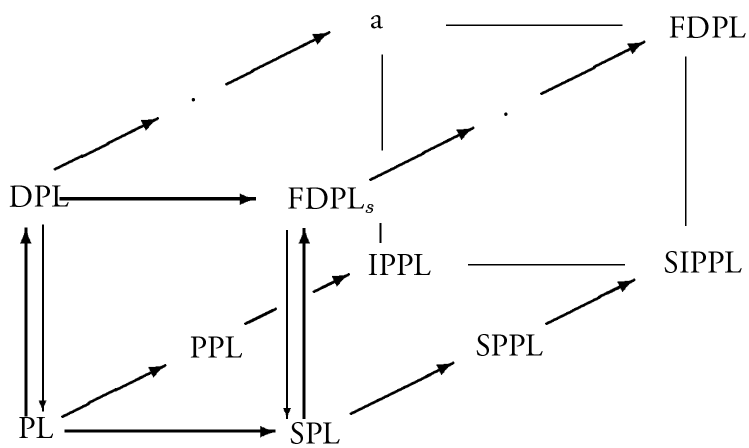
(35) $\sigma_x(\phi) := \delta_x(\varepsilon_y) \wedge \delta_y(\phi)$

This operator allows any element of x to be counted more than once in the distribution.

5.2 The Logical Landscape

This chapter defines a relatively simple dynamic version of generalized quantifier theory. It takes a reformulation of generalized quantifiers in plural logic, that is more or less equivalent to the received perspective on them as formulated in van der Does (1992), and essentially makes this dynamic by reinterpreting the symbols, mirroring the way DPL is defined by reinterpreting the symbols of predicate logic. In fact, the dynamic logic contains the original static one as a weakly-equivalent sub-logic that has the same truth-conditions. As in the singular case, the sub-logic is identified as the logic that does not use the dynamic possibilities, and only binds variables locally.

We can now Put everything discussed in the chapters so far together, in a digram of the logical landscape. As many names as possible are given, but not all are discussed. The fat arrows denote embedding. All static logics are embedded in their dynamic counterpart by translating the formulas as tests, and closing off all quantifiers with a domain closure (!). All total logics can be embedded in partial logics as formulas that happen to never become undefined. The embedding of plural logic in IPPL is proved in chapter 3. Some positions in this diagram are only theoretically filled. Of particular interest is the extension of standard DPL to interpretation under sets of partial assignments. Because (finite) partial assignments are essentially (finite) sequences of values, which are often called cases in the logic tradition (cf. section 5.1 in chapter 3). This suggests a relation with some logic over cases, similar, but not identical, to the ones recently investigated by Dekker (1993) and Elworthy (1995). Maybe something like this lives on node (a).



	name	*where*
PL	Predicate Logic	Logic 101
PPL	Plural Predicate Logic	Chapter 3
DPL	Dynamic Predicate Logic	Chapter 1
SPL	Strictly-Partial Predicate Logic	Chapter 1
$FDPL_s$	Full dynamic Predicate Logic (singular)	Chapter 2
SPPL	Strictly-partial Plural Predicate Logic	not discussed.
FDPL	Full dynamic Predicate Logic	Chapter 4
IPPL	Independent Plural Predicate Logic	Chapter 3
SIPPL	Strictly-partial Independent Plural Predicate Logic	Chapter 3

Conclusions

Finally you may ask: does this chapter give us any new insight in the field of generalized quantifier theory? I'm very tempted to say: I hope not. This chapter tries to make the existing generalized quantifier theory dynamic, and change that theory as little as possible on the way. But there are some insights to be gained from this.

First, **it constitutes an explanation of *context sets* in terms of reasonably standard dynamic semantic terms**. If you are satisfied with assignments assigning sets, the approach can probably be copied reasonably easy to DRT. How the dependencies encoded by the more complex states are to be implemented in DRT seems a bigger question, but even that does not appear to be insurmountable.

Secondly, from a dynamic perspective, it does not seem to be reasonable to consider non-conservative quantifiers. It seems that **conservativity is a dynamic prerequisite**. For one thing, if we consider quantifiers $Q(\phi, \psi)$, anaphors in ψ can be bound by antecedents inside ϕ. If we could just add anything to ψ, these anaphoric links would break, their value be undefined, and the result a mess.

In this chapter the logic on dependency structures shows its true colors. Although in the previous section, it was proved that the static logic over such states is not different from traditional logics, the dynamic case is radically different. This was set up this way, to only allow introduction of dependencies by explicitly present operators that result from well-defined translations from language. There is nothing up the sleeves of this logic: **only linguistically licensed dependencies are allowed**. This is in accordance with the program of this thesis, which studies the internal, linguistically motivated structure of meaning. External, world knowledge contributions to these dependencies may be encoded by precisely described accommodation processes that the discourse grammar can use as repair strategies, but should not be put into the semantics proper.

In the next chapter, the logic is applied to a number of linguistic cases.

5
Applications

Introduction

In this chapter the logic developed in chapters 2, 3, and 4 is used to formalize and explain the phenomena and problems mentioned in chapter 1.

This chapter consists of eight sections. Section 1 starts with outlining the translation principles used to translate natural language into logic. Because I avoided talking about type-logic, I have to be extra careful in being consistent about the formulas produced, as if some type-logic produced them (cf. (van den Berg, 1996a and 1996b) for details on dynamic type logic and resolution principles). The formulas used have a fairly complex fine-structure, because they are worst case formalizations. However, these meanings are intended to be the dynamic meanings of generalized quantifiers. Written as quantifications, the formulas have a standard format.

In section 2, interpretations are given of the standard cases from the literature. It is shown, that the interpretations are similar to the interpretations of DRT/DPL; the truth theory is the same and the dynamic effects are, if both are true, comparable, though sometimes DRT/DPL erases variable bindings that are not erased here. However, the formal mechanisms involved are very different: in this formalism, the fact that these classic cases are invariably about singulars (or are at least distributive) plays a fundamental role.

Section 3 extends the treatment to the more general case, allowing arbitrary quantifiers and plural anaphora. It also discusses the relation between readings and logical properties of the quantifiers, an issue related to work by Kanazawa (1993b) and Lappin and Francez (1994)

Section 4 returns to the notion of information packaging that was briefly mentioned in chapter 3, and shows how this can explain a large number of non-default readings, like cumulatives, adjectival quantifiers and the readings often analyzed as object-wide scope readings (but treated rather differently here).

The next three short sections briefly discuss how the theory can be applied or, by simple modifications, made to apply to a number of other cases.

Section 5 discusses other forms of nominal anaphora, including more indirect, relational anaphora, CN-anaphora, floating quantifiers and quantifiers referring back to more than one antecedent simultaneously. Section 6 discusses non-nominal anaphora, in particular modal subordination and temporal anaphora. Section 7 discusses some aspects of the grammatical theory that is needed to produce the meanings given so far. A sketch is given of how a discourse grammar might produce suitable discourse meanings, and disambiguate between the large number of them.

Finally, I end with a conclusion, which also ends the whole thesis.

1 General Principles

Introduction

In the previous chapter, I gave a redefinition of generalized quantifiers over plurals that made them dynamic. The assumption behind this is that the translations of linguistic expressions in generalized quantifier logic are essentially correct where the local interpretation is concerned, but have to be given extra properties to explain global (dynamic) properties. This is illustrated by sections 2 and 3 below. If the translation normally given to an expression is reinterpreted as an expression of generalized dynamic quantifier theory, the result is an expression that does mirror the intuitive meaning of that expression in a discourse. For the cases that are normally discussed in dynamic semantics, e.g. the donkey case and its relatives and simple distributive quantifiers, this framework gives the same, correct, predictions as to the markedness of the resulting expressions and the binding to selected indices. This theory will often assign more bindings to variables than standard DRT/DPL, due to its reluctance to erase bindings once they are made.

In this section, I will discuss the properties of quantifiers that are important to read the translations.

1.1 Some Properties of Generalized Quantifiers

The previous chapter lists six main translations of quantificational expressions, resulting in four different readings. Every quantification introduces two variables: one identified with the restriction and one with the nuclear scope. Between the values of these variables, the quantification proper (as a relation between sets) is said to hold.

All quantifiers are either singular or plural, and ambiguous between the distributive, pseudo-distributive and collective reading, resulting in three plural readings and two singular readings, the singular and distributive. I will now introduce an abbreviated notation for quantifier, that sufficient to discuss the easy cases. Only if the full complexity of quantifier interpretations is needed, will I fall back on the extended notation. The definitions given are as follows

In the following, ϕ and ψ are variables over x, and $\phi[x/x']$ is the formula you get if you replace all occurrences of x in ϕ by x'.

(1) *the plural collective quantifier* $(\mathsf{Q}^{\gamma} x \subseteq y\ (\phi, \psi))$

$$\varepsilon_{x' \subseteq y} \wedge \varepsilon_{x \subseteq x'} \wedge \mathsf{M}_{x' \subseteq y}\ [\gamma_{x'}(+\phi[x/x'])] \wedge \mathsf{M}_{x \subseteq x'}\ [\gamma_x(\psi)] \wedge Q(x', x)$$

x' is bound to the maximal set satisfying $\phi[x/x']$. x is bound to the set involved in the quantification. In general, this is a locally maximal subset of x' satisfying $\phi \wedge \psi$.

(2) *the singular (collective) quantifier* $(\mathsf{Q}^{\mathsf{s},\gamma} x \subseteq y\ (\phi, \psi))$

$$\varepsilon_{x' \subseteq y} \wedge \varepsilon_{x \subseteq x'} \wedge \mathsf{M}_{x' \subseteq y}\ [\gamma_{x'}(+\phi[x/x'])] \wedge \mathsf{M}_{x \subseteq x'}\ [\gamma_x(\mathbf{sing}(x) \wedge \psi)] \wedge Q(x', x)$$

x' is bound to the maximal set satisfying $\phi[x/x']$. x is bound to a singleton subset, containing an element involved in the quantification. In general, this is an element of x' satisfying $\phi \wedge \psi$.

(3) *the singular or plural distributive quantifier*
$\left(\mathsf{Q}^{\delta} x \subseteq y\ (\phi, \psi), \mathsf{Q}^{\mathsf{s},\delta} x \subseteq y\ (\phi, \psi), \mathsf{Q}^{\mathsf{s},\pi} x \subseteq y\ (\phi, \psi)\right)$

$$\varepsilon_{x' \subseteq y} \wedge \varepsilon_{x \subseteq x'} \wedge \mathsf{M}_{x' \subseteq y}\ [\delta_{x'}(+\phi[x/x'])] \wedge \mathsf{M}_{x \subseteq x'}\ [\delta_x(\psi)] \wedge Q(x', x)$$

x' is bound to the maximal set satisfying $\phi[x/x']$. x is bound to the set of elements involved in the quantification. In general, this is the unique maximal subset of x' such that all elements satisfy $\phi \wedge \psi$.

(4) *the plural pseudo-distributive distributive variant* $(\mathsf{Q}^{\pi} x \subseteq y\ (\phi, \psi))$

$$\varepsilon_{x' \subseteq y} \wedge \varepsilon_{x \subseteq x'} \wedge \mathsf{M}_{x' \subseteq y}\ [\pi_{x'}(+\phi[x/x'])] \wedge \mathsf{M}_{x \subseteq x'}\ [\pi_x(\psi)] \wedge Q(x', x)$$

x' is bound to the maximal set satisfying $\phi[x/x']$. x is bound to the union of set involved in the quantification. In general, this is the unique maximal set such that all elements are in a subset of x' satisfying $\phi \wedge \psi$.

Note Sometimes x is bound to the local complement $(\phi \wedge \neg\psi)$ instead. Typical examples are *few*, *no* and *not every*. This can happen when the quantification is defined as the negation of another quantifier, and also when the quantifier is marked as such in the lexicon.
By convention, corresponding to a quantification over x there will be a primed variable x' bound to the maximal set satisfying $\phi[x/x']$. This set is rarely used outside the quantification (as the discussion in the previous chapters shows, inside it is essential) where it is known as (the link involved in) a CN-anaphor. The use of two variables is only a technical device; intuitively, there is only one referent, with a sort, denoted by x' and constituting the the CN-value used to introduce the referent, and a value, denoted by x. In programming languages, we have something similar, as I mentioned in chapter 1 [1:1.5 *p*.12]. A declaration of the form `int x=3` introduces a referent (the variable `x`) with a value (3) and a type (`int`). In a similar way, we think of referents as having a value and a sort (I have been known to call this the *flavor* of the variable).

Besides quantificational NP's there are two other classes, indefinites and definites. Both are treated here as quantifiers without a real quantificational part. Although they introduce sets x and x', nothing is predicated of these, there is no proper quantifier.

In DPL, both these classes are treated in a special way. Indefinites translate as (singular) existential quantifiers introducing variables with a value, and definites translate as variables that are bound by a quantifier in an earlier expression (locally, the translation is as a free (old) variable).

Despite the huge difference in form between classic DPL and the current logic, for the particular case of singular terms, the resulting interpretations are not *that* different.

1.2 Singular Referents

As an illustration, I will compare the the fully written out forms of the simple singular readings of

(5) (q) *Some man walks.*

(i) *A man walks.*

(d) *the man walks.*

which are interpreted, for y some reasonable context set, as

(6) (q) $\varepsilon_{x'} \wedge \varepsilon_{x \subseteq x'} \wedge \mathsf{M}_{x'}[\mathbf{man}(x')] \wedge \mathsf{M}_{x \subseteq x'}[\mathbf{sing}(x) \wedge \mathbf{walk}(x)] \wedge \exists(x', x)$

(i) $\varepsilon_{x'} \wedge \varepsilon_{x \subseteq x'} \wedge \mathsf{M}_{x'}[\mathbf{man}(x')] \wedge \mathsf{M}_{x \subseteq x'}[\mathbf{sing}(x) \wedge \mathbf{walk}(x)]$

(d) $\varepsilon_{x' \subseteq y} \wedge \varepsilon_{x \subseteq x'} \wedge \mathsf{M}_{x' \subseteq y}[\mathbf{man}(x')] \wedge \mathsf{M}_{x \subseteq x'}[x = x' \wedge \mathbf{sing}(x) \wedge \mathbf{walk}(x)]$

For the moment, I only gave an explicit context set y in the case of the definite, because only in the case of the definites is it essential for its meaning. In fact I will assume that there is always some context set, although often it is underspecified in the discourse grammar. Because we are comparing here with DPL, I will assume that the picked up referent is DPL-like, which means that I assume that it is itself a singular value. Because in general, the variable introduced by a definite is demanded to be identical to the largest subset x of the context set y that satisfies the restriction, the singularity of the referent will assure identity of old and new referent values, $x = y$, which is what we expect of pronouns and definites in DPL. In DPL-like formalisms, definites pick up a salient object. The restriction of the quantifier serves to identify this salient object, rather than being new information.

Note that (6q′) and (6i′) have identical input-output behavior. Although the first is a quantificational expression and the second an indefinite, this does not result in a difference in dynamic effect. This formalism cannot distinguish *some man* from *a man*.

Both the indefinite and the definite can be simplified considerably to such a point that they become very similar in structure to the DPL case. The indefinite is equivalent to

(i′) $\varepsilon_{x'} \wedge \varepsilon_{x} \wedge \mathsf{M}_{x'}[\mathbf{man}(x')] \wedge \mathbf{man}(x) \wedge \mathbf{sing}(x) \wedge \mathbf{walk}(x)$,

which is the conjunction of the standard DPL meaning and something introducing the CN-anaphor:

$$\varepsilon_{x} \wedge \mathbf{sing}(x) \wedge \mathbf{man}(x) \wedge \mathbf{walk}(x) \wedge \varepsilon_{x'} \wedge \mathsf{M}_{x'}[\mathbf{man}(x')]$$

Remember, DPL is always about singulars, something we have to demand here. For the rest, this *is* the same as the meaning DPL in as far as x is concerned.

The definite too is similar to the meaning DPL would give it. Using the identity $x = x'$ to substitute x for x' everywhere:

(d′) $\varepsilon_{x' \subseteq y} \wedge x' = y \wedge \mathsf{M}_{x' \subseteq y}[\mathbf{man}(x)] \wedge \mathbf{sing}(x) \wedge \mathbf{walk}(x))$

Because of the assumption that the value of the picked up referent is a singleton, x will be the same as the value of y. In that case (6d′) is equivalent to

(d″) $\mathbf{man}(y) \wedge \mathbf{sing}(y) \wedge \mathbf{walk}(y) \wedge \varepsilon_{x' \subseteq y} \wedge x' = y \wedge \varepsilon_{x \subseteq y} \wedge x = y$

A combination of the DPL meaning that just uses an old variable without introducing new ones and something that introduces the new variables x, x' as equal to the variable y.

In the following, I will denote a definite introducing x by $\mathsf{def}x$, a pronoun by $\mathsf{pro}x$ and an indefinite by $\mathsf{a^s}x$. Note that $\mathsf{pro}x$ is equivalent to $\mathsf{def}x$ with an empty restriction. A superscript s denotes that it is a singular, plural is default. A superscript δ, π or γ denotes whether the interpretation is to be taken distributive, pseudo-distributive or collective.

The following illustrates the way different expressions might translate. Only collective (singular) cases are given.

a man walks	$\mathsf{a^s}x\ (\mathbf{man}(x), \mathbf{walk}(x))$	(= (5i))
the man walks	$\mathsf{def^s}x \subseteq y\ (\mathbf{man}(x), \mathbf{walk}(x))$	(= (5d))
Some man walks	$\exists^{\mathsf{s},\gamma}x \subseteq y\ (\mathbf{man}(x), \mathbf{walk}(x))$	(= (5q))
every man walks	$\forall^{\mathsf{s},\gamma}x \subseteq y\ (\mathbf{man}(x), \mathbf{walk}(x))$	
he walks	$\mathsf{pro}x \subseteq y\ (\mathbf{walk}(x))$	
	$\varepsilon_{x'\subseteq y} \wedge \varepsilon_{x\subseteq x'} \wedge \mathsf{M}_{x'\subseteq y}\,[x' = x'] \wedge \mathsf{M}_{x\subseteq x'}\,[x = x' \wedge \mathbf{sing}(x) \wedge \mathbf{walk}(x)]$	

2 Standard Cases

Introduction

The previous chapters already give (partial) interpretations to a large number of cases. In this section and the next two, I will concentrate on three special subjects. First of all, I discuss the cases that DPL also discusses. I show that the current theory still give the same, correct interpretations for these cases. Section 3 illustrates the more general dynamics possible, concentrating on a discussion of the two benchmark lists, the one mentioned on page 4 ([1:1.1]) to illustrate the different possible anaphora and the list on page 126 ([4:2.1]) that was used to argue for the particular form that states take. Finally, section 4 discusses the relation between monotonicity properties and readings that were described by Kanazawa (1993b). In this section, I will discuss simple cases of anaphoric reference, as discussed in sections 1.2 of chapter 1.

2.1 Simple Cases

A man is walking in the park. He whistles.

This example involves singular NP's. Consequently, there a two readings, a simple singular and a distributive. The singular reading is the more natural one. It is the only reading discussed in Groenendijk and Stokhof (1991)

(7) $\exists^{\mathsf{s},\gamma}x\ (\mathbf{man}(x), \mathbf{witp}(x)) \wedge \mathsf{pro^s}y \subseteq x\ (\mathbf{wh}(y))$

The first quantification introduces a, possibly non-unique, maximal set of one element. The second quantification introduces a variable y, bound to the largest subset of x (i.e. x itself), and claims that this is a set of entities that whistle.

Aside 5.1 Uniqueness of the Introduced Referent

A point raised by the singular case is what it means that the introduced value is non-unique. The question this relates to is what the semantic representation of an expression encodes. Given that the hearer cannot look in the speaker's mind, he has to do the best he can to understand what she is saying. So although she may quite likely have one particular man in the park in mind, the hearer, who we can assume will not be capable of reading minds, has to represent all possible men. In fact now, when the second sentence is heard, this constitutes extra information: it can be used to eliminate some possibilities that the speaker did not intend. Of course, the simplest way for the hearer to find out which, unique, man the speaker means is to ask her. But there may be an independent reason not to do that; what is a reasonable topic of discussion and what not, is not the subject of semantics.

Is an introduced referent unique? Yes it is, but the hearer need not have enough information to decide which of the possible values is the intended one. And there might be all kinds of reasons why determining the exact value might not be the first priority in a conversation. Discourse semantics gives the machinery to postpone determination of the unique set to a later date, or indefinitely, without blocking calculation of other values. This is very useful, because often more than one question needs resolving at the same time, and it would not do that the formal language would dead-lock in such circumstances. The formalism must be able to give a reasonably meaning to an underspecified situation.

This is particularly important in cases where the conflict cannot be resolved. For example, if the different possible non-unique cases cannot be distinguished, as in the famous sage-plant example, briefly discussed in section 2.3. In such cases, the speaker may have one of the possibilities in mind, but it would be impossible to formulate which one.

In other words, the first quantifier introduces a variable bound to one of the men that walk in the park. The second picks this value up, an claims of it, that it is someone who whistles.

The other, distributive, reading is as follows:

(8) $$\exists^{\mathsf{s},\delta} x\ (\mathbf{man}(x), \mathbf{witp}(x)) \wedge \mathsf{pro}^{\mathsf{s},\delta} y \subseteq x\ (\mathbf{wh}(y))$$

In this case, the first quantification introduces the maximal set x such that any element is a man that walks in the park. the second quantification introduces the largest subset y of x such that every element of that set is an element of x, i.e. $y = x$, and claims of this set y, that every element of it whistles.

This reading is less often discussed as a reading of this sentence, but it does exist. It is the rule-description reading that is often exemplified by the example (imagine a sergeant who shouts at a group of soldiers):

> *A soldier does not protest. He does what he has trained for. He walks towards the enemy and lets himself get shot. That's what he's trained for.*

Also cf. what van der Does (1995) has to say about this.

The Non-existence of Mixed Variety Readings

In the previous section, I mentioned that I will be assuming throughout this chapter that the variety of an anaphoric quantifier (whether operator is γ, δ or π) is disambiguated by the discourse grammar.

In the case of an antecedent noun that is singular, the anaphoric expression has to have the same variety as the antecedent. Either both singular (γ and $\mathbf{sing}$) or both distributive (δ and $\mathbf{sing}$).

In discourse grammar, it is said that both expression have *parallel structure*, a central method in the resolution mechanism proposed in Prüst et al. (1994). However, discourse grammar is not discussed in this thesis, and you have no reason to take me on my word that I'm giving you the right readings. In this case, it is easy enough to see, that the other two possible readings, those that arise from the variety not being equal, are not worth looking at:

(a) $\exists^{\mathsf{s},\gamma} x\ (\mathbf{man}(x), \mathbf{witp}(x)) \wedge \mathsf{pro}^{\mathsf{s},\delta} y \subseteq x\ (\mathbf{wh}(y))$

(b) $\exists^{\mathsf{s},\delta} x\ (\mathbf{man}(x), \mathbf{witp}(x)) \wedge \mathsf{pro}^{\mathsf{s}} y \subseteq x\ (\mathbf{wh}(y))$

In the case (a), the antecedent introduces a singular entity (a singleton set) as value of x, and the anaphor picks up the largest subset of this such that every element of this whistles. But this too is at best a singleton set. This reading is equivalent to the singular parallel-structure reading In the case of (b), the first quantifier introduces the set of all men that walk in the park, but the second picks it up, and demands that it is a singleton set. If there is exactly one man walking in the park, this will again be equivalent to the singular reading. But if there is more than one man, this will be undefined, because the restriction of the quantifier can never be false, only true or undefined.

2.2 A Man and his Donkey

The next thing to do is to look at the classic example, the donkey sentence. Both strong and weak readings are given a meaning that is compatible with the DRT/DPL.

Every farmer who owns a donkey beats it.

(9) $\forall^{\mathsf{s},\delta} x\ (\mathbf{f}(x) \wedge \exists^{\mathsf{s},\delta} y\ (\mathbf{donkey}(y), \mathbf{own}(x,y)), \mathsf{pro}^{\mathsf{s},\delta} y_2 \subseteq y\ (\mathbf{beat}(x,y_2)))$

The outside quantifier is distributive, so $\mathbf{f}(x) \wedge \exists^{\mathsf{s},\delta} y\ (\mathbf{donkey}(y), \mathbf{own}(x,y))$ is interpreted separately for every farmer. For this farmer, y is assigned all the donkeys that farmer owns as value. The final result of the first argument is that x is assigned the farmers and y all the donkeys they own, where the donkeys are functionally dependent on the farmers. In the second argument, the nuclear scope, there is again a distribution over the farmers, so for every farmer, $\mathsf{pro}^{\mathsf{s},\delta} y_2 \subseteq y\ (\mathbf{beat}(x,y_2))$ is interpreted for x that farmer and y the donkeys he owns. The donkeys are then picked up by the pronoun, and it is predicated of this farmer that he beats these donkeys. The output of the quantification is a state in which x is assigned the farmers, and in which y and y_2 are assigned the donkeys the farmers own.

This interpretation has the same truth-values as those of the interpretation given by DRT and DPL to donkey sentences. However, the three theories achieve this result in rather different ways. In DRT, the universal quantifier binds all free variables in its scope, effectively quantifying over farmer-donkey pairs (cf. Kamp (1981)). In DPL, the effect is achieved by the hidden universal quantification over output states that the negation contains ([1:1.2]). Here, the effect is achieved by the fact that the antecedent has the distributive reading, resulting in the unique maximal set of donkeys, combined with the fact that *it* is a pronoun not in the scope of the quantifier that binds it, and which is therefore an E-type pronoun (Heim (1990) also gives an analysis in terms of E-types).

Note that distribution makes it possible to give a pointwise analysis in the sense that we can look at one farmer at a time. In this perspective, the donkey sentence is nothing but the distributive-indefinite case formalized in (8).

Every driver who has a dime puts it in the meter.

In the light of this last remark, it will not come as a surprise that there is also a counterpart of the singular-indefinite reading (7) discussed above.

(10) $\forall^{\mathsf{s},\delta} x\ (\mathbf{d}(x) \wedge \exists^{\mathsf{s},\gamma} y\ (\mathbf{dime}(y), \mathbf{has}(x,y)), \mathsf{pro}^{\mathsf{s}} y_2 \subseteq y\ (\mathbf{itm}(x,y_2)))$

The analysis is very similar to that of the strong donkey sentence, except that here, per driver only one, possibly non-unique, dime is introduced as a value for y. The result is that x is all the drivers, and that y and y_2 are identical, but possibly non-unique sets of dimes, such that there is exactly one dime for every driver.

Both the *donkey* and the *dime* sentence are ambiguous between both readings. How these readings are disambiguated is not explained by these formulas. It may be that the distributive reading is triggered by a more generic context, or licensed by the aspect or maybe it is just world-knowledge. Logic alone can not solve this problem.

2.3 A Sentence that Caused DPL some Problems

Everyone who bought a sage-plant here, bought five others with it.

Given that it is explicitly allowed to have non-unique output states, the sentence that causes problems with this is not a problem here. The meaning of *other* is discussed in section 5.2, it introduces new values using two anaphoric links: to the discourse referent it is different from, and to the CN-value it is to be chosen from. The singular case of the indefinite translates as follows.

$$\forall^{\mathsf{s},\delta} x\ (\mathbf{c}(x) \wedge \exists^{\mathsf{s},\gamma} y\ (\mathbf{sp}(y), \mathbf{b}(x,y)), \varepsilon_{y_2} \wedge y_2 = y \wedge \left[5^{\mathsf{s},\delta} y_3 \subseteq y'\ (y_2 \subsetneq y_3, \mathbf{b}(x,y_3))\right])$$

This has a similar form to the dime case. The first argument of the quantification introduces, for every client, y with as a value one of the, non-unique, sage-plants. Furthermore, the set of all sage-plants is assigned to the CN-referent y'. In the second argument, y is picked up and identified with y_2. Because the nw value should not be dependent on the old one (that would be impossible, since they are disjoint memers of the same CN-referent), the new variable is introduced as independent from the old one, but with the same value. Then y_3 is introduced as the set of all sage-plants bought by a particular customer that are not equal to the plant (denoted by y_2) that was already introduced.

Of course, this does not solve the real problem with this sentence, which is a problem for theories that demand uniqueness of the referent-value. The only thing this solves is that the original plant is not counted again among the five other ones, and that we do not get a recursion, buying for any of the five, yet five more, *ad infinitum*. For the uniqueness question, cf. aside 5.1.

If all singular quantifiers are ambiguous between singular and distributive readings, there should also be a distributive version of this. It turns out that this gives a slightly better interpretation. Assuming the principle that was also assumed in the donke/dime cases, that antecedent and anaphoric clause are parallel in their variety, the distributive reading:

$$\forall^{\mathsf{s},\delta} x\ (\mathbf{c}(x) \wedge \exists^{\mathsf{s},\delta} y\ (\mathbf{sp}(y), \mathbf{b}(x,y)), \varepsilon_{y_2} \wedge y_2 = y \wedge \delta_{y_2} \left[5^{\mathsf{s},\delta} y_3 \subseteq y'\ (y_2 \subsetneq y_3, \mathbf{b}(x, y_3)) \right])$$

This has some surprising properties. The distributive reading of *a sage-plant* introduces a referent with all six sage-plants as value. The anaphor then picks this up, but inspects them one at a time, and for each, says that there are five others. It turns out, that the output state of this is in fact unique.

That the distributive reading has such fortunate properties is rather special, even simple variants, like

(11) *Everyone who bought two sage-plants here, bought five others with them.*

pose problems. The distributive reading of this will demand that the maximum number of bought sage-plants amounts to two, which is either be false (if there are seven, as we expect) or undefined, if there *are* two, because then *other* will have nothing to pick up, resulting in undefined. The only reading that make sense here intuitively is the adjectival reading (cf. section 4.1 below).

2.4 A Sentence that Caused DPL many Problems

Either there is no bathroom in the house, or it is in a funny place.

This sentence is really only a problem for theories that do not have double negation reduction. Theories like the dynamic Montague grammar (DMG) based theory of Dekker (1993) or the more recent double negation DRT of Krahmer (1995) also gets this right.

(12) $\neg \exists^{\mathsf{s},\delta} x\ (\mathbf{br}(x), \mathbf{ih}(x)) \vee \mathsf{pro}^{\mathsf{s},\delta} x_2 \subseteq x\ (\mathbf{fp}(x))$

The first thing to do, is to write out the negation $\phi \vee \psi$ as $\neg(\neg\phi \wedge \neg\psi)$:

(13) $\neg(\neg\neg \exists^{\mathsf{s},\delta} x\ (\mathbf{br}(x), \mathbf{ih}(x)) \wedge \neg\, \mathsf{pro}^{\mathsf{s},\delta} x_2 \subseteq x\ (\mathbf{fp}(x)))$

which is true if

(14) $\neg(\exists^{\mathsf{s},\delta} x\ (\mathbf{br}(x), \mathbf{ih}(x)) \wedge \neg\, \mathsf{pro}^{\mathsf{s},\delta} x_2 \subseteq x\ (\mathbf{fp}(x)))$

is true, which is the case if there is no output that assigns to x a bathroom in the house that is not in a funny place. If true, the output of this quantification is as follows. If there are bathrooms, they will all be in funny places. The set of these is assigned to x. If there are no bathrooms, x will be empty. Note that in this latter case, there *is* an output state, but it does not assign a value to x.

Note that when the non-distributive (i.e. singular) reading is chosen, there *has* to be a bathroom, because the sets involved in collective quantification are demanded to be non-empty. This is not surprising, since I built it in just on the evidence of cases like this (cf. page 148), and it would be more satisfying if this were derived from more primitive facts.

3 General Cases

3.1 The Benchmark Lists of Chapters 1 and 4

I will now turn to the two lists of major examples of possible phenomena mentioned above to guide our search for the holy grail of dynamic semantics.

In chapter 1, a list was given to illustrate a number of different ways that we can refer back to an antecedent. Here the list is again, now with the interpretations added. They are discussed further below:

(15) a. *A man entered the room. He wore a blue sweater.*
$\mathsf{a}^{\mathsf{s}} x\ (\mathbf{man}(x), \mathbf{enter}(x)) \wedge \mathsf{pro}^{\mathsf{s}} y \subseteq x\ (\mathbf{wbs}(y))$

b. *A man entered the room. The man wore a blue sweater.*
$\mathsf{a}^{\mathsf{s}} x\ (\mathbf{man}(x), \mathbf{enter}(x)) \wedge \mathsf{def}^{\mathsf{s}} y \subseteq x\ (\mathbf{man}(y), \mathbf{wbs}(y))$

c. *A painter entered the room. The artist wore a blue sweater.*
$\mathsf{a}^{\mathsf{s}} x\ (\mathbf{painter}(x), \mathbf{enter}(x)) \wedge \mathsf{def}^{\mathsf{s}} y \subseteq x\ (\mathbf{artist}(y), \mathbf{wbs}(y))$

d. *An artist entered the room. The painter wore a blue sweater.*
$\mathsf{a}^{\mathsf{s}} x\ (\mathbf{artist}(x), \mathbf{enter}(x)) \wedge \mathsf{def}^{\mathsf{s}} y \subseteq x\ (\mathbf{painter}(y), \mathbf{wbs}(y))$

e. *A man and a woman entered the room. The man wore a blue sweater.*
discussed in section 5.2 (where $x_1 + x_2$ is discussed).

f. *A married couple entered the room. The man wore a blue sweater.*
$\mathsf{a} x\ (\mathbf{couple}(x), \mathbf{enter}(x)) \wedge \mathsf{def}^{\mathsf{s}} y \subseteq x\ (\mathbf{man}(y), \mathbf{wbs}(y))$

g. *A married couple entered the room. HE wore a blue sweater.*
$\mathsf{a} x\ (\mathbf{couple}(x), \mathbf{enter}(x)) \wedge \mathsf{pro}^{\mathsf{s}} y \subseteq x\ (\mathbf{wbs}(y))$

h. *Some*^{9} *children entered the room. Some*$^{10}_{9}$ *boys wore red hats, another*$^{11}_{9\neq 10}$ *boy wore a blue cap and two*$^{12}_{9}$ *girls, who were looking at the window, held their*$^{13}_{12}$ *hats in their*$^{14}_{12}$ *hands.*
$\exists^{\gamma} x_9\ (\mathbf{child}(x_9), \mathbf{enter}(x_9)) \wedge \exists^{\gamma} x_{10} \subseteq x_9\ (\mathbf{boy}(x_{10}), \mathbf{wbs}(x_{10})) \wedge \ldots$ The rest is similar, *another* is discussed in section 5.2.

Let us go through this list one at a time.

A man entered the room. He wore a blue sweater.

This case was already discussed in section 2.1, except that there shrubbery was involved and the man whistled.

A man entered the room. The man wore a blue sweater.

This is similar. The first conjunct introduces a man that entered as the value for x, and the definite picks that man up, assigns the man to y and claims of it that he wore a blue sweater.

A painter entered the room. The artist wore a blue sweater.

This again works the same way, but a new aspect is introduced. Because the definite is more general than the antecedent, it might potentially denote more objects. However, because definites do not have an independent choice from the model (the world) but pick up an earlier referent, nothing new happens. One way of looking at this is as an intermediate between copying the antecedent (the previous case) and using a pronoun (the most general of definites). This probably mainly used as a stylistic device to avoid repeating the same word to often, although varying the generality of the definite can also serve to give concrete form to the discourse structure.

An artist entered the room. The painter wore a blue sweater.

This is a rather different case. Here the definite is less general than its antecedent, opening up the possibility that the antecedent does not satisfy the noun of the definite. For this reason, this micro-discourse does feel slightly uncomfortable. However, there is a special register that uses this as the norm, called *newspaper language*. A typical example is given by

> *A woman beat up a policeman yesterday. The 33 year old mother of two held the 42 year old inspector from Amsterdam responsible for her 27 speeding tickets.*

It is used to convey extra information without the need of more convoluted language:

> *A woman beat up a policeman yesterday. The woman, who was a 33 year old mother of two held the policeman who was a 42 year old inspector from Amsterdam responsible for her 27 speeding tickets.*

A useful device in a context were conciseness and simplicity are merits. Note that in this formalism, no complicated accommodation process is needed, only the meta-constraint that a reasonable discourse will have to have a defined truth value.

A married couple entered the room. The man wore a blue sweater.

This case cannot be treated by the formalism discussed so far, but we can treat it if we follow the example of a number of other authors and translate the expression *couple* as a "hidden" plural, introducing a plural referent. In section 5.3 I will discuss why this is wrong, and suggest an alternative. If we do allow the cheat, the first introduces a referent that is assigned two people as value, one of which is a man. The definite picks up the largest subset of that referent which are all men, and claims of this subset that it is one man, who wears a blue sweater. Normal plurals work in similar ways:

> *Some children entered the room. The boys wore red hats.*
>
> $\exists^{\gamma} x\ (\mathbf{child}(x), \mathbf{enter}(x)) \wedge \mathsf{def}^{\mathsf{s}} y \subseteq x\ (\mathbf{boy}(y), \mathbf{wbs}(y))$

The first conjunct introduces the children that entered the room. The second conjunct selects the largest subset of them that are boys, and claims of these that they wore red hats. Which is what it should mean.

A married couple entered the room. HE wore a blue sweater.

If this is acceptable at all, it has the same meaning as the previous example. The problem with it is, that the pronoun has nothing to agree with and furthermore, that gender is not predicated of the argument:

A woman was entering from the right. He wore a red hat.

Although obviously incorrect, the incorrectness does not seem to lie in the resolution mechanism: the pronoun *is* resolved to *a woman*. For this reason, it is not clear whether resolving *he* in the original example to the man of the couple is something that happens direct, or involves some repair strategy.

We now turn to the second list of benchmarks for our theory.

In chapter 4, I got a lot of extra mileage out of the old favorite *Every man loves a woman*. The specific choice of states was argued for using a number of possible sentences following it. Maybe *every* is not really an "other" quantifier, but it is pretty useful to see how distributive quantifiers introduce values with dependencies. Until now, not many of the special aspects of states have been discussed. Almost everything discussed so far can easily be dealt with using simpler states that are simple assignments assigning sets, although it would take some effort to get the summing up aspect of distribution right in that case. As for the previous list, it is repeated here with meanings added (Note that I only give the meaning for the singular indefinite object, for the distributive reading, just add δ where appropriate).

(16) *Every*^{1} *man loves a*^{2} *woman.*

$\forall^{\mathsf{s},\delta} x_1\,(\mathbf{man}(x_1), \mathsf{a}^{\mathsf{s}} y_1\,(\mathbf{woman}(y_1), \mathbf{love}(x_1, y_1)))$

a. *They*$_1$ *bring them*$_2$ *flowers to prove this.*

$\mathsf{pro} x_2 \subseteq x_1\,(\mathsf{pro} y_2 \subseteq y_1\,(\mathbf{bfp}(x_2, y_2)))$

b. *He*$_1$ *brings her*$_2$ *flowers to prove this.*

$\mathsf{pro}^{\mathsf{s},\delta} x_2 \subseteq x_1\,(\mathsf{pro}^{\mathsf{s}} y_2 \subseteq y_1\,(\mathbf{bfp}(x_2, y_2)))$

c. *Every*$_1$ *old man brings her*$_2$ *a present to prove this.*

$\forall^{\mathsf{s},\delta} x_2 \subseteq x_1\,(\mathbf{man}(x_2) \wedge \mathbf{old}(x_2), \mathsf{pro}^{\mathsf{s}} y_2 \subseteq y_1\,(\mathbf{bpr}(x_2, y_2)))$

d. *The*$_1$ *old men bring them*$_2$ *presents to prove this.*

$\mathsf{def}\, x_2 \subseteq x_1\,(\mathbf{man}(x_2) \wedge \mathbf{old}(x_2), \mathsf{pro} y_2 \subseteq y_1\,(\mathbf{bpr}(x_2, y_2)))$

e. *And they*$_2$ *love them*$_1$ *right back.*

$\mathsf{pro} y_2 \subseteq y_1\,(\mathsf{pro} x_2 \subseteq x_1\,(\mathbf{bfp}(x_2, y_2)))$

f. *And she*$_2$ *loves him*$_1$ *right back.*

$\mathsf{pro}^{\mathsf{s},\delta} y_2 \subseteq y_1\,(\mathsf{pro}^{\mathsf{s},\delta} x_2 \subseteq x_1\,(\mathbf{bfp}(x_2, y_2)))$

g. *Yesterday, several dogs dug holes in my garden.*

well… we can forget about this one, can't we.

Every man loves a woman.

First of all, look at the antecedent sentence. This is ambiguous between two readings. the first, where *a woman* is singular, and the second, where *a woman* is distributive. Let us see how an output state of this sentence looks for both meanings. Below, H will denote an arbitrary output of the antecedent sentence. It will have the following properties for the singular case:

$H(x_1)$ = the set of all men,

$H(y_2)$ = a set of women such that for every man there is exactly one woman that he loves. The latter is encoded by the fact that for every man d,

$H|_{x_1=d}(y_1)$ = a singleton set containing one woman d loves.

And for the distributive case:

$H(x_1)$ = the set of all men,

$H(y_2)$ = a set of women such that for every man it contains all the women that he loves. The latter is encoded by the fact that for every man d,

$H|_{x_1=d}(y_1)$ = the set of women d loves.

Below, I will only discuss the singular case. It is not difficult to rephrase the next explanations to fit the distributive reading; I will therefore not discuss this reading separately. Let us now go through the cases one at a time. It starts simple.

They bring them flowers to prove this.

Here x_2 is identified with x_1 and y_2 with y_1, and of these it is claimed that x_2 bring x_1 flowers to prove they love them. The pronouns *they* and *them* are ambiguous between collective and (pseudo-) distributive.

He brings her flowers to prove this.

This case serves to illustrate how the dependencies encoded in the state can be used to redistribute over. The singular pronoun *he* enforces distributivity (cf. the discussion in section 2.1), the rest of the sentence is interpreted under a distribution over x. This means that for every man d, $\mathsf{pro} y_2 \subseteq y_1\ (\mathbf{bfp}(x_2, y_2))$ is interpreted in the state $H|_{x_1=d}$. In this state, the pronoun will pick up the unique woman loved by d. Of this value $H|_{x_1=d}(y_1)$ it is claimed that she is loved by d. Note that this case is similar in form to the famous example

> *Every chess set comes with a spare pawn. It is glued to the bottom of the lid.*

Every old man brings her a present to prove this.

This case is different. Here a new quantification is performed, and the anaphoric value serves as the context set to that quantification. Suppose the output state is K. Variable x_2 is bound to the set of old men $K(x_2)$ among the men in x_1. Because of the definition of $\varepsilon_{x_2 \subseteq x_1}$ the resulting state has the property that for every old man e, $K|_{x_2=e}(x_1) = d$ and $K|_{x_1=e} = K|_{x_2=e}$. The two variables are "synchronized". Because for every man d, $K|_{x_1=d}(y_1)$ is the woman d loves, and because of this synchronization, $K|_{x_2=d}(y_1)$ is this same woman. Consequently, when the distributive universal interprets $\mathsf{pro^s} y_2 \subseteq y_1\ (\mathbf{bfp}(x_2, y_2))$ in a state $K|_{x_2=e}$ for every old man e, the singular pronoun sets y_2 the same value as y_1 and claims of this that x_2 (e) brings y_2 ($K|_{x_2=e}(y_2)$) flowers. In the final state L after processing the whole sentence, $L(y_2)$ will be a subset of $L(y_1)$: it will consist of those women loved by an old man. The pronoun in the scope of the distribution operator only assigns y_2 the value of y_1 for subsets $L|_{x_2=e}$ where e is an old man. For d a young man, the pronoun never "sees" $L|_{x_2=d}$, and hence no value for y_2 will be introduced ($L|_{x_2=d}(y_2) = \emptyset$). Furthermore, although not mentioned in the translation, a present will be introduced for every y, but the corresponding variable will be undefined for men that are not old. Another dependency will be created.

The old men bring them presents to prove this.

This case is similar, although here the work is done, not by the distribution operator δ, but by the collection operator γ. Again the subject binds x_2 to the set $E \subseteq H(x_1)$ that consists of the old men. The output of the anaphoric sentence is the union L of $K|_{x\notin E}$ and the output of $\mathsf{pro} y_2 \subseteq y_1\ (\mathbf{bpr}(x_2, y_2))$ with input state $K|_{x\in E}$. consequently, $L(y_2)$ will again be the set of woman loved by old men, and the rest of the state is also equal to the output of (c).

And they love them right back.

And she loves him right back.

Cases (e) and (f) are essentially equivalent to (a) and (b). In (e), y_2 is first equated with y_1 and then x_2 with x_1, but the result is similar to (a): a predication over the men and women. In (f), y_2 is equated with y_1 and then distributed over. for every woman f, $\mathsf{pro^s} x_2 \subseteq x_1\ (\mathbf{bfp}(x_2, y_2))$ is interpreted in the state $H|y_2 = f$, and in that state, x_2 is equated with x_1. Note that the inner pronoun is also distributive. This is the default, because the discourse grammar rules that implement the resolution mechanisms (as said before, topic not dealt with in this thesis) prefer resolution resulting in structurally parallel structures. The distributivity of the outer pronoun is preferred, because singularity would demand there to be one woman for all men, a highly unlikely situation. Logically, both readings are just as likely. Parallelism alone may even prefer the singular reading, depending on the discourse parsing chosen, but world-knowledge makes the distributive reading the one preferred.

3.2 Monotonicity Properties and Readings

Upto now, I did not discuss examples of determiners which are monotone decreasing (MON$\downarrow$) in their second argument. These pose a particular problem, because of the way they interact with negation. This is the next subject. To know why this discussion is important, I will first repeat some observations about the relation between the monotonicity properties a particular quantifier has, and the dynamic readings it allows.

Kanazawa

In 1993b, Kanazawa observes, that the different readings discussed by, amongst others, Chierchia (1992) and van Eijck and de Vries (1991), behave in a strikingly regular way. There is an almost 1-1 correspondence between the preferred reading and the monotonicity properties of that quantifier. In the DPL version of quantifiers, quantification is ambiguous between a strong form, $Q(\phi, \phi \rightarrow \psi)$, and a weak form, $Q(\phi, \phi \wedge \psi)$. The strong form implies a universal quantification over all variables that are bound by ϕ, because of the negation implied in $\rightarrow$, whereas the weak form merely implies existential quantification [1:2.3 *p*.22].

Kanazawa illustrates the ambiguities using the following example sentences and corresponding paraphrases:

(17) ($\downarrow$MON$\uparrow$): *Every student who borrowed a book from Peter returned it.*

Every student who borrowed a book from Peter returned every book he or she borrowed form Peter.

(18) ($\downarrow$MON$\downarrow$): *No student who borrowed a book from Peter returned it.*

No student who borrowed a book from Peter returned a book he or she borrowed from Peter.

(19) ($\uparrow$MON$\uparrow$): *At least two students who borrowed a book from Peter returned it.*

At least two students who borrowed a book from Peter returned a book he or she borrowed from Peter.

(20) ($\uparrow$MON$\downarrow$): *Not every student who borrowed a book from Peter returned it.*

Not every student who borrowed a book from Peter returned every book he or she borrowed from Peter.

He concludes from this, that the relations between monotonicity and weak/strong readings are as given this table.

		strong/weak
$\downarrow$MON$\uparrow$	*every*	strong preferred
$\downarrow$MON$\downarrow$	*no*	weak only
$\uparrow$MON$\uparrow$	*at least two*	weak only
$\uparrow$MON$\downarrow$	*not every*	strong preferred?

This table (repeated directly from Kanazawa (1993b)) already explicitly mentions two problems. The literature definitely allows *every* to have both readings, In fact, we already saw, that *every* can take the weak reading almost as easily as the strong one, depending on the exact lexical content of its components. Also, it is not clear whether the strong reading is the only one possible for *not every*. Hence the “preferred?” in the table.

We also have another problem. The notion of weak and strong is closely linked to the particular form that generalized quantifiers take in DPL. These notions are not directly applicable to the current framework. The first thing we have to do to be able to do something with Kanazawa’s observations is to rephrase them in our language.

To do this, first remember (or cf. section 2.2) that the two readings of *Every farmer who owns a donkey beats it* (traditionally called the donkey and dime reading) correspond to an ambiguity in the NP *a donkey*. If it is interpreted as purely singular, only one entity is introduced at a time, and the corresponding reading is the dime or weak reading. If the NP is interpreted distributively, the set of all donkeys (of that farmer) is introduced, and the donkey or strong reading results.

The result is the following addition to Kanazawa's table. I added two columns. The first (δ_x) documents how many of the borrowed books per student are involved, all or some, in the case of the distributive interpretation, the second documents this for the singular case.

		strong/weak	δ_x	singular
↓MON↑	*every*	strong preferred	**all**	**some**
↓MON↓	*no*	weak only	**all**	**all**
↑MON↑	*at least two*	weak only	**all**	**some**
↑MON↓	*not every*	strong preferred?	**all**	**all**

My explanation of the phenomena is to take the negation in the quantifiers seriously, so *not every* Φ really *is not (every* Φ*)* and *no* is *not some*.

Let us go through these cases. The first row is the one we already know. There is an obvious difference between the truth conditions of *every(student and the books borrowed)* and *every(student and some book borrowed)*. But in the second row, the truth conditions of *not some(student and the books borrowed)* and *not some(student and some book borrowed)* are the same. Both are false if there is a student and some borrowed book such that it is returned. The introduced referent is highly different, though. In the distributive case, the set of books is introduced in the context, in the singular case one, non-unique, book is introduced. In the latter case, the output will in general not be unique, different values will be possible. The other two are similar. In the case of *at least two* (and *some*) the result is ambiguous between a reading that claims that all books are returned for at least two students, and one that claims that for at least two students some, but not necessarily all, books are. The last row is similar to *no* again. Whether the set of books is not returned as a whole, or one book in particular is not returned does not influence the truth value, although it does influence the dynamic behavior.

4 Information Packaging

The standard theory of generalized quantification, and also the theory discussed so far, simplifies the treatment of predication by assuming that there are only two parts to a quantification: a part introducing the entities the predication is about (the quantifier restriction), and a part that is predicated of it (the nuclear scope). The quantifier then tells us how the predication holds of the entities. However, real life discourse units, the objects of language that translate as predications exhibit much more subtlety —expressing extra information like non-restrictive subordinate clauses and adjectives.

Take the following minimal pair, where in the second sentence, the subordinate clause is supposed to be a comment:

(21) *Exactly three women who went to Paris with a boyfriend returned to Amsterdam this morning.*

(22) *Exactly three women, who went to Paris with a boyfriend, returned to Amsterdam this morning.*

The first is a simple quantificational format, claiming that of the women who went to Paris with a boyfriend, exactly three returned this morning. The second expresses that exactly three women returned this morning to Amsterdam, and they happen to have been to Paris with a boyfriend.

In English, these "subtleties" are mainly expressed by intonation, but other languages (for example Hungarian) express similar subtleties by configurational mechanisms.

However, he only way to express this in generalized quantifier theory is to displace the phrase, and construct something like $Qx(\mathbf{women}(x), \mathbf{return}(x)) \wedge \mathbf{toParis}(x)$. Order is important in language, and you cannot arbitrarily change the order of terms in an expression, and think you can get away with it. Just to illustrate this basic fact, compare the following two sentences

Some girls first went to London. Exactly three women, who went to Paris next, returned to Amsterdam on Friday.

Some girls first went to London. Exactly three women returned to Amsterdam on friday. They went to Paris next.

The temporal order is broken when phrases are moved around. That moving phrases around changes the meaning of the discourse will not come as a surprise, but it poses a problem nonetheless. If we want formalize non-restrictive clauses in a dynamic semantics, we cannot do this be moving the phrase around, the interpretation has to be *in situ*.

In the case of definites, the difference becomes important, though the technical problem is the same for normal quantifiers, because of the difference of what is actually picked up. There is a contrastive stress intonation

The stupid FARmers were still protesting.

which picks up all the farmers in the antecedent set, and then claims of these that they are stupid and are protesting, whereas the intonation

The STUpid farmers were still protesting.

picks up the stupid farmers in the antecedent set.

As it turns out, the generalized dynamic quantifier approach gives a very natural method to express these differences in meaning. However, the abbreviated form is not very suited for this. In the extended notation, we can start making a stab at this. Take (21–22). suppressing the variety operator to simplify the notation a bit, these translate as:

(21′) $\varepsilon_{x'\subseteq y} \wedge \varepsilon_{x\subseteq x'} \wedge \mathsf{M}_{x'\subseteq y}\,[\mathbf{w}(x') \wedge \mathbf{wparis}(x')] \wedge \mathsf{M}_{x\subseteq x'}\,[\mathbf{ret}(x)] \wedge 3(x', x)$

(22′) $\varepsilon_{x'\subseteq y} \wedge \varepsilon_{x\subseteq x'} \wedge \mathsf{M}_{x'\subseteq y}\,[\mathbf{w}(x')] \wedge \mathbf{wparis}(x) \wedge \mathsf{M}_{x\subseteq x'}\,[\mathbf{ret}(x)] \wedge 3(x', x)$

(22″) $\varepsilon_{x'\subseteq y} \wedge \varepsilon_{x\subseteq x'} \wedge \mathsf{M}_{x'\subseteq y}\,[\mathbf{w}(x')] \wedge \mathbf{wparis}(x') \wedge \mathsf{M}_{x\subseteq x'}\,[\mathbf{ret}(x)] \wedge 3(x', x)$

The first is the standard reading: x' is assigned the set of all women in the context (y) that went to Paris. Then x is assigned the subset of x' that returned to Amsterdam. Of x it is claimed that the cardinality is 3. In (22′), x' is assigned the set of all women in the context (y). Then it is claimed that x, to be specified later, consists of entities that went to Paris, and finally the subset x of women is set to those entities that returned to Amsterdam. Of x it is claimed that the cardinality is 3. In (22″), x' is assigned the set of all women in the context (y). Then it is claimed that all these women went to Paris, and finally the subset x of women is set to those entities that returned to Amsterdam. Of x it is claimed that the cardinality is 3. This last is the most unlikely of the three readings, but this may be made more probable by a reasonable context.

Once we are prepared to allow for predicates to be in different places than inside the maximization operators, a number of other phenomena come within our reach. The most important of these are adjectival quantifiers and the cumulative and branching readings.

4.1 Adjectival Quantifiers

Adjectival quantifiers are encoded in a similar way. It is assumed that such quantifiers are used in the construction of the set rather than a predicate of them. As an example,

Four women entered the bar yesterday.

has a reading where there were more women who entered, but for some reason the speaker intends to start speaking about four specific ones:

$$\varepsilon_{x' \subseteq y} \wedge \varepsilon_{x \subseteq x'} \wedge \mathsf{M}_{x' \subseteq y}\,[\mathbf{wom}(x')] \wedge \mathsf{M}_{x \subseteq x'}\,[\mathbf{ent}(x) \wedge 4(x', x)]$$

Because the quantifier is in the scope of the maximization operator, the set introduced will have to satisfy it, which means that the set x introduced consists of 4 women. Note that if there were more than 4 women that entered the bar yesterday, the referent will not be unique. The utterance only makes clear that the speaker intends to talk about 4 specific ones, but it does not show which 4 particular ones she means. We cannot second guess her, we can only encode what the possible values are. Note that effectively, this makes an adjectival quantifier into something that behaves as an indefinite, which is how adjectival quantifiers are often explained (Bartsch (1973)).

4.2 Cumulatives and Branching Readings

In section 4.4 of chapter I I already alluded to the fact that the two readings of

Every man loves a woman.

are analogous to the two of the readings of

600 women lift 300 pianos.

This sentence has, at least, four readings. First, the distributive reading, which claims that each of the 600 women lifted 300 pianos. Second, the collective reading, which claims that the 600 women together lifted 300 pianos (Guinness book of records stuff). The third reading, brought to our attention by Scha (1981), is the cumulative, claiming that women lifted pianos, and there are 600 women involved in these liftings, and 300 pianos. The fourth reading is the reading that is identical to the distributive reading, except that for every woman, it is the same 300 pianos. This last reading is sometimes called the branching reading (cf. Barwise and Cooper 1979, Westerståhl 1987, Sher 1990, Beghelli et al. 1995).

Note that the case of *every man loves a woman* there are only two distinct readings. Because of the lexical marking of *every* as distributive the collective reading is not present. Furthermore, because of the singularity of the object NP, the cumulative and branching reading are identical. The latter is what is sometimes called the wide-scope object reading, because it is equivalent to *a woman is loved by every man*. This last fact is an accident, a similar scope inversion does not result in the right meanings of the general numeral case, although the same ambiguity exists [1:4.4]. For example, in the real scope-reversed reading the distributive reading of *300 pianos are lifted by 600 women* might involve different set of 600 women for every piano. Which is not a valid interpretation of the original sentence [1:4.4].

Just as in the case of non-restrictive clauses, we need to use the extended notation. I propose the following analyses for the cumulative and branching readings.

(23) $$\begin{aligned}&\varepsilon_{x'} \wedge \varepsilon_{x \subseteq x'} \wedge \mathsf{M}_{x'}\,[\pi_x(\mathbf{w}(x'))] \qquad \text{(cumulative)}\\ &\wedge \mathsf{M}_x\,[\pi_x(\varepsilon_{y'} \wedge \varepsilon_{y \subseteq y'} \wedge \mathsf{M}_{y'}\,[\mathbf{p}(y')] \wedge \mathsf{M}_y\,[\mathbf{lift}(x,y)])]\\ &\wedge 300(y',y) \wedge 600(x',x)\end{aligned}$$

(24) $$\begin{aligned}&\varepsilon_{x'} \wedge \varepsilon_{x \subseteq x'} \wedge \mathsf{M}_{x'}\,[\pi_x(\mathbf{w}(x'))] \qquad \text{(branching)}\\ &\wedge \mathsf{M}_x\,[\varepsilon_{y'} \wedge \varepsilon_{y \subseteq y'} \wedge \pi_x(\mathsf{M}_{y'}\,[\mathbf{p}(y')] \wedge \mathsf{M}_y\,[\mathbf{lift}(x,y)])]\\ &\wedge 300(y',y) \wedge 600(x',x)\end{aligned}$$

The first of these introduces, for every women, the set of pianos that that the women lifted alone or in groups. The output will assign to y the set of pianos lifted by any woman. Of this value, it is claimed that it has cardinality 300. A similar reading exists with δ rather than π, which only counts the pianos lifted by individual women.

The second is similar, but the set $y \subseteq y'$ is introduced outside the scope of the distribution operator. Consequently, values in y do not depend on values in x. then the values of x are distributed over, and for every element it is claimed that it lifted the 300 piano's. in this case, the piano's are lifted collectively, not a mean feat. Of course, the more natural interpretation is a pseudo-distributive reading for the piano's too:

(25) $$\varepsilon_{x'} \wedge \varepsilon_{x\subseteq x'} \wedge \mathsf{M}_{x'}\left[\pi_x(\mathbf{w}(x'))\right] \qquad \text{(cumulative}')$$
$$\wedge\, \mathsf{M}_x\left[\pi_x(\varepsilon_{y'} \wedge \varepsilon_{y\subseteq y'} \wedge \mathsf{M}_{y'}\left[\pi_{y'}(\mathbf{p}(y'))\right] \wedge \mathsf{M}_y\left[\pi_y(\mathbf{lift}(x,y))\right])\right]$$
$$\wedge\, 300(y',y) \wedge 600(x',x)$$

(26) $$\varepsilon_{x'} \wedge \varepsilon_{x\subseteq x'} \wedge \mathsf{M}_{x'}\left[\pi_x(\mathbf{w}(x'))\right] \qquad \text{(branching}')$$
$$\wedge\, \mathsf{M}_x\left[\varepsilon_{y'} \wedge \varepsilon_{y\subseteq y'} \wedge \pi_x(\mathsf{M}_{y'}\left[\pi_{y'}(\mathbf{p}(y'))\right] \wedge \mathsf{M}_y\left[\pi_y(\mathbf{lift}(x,y))\right])\right]$$
$$\wedge\, 300(y',y) \wedge 600(x',x)$$

However, even this does not really give the right meaning. For suppose that thee are four people, and every pair lifted a piano, then six pianos are lifted, but this formula, counting pianos by counting covers generated by individuals, will only "see" four. The operator σ_x, defined in section 5 of chapter 4 would get this right, because it introduces arbitrary covers, but it is too strong by far. These won't come as a surprise, they have been discussed by a large number of authors (van der Does 1992, Verkuyl 1994) who at best partly solve it for some cases, and I cannot really give a solution either. In fact, this is one place where the two-stages analysis of the pseudo-distributive would come in handy. If the we explain pseudo-distribution —or rather, what pseudo-distributivity should be— in terms of a distribution over events, then it may be possible to maximize the set of lifting events, and then afterwards count the individuals, and pianos, involved (rather than stages of individuals). This discussion will have to be postponed to later.

5 Future Extensions: Other Anaphora

In this section, I will briefly sketch how the theory developed here might be applied to other cases of anaphora, and what changes have to be made.

5.1 Other Relations

So far, we have only been concerned with anaphora which introduce, in some way, a subset of their antecedent. In language, other links exist, and has sometimes been claimed that these other links are the majority of "glue" in language. Let us look at some typical examples:

(27) *A*^{1} *car entered the village. The*$^1_2$ *driver looked tired.*

(28) *Some*^{3} *children played in the park. The*$^4_3$ *mothers were sitting on a bench.*

In the first sentence of (27), the referent x_1 is introduced with a car as value. In the second sentence, the driver of the car picked up, and that driver is claimed to be tired. In the first sentence of (28), the referent x_1 is introduced with the children as value. and in the second sentence, the mothers of those children are picked up.

The question is, how can we pick something that is not explicitly introduced? One answer is to assume that it *is* something explicit that is picked up. This is done by translating the definite as a functional relation. E.g. *driver* corresponds to 2-place relation between individuals and their cars, And *mother* corresponds to 2-place relation between individuals and their offspring. This poses a number of technical problems. The most important one is, that the relation is often reciprocal. It is just as easy to say:

(29) *Some*^{3} *mothers were sitting on a bench. The*$^4_3$ *children played in the park.*

And all of these expressions can occur on their own, without the functional part of the content being used.

Within dynamic logic, it is easy to solve these technical problems by a method first discussed by Dekker (1992), called existential disclosure. The trick is to bind all extra variables with an existential quantifier. So *driver* translates as $\exists y \subseteq z\ (\mathbf{driver} - \mathbf{of} xy)$, an expression valid of x if there is some y with antecedent x that x drives, and similar for *mother*. Example (27) then translates as:

(30) $\exists^{\delta} x\ (\mathbf{c}(x), \mathbf{play}(x)) \wedge \mathsf{def}\, y\ (\exists^{\delta} z \subseteq x\ (\mathbf{m}(y, z)), \mathbf{sob}(y))$

This is a very tempting method, which works quite well. It does lack in the aesthetics department, though. Intuitively, it is definite the mother that picks up the referent, not some hidden quantifier falling from the air. In chapter 4, a notion of dependent random assignment that can mimic the effect of an internal existential quantifier. The translation becomes (cf. page 138 and proposition 5.1 on page 153)

(31) $\exists^{\delta} x\ (\mathbf{c}(x), \mathbf{play}(x)) \wedge \mathsf{def}\, y \mathbf{of} x\ (\delta_y(\mathbf{m}(y, x)), \mathbf{sob}(y))$

The definite introduces a new referent y, that co-varies with x (i.e. different y may correspond with different x) such that for every element d of y, d is the mother of the values that this d co-varies with.

5.2 Other Nominal Anaphors

There are several context sensitive expressions in language that do not directly follow the example of determiners/pronouns. Here, I will briefly discuss the most important of these, to wit *one*, *another*, and the slightly different category of floating quantifiers *each* and *all*.

one-anaphora

Certain anaphora pick up the CN variable from an antecedent rather than the normal value. Let us start with the expression that gives it this name

Mary has a cat. Susan has one too.

$\exists^{\mathsf{s},\gamma} z_1\ (\mathbf{mary}(z_1), \exists^{\mathsf{s},\gamma} x\ (\mathbf{cat}(x), \mathbf{have}(z_1, x)))$

$$\wedge \exists^{\mathsf{s},\gamma} z_2 \, (\mathbf{susan}(z_2), \mathsf{pro}^{\mathsf{s}} x_2 \subseteq x' \, (\mathbf{have}(z_2, x)))$$

A more adventurous example is given by *another*. It picks up both the normal and CN value and does some fancy index gymnastics on them to introduce a new value. As a simple, but may be not quite correct, formalization, take

> *A^1 cat sat on the mat. An_1other walked towards it.*
>
> $\exists^{\mathsf{s},\gamma} x_1 \, (\mathbf{cat}(x_1), \mathbf{otm}(x_1)) \wedge \varepsilon_{x_2} \wedge x_2 = x_1 \wedge [\mathsf{a}^{\mathsf{s}} y \subset x_1' \, (\mathbf{other}(y, x_2), \mathbf{otm}(y))]$

Where I assume that the expression $\mathbf{other}(y, x')$ is true, if y is disjoint from x. Consequently the second phrase introduces a variable y that is in the CN extension of the antecedent, but contains different elements from x. It also introduces its own CN extension, which will be $x' - x$, the antecedent CN value minus the x-value. This makes it possible to stack *other*.

(32) *A cat sat on the mat. Another walked to it. and (yet) another just looked through the window.*

Floating Quantifiers

Floating quantifiers also have something anaphoric. They definitely link with a determiner that they are not directly next to. However, consider some simple example

(33) *Three boys bought a pizza together and a soda each.*

It seems clear to me, that what word like *together* and *each* do is fill the position of the variety operator. Especially in the case of floating *each* there is almost no doubt in my mind, that its interpretation is exactly equal to δ_x for some appropriate x. Whether *together* is equal to γ_x is less sure, maybe it does add some more glue (cf. van der Does 1992, Lønning 1987a, Lønning 1987b, Schwarzschild 1992a).

Adding Variables Together

One case I did not mention at all in the above is the summing of variable values, as illustrated by

(34) *John^1 met Mary^2 in Paris. they$_{1+2}$ had a great time.*

(35) *First a^1 man entered the bar. then a^2 woman followed him. They$_{1+2}$ started to fight. Finally, she$_2$ shot him$_1$.*

(36) *When a^1 student of the physics department marries a^2 student of the mathematics department, the$^3_{1+2}$ man usually finishes his$_3$ studies before the$^4_{1+2}$ woman does, but she$_4$ gets better notes.*

Can this be formalized? Or, to put it more precisely,can we give a meaning to a random assignment of the form $\varepsilon_{y \subseteq x_1 + x_2}$, such that the meaning of (34) is given by:

(37) $\exists^{\mathsf{s},\delta} x_1 \, (\mathbf{john}(x_1), \exists^{\mathsf{s},\delta} x_2 \, (\mathbf{mary}(x_2), \mathbf{meet}(x_1, x_2)) \wedge \mathsf{pro} y \subseteq x_1 + x_2 \, (\mathbf{fun}(y))$

Aside 5.2 The Difference between the Quantifiers Each and Every

As a quantifier *each* is the distributive universal quantifier. What then is the difference between *each* and *every*? It seems to me that we can only explain this using events, something we explicitly omitted in this theory, that concentrates on nominal phenomena. Take the following minimal pair

a. *Every boy entered the bar.*
Each boy entered the bar.

Do you not share with me the intuition, that whereas in the case of *every*, the important point is, no boy did not enter the bar, the case of *each* seems to imply that the boys did this separately. That, when watching this happen we saw first one boy, then another, then another, etc. It seems to demand that there were different entering events for the boys, whereas *every* allows some boys to enter together, as long as the togetherness is not essential. That it is not separation in time that matters is shown by the following pair

b. *Every girl stood on a black dot.*
Each girl stood on a black dot.

Again, *every* does not mind that several girls stood on the same dot, but *each* seems to prefer separateness, one dot per girl. The exact implementation of this intuition, and research into whether this intuition is correct at all, is left to another place.

The problem with this is that x and y may have relationships with other variables and with each other. This complicates matters considerably. All is not lost though, we only need minor modifications to the formalism to make it possible to encode added variables. First, consider a simpler case. Suppose x_1 and x_2 were introduced as subsets of z

(38) $\exists^{\mathsf{s},\delta} x_1 \subseteq z\ (\mathbf{john}(x_1), \exists^{\mathsf{s},\delta} x_2 \subseteq z\ (\mathbf{mary}(x_2), \mathbf{meet}(x_1, x_2)) \wedge \ldots$

And suppose furthermore that x_1 and x_2 do not have an overlap. Then the following will introduce a summed object $x_1 + x_2$ that can made to preserve all dynamics in similar ways to $\varepsilon_{x \subseteq y}$

(39) $\delta_{x_1}(\varepsilon_{y \subseteq x_1}) \wedge \delta_{x_2}(\varepsilon_{y \subseteq x_2})$

First, this introduces y for the part of the state for which x_1 is defined, then y is introduced for the part where x_2 is defined. This works if x_1 and x_2 are disjoint subreferents of some variable z, because then δ_{x_1} will evaluate its argument for the sub-state for which x_1 is defined (and therefore x_2 is undefined) and δ_{x_2} will evaluate its argument for the sub-state for which x_2 is defined (and therefore x_1 is undefined). They *have* to be subsets of the same variable and they *have* to be disjoint, otherwise there may be an overlap on which they are both defined and then the second random assignment will result in undefined.

We can do slightly better than this. The existence of z makes it possible to pick *it* up rather than its derived referents x_1 and x_2:

(40) $\varepsilon_{y \subseteq x_1 + x_2} := \varepsilon_{y \subseteq z} \wedge \mathsf{M}_{y \subseteq z}\,[y \subseteq x_1 \wedge y \subseteq x_2]$

where $y \subseteq x_1$ is the normal subset predicate that does not necessarily preserve dependencies (the first conjunct already does this).

This solution is still somewhat unsatisfactory. It only works if all variables in the sum were introduced as subsets of one superset, something for which there does not seem any *a priori* reason. This means that someone hearing only these two sentences would not only not be able to interpret them but even not be able to judge whether this is well-structured at all. Surely, this does not make sense. What happens is that such a superset is created at the moment of interpretation of the sum, *accommodated*, if you will.

I will now give a definition of $\varepsilon_{x\subseteq y}$ that satisfies everything we want it to. This is done, not by explicitly introducing a variable corresponding to the superset, but by defining a transformation on states that results in a state with properties *as if* there is such a variable:

5.1 Definition (Summed Random Assignment)
The summed random assignment ($\varepsilon_{y\subseteq x_1+x_2}$) is defined by:

$$\begin{aligned} G[\![\varepsilon_{y\subseteq x_1+x_2}]\!]^d H \quad \text{iff} \quad & G(y)=\emptyset \mathbin{\&} \exists H_1, H2 \;\; H = G\cup H_1 \cup H_2 \mathbin{\&} \\ & G[x_1 := \star][\![\varepsilon_{y\subseteq x_2}]\!]^d H_1 \mathbin{\&} G[x_2 := \star][\![\varepsilon_{y\subseteq x_1}]\!]^d H_2 \end{aligned}$$

The definition given consists of two steps, first G is mapped onto $G[x_1 := \star] \cup G[x_2 := \star] \cup G$, which according to lemma 3.6 of on page 100 of chapter 3 expresses the same dependencies. Then variable y is bound, introducing new dependencies, but preserving the old ones. Note the resemblance with (39). In the definition we make sure that x_1 and x_2 do not get in the way of each other.

This is only half the solution of example (15e), because this does not give a way of defining the conjunction of two noun-phrases. For this the following gives a simplified, no variety operators or context sets, translation:

A man and a woman entered.
$\varepsilon_x \wedge \mathsf{M}_x\,[\mathbf{m}(x)] \wedge \varepsilon_y \wedge \mathsf{M}_y\,[\mathbf{w}(y)] \wedge \mathsf{pro} z \subseteq x+y \;\wedge \mathsf{M}_{z\subseteq x+y}\,[\mathbf{w}(z)]$

Where $\mathsf{pro} z \subseteq x+y$ is not a real pronoun (nor an PRO in the syntactician's sense), but just something that combines the two referents together. But here again, we reach the point in this further extensions section where I have to postpone further discussion to later.

5.3 Committees and Couples

Certain singular words like *couple*, *group* or *committees* denote plural objects themselves. However, these are actually singular, which are counted as one individual, as shown by:

(41) *Four couples bought a pizza.*

which suggests that four, not eight, pizzas are sold. And even if you maintain that this is because plurals of such words in one way or another encapsulate the plural objects, but the singulars simply denote sets, what about the following two examples:

(42) *Every couple bought a pizza.*

(43) *Four couples bought a pizza. The women ate their part directly, but the men waited.*

In (42), still four pizzas, in the same situation, passed hands, and in (43), the men and women are retrieved from the four couples. This suggests that the simplification of treating *a married couple* in (15f–g) as a set was indeed, as I admitted, cheating, and does not mirror what actually happens.

The problem with these words is that a pronoun referring to them can either be singular, and refer to the object itself, or plural, and refer to the members of the object[1]. It is not difficult to implement this in this logic, we just have to add plural pronouns that refer back via $\varepsilon_{x' R_{\subseteq} x}$, where $R_{\subseteq}$ is the internal subset relation. it would even get all dependencies right, and it would also be easily recognizable for the grammar that this applies, because it may be seen as lexically marked in the antecedent. But it still feels uncomfortable, and I would like to find a deeper solution (cf. Schwarzschild 1992b, Chierchia 1996).

6 Future Extensions: Non-nominal Anaphora

That non-nominal anaphora behave in a similar way as nominal anaphora is something that has been observed by a large number of different authors. For temporal anaphora, this was discussed by Partee (1984), and for modal indices, a similar observation was made by Kibble (1994).

Modal Subordination

Modal indices are very close to nominal anaphora, and in many ways, even simpler, once we get over the fact that it is not clear at all what they are indices of. Without assuming to much about the ontology of modals, a subject far beyond the boundaries of this thesis, it seems to assume that modal terms in language are about partial worlds ("events' or even just "worlds"), which I will call situations. That they are simpler is caused by the fact that situation do not come in crowd, they are always inspected distributively, one after the other. Take the following classic, discussed by Roberts (1983)

(44) *A thief might come in. He would steal the furniture.*

In this, the first sentence is about certain (partial) worlds, namely those in which a thief does enter. The second states of these worlds, that the thief in that world would steal the furniture. Note the similarity between this case and the nominal case:

(45) *Every large town has a cityhall. It's downtown.*

(46) *Every man loves a woman. She loves him right back.*

1. Old style-books on British English even say that this should already be the case for the antecedent sentence for some of these words. For example, according to these, you should say *The government has decided* if it is a decision of the government as a formal unit, but *The government have dined* if it is something that the members do.

The first sentence of example (45) is about certain towns, namely those which are large. The second sentence states of these towns, that the cityhall in that town is downtown. Example (46) was discussed before, and has a similar structure.

To me, these parallels make it clear, that an expression like *might* is essentially an existential quantification distributively introducing a set of situations. The set is the maximal set satisfying the argument of the modal operator. Expressions like *would* pick this set up and predicate some more properties of its elements. Modal subordination, then, is just a normal E-type link, it just involves situations rather than entities. This analysis works, whatever the preferred ontology for modals, provided the language contains variables to denote these modal objects, which seems reasonable, they are needed for the events anyway.

One who tries to give an account of modal subordination broadly under this perspective is Kibble (1994), although his choice of state as a pairs of worlds and assignments, though quite common in the relevant literature, makes him miss the generalization that the cases are absolutely identical.

7 Future Extensions: Discourse Grammar

One thing explicitly omitted from this thesis is a grammatical system that generates the meanings described. The Discourse Grammar I have been working on over the last years together with Prüst (1992, 1994) is based on the Linguistic Discourse Model of Polanyi (1986). The theory so far only generates static logic, because that is enough for VP-anaphora, the topic of Prüst's work. In (van den Berg 1996a) I give a version that generates dynamic logic.

The form of the logical language was chosen to fit as well as possible into the semantic slot of discourse grammar. It is assumed, that sentence grammar generates ready-to-use expressions with unification variables in the position of the context set variables. This means that in those formulas, all variables bind locally, and standard type-logical methods can be used. The discourse grammar will then resolve the unification variables. This introduces links in a way reminiscent of the merge operator in DRT, but using much more structural information of what we called (in Prüst et al. (1994)) the syntacto-semantic structure. This structure is an encoding of part of the information packaging; the part not captured by the methods described above although like it: configurational.

Discourse grammatical considerations do contribute a number of interesting points. I will restrict myself here to two observations. The preference for structural parallelism and the suggestions for the interaction of VP-anaphora and nominal anaphora.

7.1 Parallel Structure

It is a well observed fact, that predications in a list tend to have similar formal properties (aspect, variety etc.) (cf.Lang (1977), Kameyama (1986), Prüst (1992), Prüst et al. (1994)). To give a relevant example

(47) *Every woman bought a pizza and so did seven men.*

(48) *Three women bought a pizza and so did seven men.*

(49) *Three women bought a pizza together and so did seven men.*

In (47) the seven men buy a pizza each, whereas in (48) they bought one together. In (49), either both NPs are distributive are both are collective.

This is one way in which ambiguities are curtailed by the grammar. Not all possible permutations of readings are possible: the grammar will only allow certain combinations. Note another consequence of the existence such parallel structure constraints. At the level of discourse meaning, the (pseudo-) distributivity or collectivity *is* expressed in some way, otherwise synchronization cannot be implemented. This does not mean that Verkuyl is wrong in assuming that the variety is in some sense caused by different ways an event can be rolled out in time, but it does mean that somewhere in this rolling out, operators like the collective/distributive operators are involved.

7.2 Interactions with VP's

Before I end my musings about future extensions of the theory, I have to mention one last curiosity. In Prüst et al. (1994), I co-developed a theory of VP-anaphora that is reasonably good in explaining a large number of phenomena. But that theory does raise a question for dynamic semantics. Foregoing all subtleties, the theory explains VP-anaphora by copying expressions on the logical level. The problem the is: what happens when dynamic quantifiers are copied. Take the following example

(50) *Every man loves a woman and every boy does too.*

Assuming that the meaning of this just involves copying the VP, we get

(51) $$\forall^{\mathsf{s},\delta} x_1\,(\mathbf{man}(x_1), \mathsf{a}^{\mathsf{s}} y_1\,(\mathbf{woman}(y_1), \mathbf{love}(x_1, y_1))) \\ \wedge \forall^{\mathsf{s},\delta} x_2\,(\mathbf{boy}(x_2), \mathsf{a}^{\mathsf{s}} y_1\,(\mathbf{woman}(y_1), \mathbf{love}(x_1, y_1)))$$

This seems to be an incorrect meaning. The second quantifier binding y_1 will try to bind an already bound variable, which it cannot. So the result is an undefined formula. But all is not what it seems. In this kind of list constructions, we can assume that both NPs introduce subsets of the same superset. This is in fact part of the meaning that discourse grammar assigns such structures. This means that we have

(52) $$\forall^{\mathsf{s},\delta} x_1 \subseteq z\,(\mathbf{man}(x_1), \mathsf{a}^{\mathsf{s}} y_1\,(\mathbf{woman}(y_1), \mathbf{love}(x_1, y_1))) \\ \wedge \forall \subseteq \mathsf{z}^{\mathsf{s},\delta} x_2\,(\mathbf{boy}(x_2), \mathsf{a}^{\mathsf{s}} y_1\,(\mathbf{woman}(y_1), \mathbf{love}(x_1, y_1)))$$

This is a completely different case. If x_1 and x_2 are disjoint subsets of z the first quantifier binding y_1 will only "see" the part of the state for which x_1 is defined, and the second will only see the par for which x_2 is defined. This means that this formula *is* defined. It introduces three variables x_1, x_2 and y_1, and binds y_1 to the total of women loved by either men or boys. The two different subsets of women are still available using a construction like *the women of the men*, where *of* is the linguistic realization of the $\mathbf{of}$ in $\varepsilon_{x\mathbf{of}y}$. Further discussion has to wait for another place.

8 Conclusions of the Thesis

8.1 General Conclusions

This research started out with two goals:

- The main, linguistic goal was to develop a semantics for discourse grammar that has a reasonably standard format, so that it is easy to make existing grammar formalisms produce it.
- The further, mathematical goal was to see how far we get with modifications of the dynamically bound variable paradigm while preserving most of the format of standard (generalized quantifier) logic.

As I hope to have made clear in the preceding text, dynamic binding brings us much further than you may at first think. As we saw, a large number of phenomena can be successfully explained.

To achieve this success, the standard theory of dynamic semantics is changed in two ways. First, the system is partialized, and second, formulas are interpreted relative to sets of assignments. Both moves have been proposed by other authors, and are well argued for in the text. The move to partial logic can be found in most work on presupposition (most recently, in the dissertation of Krahmer (1995)), and the idea to use sets of partial assignments is independently proposed in current logic research (Németi 1993) to encode dependencies between variables.

The result is a formalism that achieves the linguistic goal, although the definition of the discourse grammar itself has to wait for another place: the first results are published in (van den Berg 1996a). The syntax of the formal language was kept virtually identical to generalized quantifier theory with context sets, making it feasible to adapt standard grammar formalisms to generate sentence and discourse translations. Early results (van den Berg 1996a) suggest that methods already developed for static logic, as, for example, defined by Prüst et al. (1994) can be modified without any problems.

8.2 Conclusions: Linguistics

Discourse analysis is a central topic of current linguistics, and giving a reasonably complete language for its semantics is therefore an important research topic. I hope to have shown that the logical language developed in this thesis extends in a number of major ways the current state of the art (Kamp and Reyle 1993, and work based on that) by giving a logical language that can express dependencies between quantifier domains and generates the plural referents automatically rather than in a separate construction.

The role of **maximization**, already discussed in the literature, is firmly rooted with one foot in standard plural generalized quantifier theory, where it served to reconstruct the standard readings, and with the other foot in dynamic logic, where it relates to E-type phenomena.

The logic expresses what might be called **substructural information** about variables values via states expressing dependencies. This is information that in other theories is normally assumed to be saliently given in the non-linguistic context. Here such extra information is only added when the language demands, carefully distinguishing what information *is* produced by world knowledge, and what information the language gives us. The fact that **distribution is the source of dependencies** in discourse may be almost more important than the fact that it constitutes a particular variety of quantification. Although the two roles are so intertwined that it is hard to distinguish them.

8.3 Conclusions: Logic

A side effect of the development of a theory for discourse semantics was the development of the *theory* itself. The (singular) partial dynamic logic discussed in chapter 2 turned out to have some interesting logical properties of its own. It gives a special case of partial relational algebra, with very particular properties. The definition of partial logic in terms of *defined* and *true*, inspired by facts from linguistics, turned out to suggest a different notion of **partial-bisimulation** than you might come up with than the one you get starting with *true* and *false*. It appeared surprisingly easy to find a characterization of the set of operators that is safe for partial-bisimulation in our sense, whereas the case of, as I called it, $^+/_-$-bisimulation is very complicated.

In chapters 3 and 4, I introduced a notion of interpretation relative to sets of assignments, to encode **dependencies** between the elements of plural objects. This relates to work currently done in logic, but applies it in rather different ways. It remains to be seen, whether methods developed here may applicable in the frameworks that logicians need them for. In particular, it is not absolutely clear what the properties of the sub-logic with only singular quantifiers and distributive predicates are, which is the case logicians are normally concerned with. However, the formalism introduced gives a very rich logical language to analyze dependency structures, that extends on the work of Németi (1993), Alechina and van Lambalgen (1995), and van Benthem (1996).

Bibliography

Alechina, N.: 1995a, *Modal Quantifiers*, Ph.D. thesis, Institute for Logic, Language and Computation (ILLC), University of Amsterdam

Alechina, N.: 1995b, On a Decidable Generalized Quantifier Logic Corresponding to a Decidable Fragment of First-Order Logic, *Journal of Logic, Language, and Information* 4(3)

Alechina, N. and van Benthem, J.: 1993, *Modal Quantification over Structured Domains*, research report ML–93–02, ILLC, University of Amsterdam, To appear in M. de Rijke (ed.), *Advances in Intensional Logic*, Kluwer, Dordrecht.

Alechina, N. and van Lambalgen, M.: 1995, *Generalized Quantification as Substructural Logic*, research report ML–95–05, ILLC, University of Amsterdam

Andréka, H., van Benthem, J., and Németi, I.: 1994, *Back and Forth Between Modal Logic and Classical Logic*, Appeared in *Bulletin of the Interest Group in Pure and Applied Logic*, London and Saarbrücken, August 1995. To appear in Journal op Philosophical Logic

Baker, M. C.: 1992, Unmatched Chains and the Representation of Plural Pronouns, *Natural Language Semantics* 1(1), 33–73

Barker, C. and Dowty, D. (eds.): 1992, *SALT II Proceedings of the Second Conference on Semantics and Linguistic Theory*, No. 40 in Working Papers in Linguistics, Columbus, Ohio, Ohio State University

Bartsch, R.: 1973, The Semantics and Syntax of Number and Numbers, in Kimbal 1973

Barwise, J.: 1987, Noun Phrases, Generalized Quantifiers and Anaphora, in E. Engdahl and P. Gärdenfors (eds.), *Generalized Quantifiers /Linguistic and Logical Approaches*, Studies in Linguistics and Philosophy, pp 1–29, Reidel, Dordrecht

Barwise, J. and Cooper, R.: 1979, On Branching Quantifiers in English, *Journal of Philosophical Logic* 8, 47–80

Barwise, J. and Cooper, R.: 1981, Generalized Quantifiers and Natural Language, *Linguistics and Philosophy* 4, 159–219

Baüerle, R., Schwarze, C., and von Stechow, A. (eds.): 1983, *Meaning, Use and the Interpretation of Language*, de Gruyter, Berlin

Beaver, D.: 1993, *What Comes First in Dynamic Semantics*, Technical report, University of Edinburgh

Beaver, D.: 1995, *Presuppositions and Assertions in Dynamic Semantics*, Dissertation, University of Edinburgh, Edinburgh

Beghelli, F., Ben-Shalom, D., and Szabolcsi, A.: 1995, Variation, Distributivity and the Illusion of Branching, in Szabolcsi 1995, *forthcoming*

Ben-Shalom, D.: 1994, A Tree Characterization of Generalized Quantifier Reducibility, in M. Kanazawa and C. Piñón 1994

Bennett, M. R.: 1974, *Some Extensions of a Montague Fragment of English*, Ph.D. thesis, University of California, Los Angeles

van Benthem, J.: 1984, Questions about Quantifiers, *Journal of Symbolic Logic* 49(2), 443–466

van Benthem, J.: 1986a, *Essays in Logical Semantics*, Vol. 29 of *Studies in Linguistics and Philosophy*, D. Reidel Publishing Company, Dordrecht

van Benthem, J.: 1986b, Partiality and Non-Monotonicity in Classical Logic, *Logique et Analyse* 29, 225–247

van Benthem, J.: 1989a, Polyadic Quantifiers, *Linguistics and Philosophy* 12, 437–464

van Benthem, J.: 1989b, Semantic Parallels in Natural Language and Computation, in Ebbinghaus et al. 1987

van Benthem, J.: 1991, *Language in Action: Categories, Lambdas and Dynamic Logic*, No. 130 in Studies in Logic and the Foundations of Mathematics, Elsevier, Amsterdam

van Benthem, J.: 1993, *Program Constructions that are Safe for Bisimulation*, Technical Report CSLI-93-179, CSLI Prepublication Series

van Benthem, J.: 1996, *Exploring Logical Dynamics*, Cambridge University Press, to appear

van Benthem, J., van Eijck, J., and Stebletsova, V.: 1993, *Modal Logic, Transition Systems and Processes*, Technical Report CS-R9321, CWI

van Benthem, J. and ter Meulen, A. (eds.): 1985, *Generalized Quantifiers in Natural Language*, No. 4 in GRASS, Foris

van Benthem, J. and Westerståhl, D.: 1994, *Directions in Generalized Quantifier Theory*, Research Report LP–94–06, ILLC, University of Amsterdam, Amsterdam, To appear in Studia Logica

Bergstra, J. and Klop, J. W.: 1983, Algebra of Communication Processes, in Lenstra et al. 1986a

Bergstra, J. and Klop, J. W.: 1986, Process Algebra: Specification and Verification in Bisimulation Semantics, in Lenstra et al. 1986b

van den Berg, M. H.: 1990, A Dynamic Logic for Plurals, in Stokhof and Torenvliet 1990

van den Berg, M. H.: 1991a, Dynamic Generalized Quantifiers, in van der Does and van Eijck 1996

van den Berg, M. H.: 1993a, Dynamic Semantics, in M. Pinkal, R. Scha, and L. Schubert (eds.), *Semantic Formalisms in Natural Language Processing*, No. 57 in Dagstuhl-Seminar-Report, pp 5–9, Schloß Dagstuhl, Wadern, germany

van den Berg, M. H.: 1993b, Full Dynamic Plural Logic, in Bimbó and Máté 1993

van den Berg, M. H.: 1994, A Direct Definition of Generalized Dynamic Quantifiers, in Dekker and Stokhof 1994

van den Berg, M. H.: 1996a, Discourse Grammar and Dynamic Logic, in Dekker and Stokhof 1996

van den Berg, M. H.: 1996b, *Models for Dynamic Type Logics* Unpublished Manuscript, ILLC, University of Amsterdam.

van den Berg, M. H. and Prüst, H.: 1990, Common Denominators and Default Unification, in T. van der Woude and W. Sijtsma (eds.), *Computational Linguistics in the Netherlands*, No. OTS-WP-CL-91-001 in OTS Working Papers, University of Uterecht, Papers for the first CLIN meeting

Bimbó, K. and Máté, A. (eds.): 1993, *Proceedings of the Fourth Symposium on Logic and Language*, Budapest

Blamey, S.: 1986, Partial logic, in Gabbay and Guenther 1984, Vol III, pp 1–70

Blass, A. and Gurevich, Y.: 1986, Henkin Quantifiers and Complete Problems, *Annals of Pure and Applied Logic* 32, 1–16

Caenepeel et al. (eds.): 1991, *Workshop on Discourse Coherence*, University of Edinburgh

Cepparello, G.: 1995, *Studies in Dynamic Logic*, Illcc-series ds-1995-10, ILLC, University of Amsterdam

Chang, C. C. and Keisler, H. J.: 1990, *Model Theory*, Vol. 73 of *Studies in Logic and the Foundations of Mathematics*, North-Holland, Amsterdam, third edition edition, First edition: 1973

Chierchia, G.: 1992, Anaphora and Dynamic Binding, *Linguistics and Philosophy* 15, 111–183, An earlier version appeared 1991 in DYANA deliverable R2.2.A

Chierchia, G.: 1996, Mass nouns and the Notion of "Semantic Parameter", in Dekker and Stokhof 1996

Choe, J.-W.: 1987, *Anti-Quantifiers and a Theory of Distributivity*, Ph.D. thesis, University of Massachusetts, Amherst

Cresswell, M. J.: 1990, *Entities and Indices*, Kluwer Academic Publishers, Dordrecht, Boston, London

Dalrymple, M., Kanazawa, M., Mchombo, S., and Peters, S.: 1994, What Do Reciprocals Mean?, in M. Harvey and L. Santelmann (eds.), *Proceedings of the Fourth Semantics and Linguistics Conference: SALT IV*, Cornell University Working Papers in Linguistics

Davidson, D. and Harman, G.: 1972, *Semantics of Natural Language*, The University of Chicago Press, Chicago

van Deemter, K.: 1991, *On the Composition of Meaning. Four Variations on the Theme of Compositionality in Natural Language Processing.*, Ph.D. thesis, University of Amsterdam, Amsterdam

Dekker, P.: 1990, Dynamic Interpretation, Flexibility and Monotonicity, in Stokhof and Torenvliet 1990

Dekker, P.: 1992, Existential Disclosure, Implicite Arguments in Dynamic Semantics, *Linguistics and Philosophy*, An earlier version appeared in Goldberg and Laslo Kálmán and Szábo (eds.) 1992, Proceedings of the Third Symposium on Logic and Language

Dekker, P.: 1993, *Transentential Meditations Ups and Downs in Dynamic Semantics*, Ph.D. thesis, University of Amsterdam

Dekker, P.: 1994, *Dynamic Predicate Logic with Anaphora,*, Technical Report LP–94–11, ILLC

Dekker, P. and Stokhof, M. (eds.): 1991, *Proceedings of the Eight Amsterdam Colloquium*, Amsterdam, ILLC/Department of Philosophy, University of Amsterdam

Dekker, P. and Stokhof, M. (eds.): 1994, *Proceedings of the Ninth Amsterdam Colloquium*, Amsterdam, ILLC/Department of Philosophy, University of Amsterdam

Dekker, P. and Stokhof, M. (eds.): 1996, *Proceedings of the Tenth Amsterdam Colloquium*, Amsterdam, ILLC/Department of Philosophy, University of Amsterdam

van der Does, J.: 1992, *Applied Quantifier Logics. Collectives & Naked Infinitives*, Ph.D. thesis, University of Amsterdam

van der Does, J.: 1993, Formalizing E-Type Anaphora, in Dekker and Stokhof 1994, pp 229–248

van der Does, J.: 1994, In Complex Plural Noun Phrases, in Kanazawa and Piñón 1994, pp 81–115

van der Does, J. and van Eijck, J.: 1996, *Generalized Quantifier Theory and Applications*, CSLI Publications, Stanford, California

van der Does, J. and Verkuyl, H.: 1995, Quantification and Predication, Technical Report LP–93–XX, ILLC

Ebbinghaus, H. D. et al. (eds.): 1987, *Logic Colloquium'87*, North-Holland, Elsevier

Ehlich, K. and van Riemsdijk, H. (eds.): 1983, *Connectedness in Sentence, Discourse and Text*, Tilburg University, Holland

van Eijck, J.: 1985, Generalized Quantifiers and Traditional Logic, in J. van Benthem and A. ter Meulen (eds.), *Generalized Quantifiers in Natural Language*, GRASS Vol. 4, Foris, Dordrecht

van Eijck, J.: 1991a, Quantification, in Wunderlich and von Stechow 1984
van Eijck, J.: 1991b, Quantification and Partiality, in J. van der Does and J. van Eijck (eds.), *Generalized Quantifier Theory and Applications*, pp 143–154, Dutch Network for Language, Logic, and Information, Amsterdam, Also to appear with CSLI Lecture Notes.
van Eijck, J. and de Vries, F.-J.: 1991, Dynamic Interpretation and Hoare Deduction, *Journal of Logic, Language, and Information* 1, 1–44
Elworthy, D. A. H.: 1995, A Theory of Anaphoric Information, *Linguistics and Philosophy* 18(3), 297–332
Evans, G.: 1977, Pronouns, Quantifiers and Relative Clauses (1), *The Canadian Journal of Philosophy* pp 337–362, reprinted in: Evans G., 1985, Collected Papers, Dordrecht, Foris, pp. 214–248
Evans, G.: 1980, Pronouns, *Linguistic Inquiry* 11, 337–362
Feferman, S.: 1984, Towards Useful Type Free Theories I., *Journal of Symbolic Logic* 49, 75–111
Fenstad, J.-E.: 1996, Partiality
Fernando, T.: 1992, Transition Systems and Dynamic Semantics, in D. Pearce and G. Wagner (eds.), *Logics in AI*, No. 633 in LNCS, Springer Verlag, Berlin, subseries LNAI
Fernando, T.: 1994a, Generalized Quantifiers as Second-order Programs –"Dynamically" speaking, naturally, in Dekker and Stokhof 1994, pp 249–268
Fernando, T.: 1994b, Translations in the Dynamic Analysis of Presuppositions, in *Proceedings of the 5th Symposium on Logic and language (to appear)*
Fernando, T.: 1994c, What is a DRS?, in H. Bunt, R. Muskens, and G. Rentier (eds.), *International Workshop on Computational Semantics*, Tilburg
Gabbay, D. and Guenther, F. (eds.): 1984, *Handbook of Philosophical Logic*, Vol. 2, Reidel, Dordrecht
Geach, P. T.: 1972, A Program for Syntax, in Davidson and Harman 1972
Gill, D.: 1989, *Scopal Quantifiers, Some Universals of Lexical Effability*, The Porter Institute for Poetics and Semiotics, Tel Aviv University, Tel Aviv, Israel, draft. In a reader of a course given by Barbara Partee, at the 1991 Linguistic institute at Santa Cruz, LINS 243
Goldberg, Kálmán, L., and Szábo (eds.): 1992, *Proceedings of the Third Symposium on Logic and Language*, Dordrecht, Revfülöp, Reidel
Goldblatt, R.: 1987, *Logics of Time and Computation*, Lecture Notes, CSLI Publications, Stanford, California
Groenendijk, J., Janssen, T., and Stokhof, M. (eds.): 1981a, *Formal Methods in the Study of Language*, Amsterdam, Mathematical Centre
Groenendijk, J., Janssen, T., and Stokhof, M. (eds.): 1981b, *Truth, Interpretation and Information*, Foris, Dordrecht, partial reprint of Groenendijk et al. 1981a
Groenendijk, J. and Stokhof, M.: 1990, Dynamic Montague Grammar, in Kálmán et al. 1990
Groenendijk, J. and Stokhof, M.: 1991, Dynamic Predicate Logic, *Linguistics and Philosophy* 14, 39–100
Haack, S.: 1974, *Deviant Logic*, Cambridge University Press, Cambridge
Haack, S.: 1978, *Philosphy of Logics*, Cambridge University Press, Cambridge
Haanparanta, L.: 1990, *Language, Knowledge and Intentionality-Perspectives on the Philosophy of Jaakko Hintikka*, Acta Philosphica Fennica, North-Holland, Amsterdam
Harel, D.: 1981, Dynamic Logic, in Gabbay and Guenther 1984
Heim, I.: 1982, *The Semantics of Definite and Indefinite Noun Phrase*, Ph.D. thesis, University of Massachusetts, Amherst
Heim, I.: 1983, File Change Semantics and Familiarity Theory of Definiteness, in Baüerle et al. 1983

Heim, I.: 1990, E-type Anaphora, *Linguistics and Philosophy* 13, 137–177

Hoeksema, J.: 1981, Plurality and Conjunction, in A. ter Meulen (ed.), *Studies in Model-theoretic Semantics*, GRASS Vol. 1, Foris, Dordrecht

Visser, A.: 1995a, *An Axiomatisation of Strong Negation and Relational Composition*, Technical Report 151, Utrecht Research Institute for Philosophy, Utrecht

Janssen, T.: 1983, *Foundations and Applications of Montague Grammar*, Ph.D. thesis, University of Amsterdam

Jaspers, J.: 1994, *Calculi for Constructive Communication*, Dissertation, ITK, Tilburg and ILLC, Amsterdam

Kadmon, N.: 1990, Uniqueness, *Linguistics and Philosophy* 13, 273–324

Kadmon, N. and Landman, F.: 1990, Polarity Sensitive Any and Free Choice Any, in Stokhof and Torenvliet 1990

Kálmán, L. et al. (eds.): 1990, *Proceedings of the Second Symposium on Logic and Language*, Budapest

Kameyama, M.: 1986, A Property Sharing Constraint in Centering, in *Proceedings of the 24th Annual Meeting of the ACL*, pp 200–207

Kamp, H.: 1981, A Theory of Truth and Semantic Representation, in Groenendijk et al. 1981a, pp 277–322

Kamp, H.: 1984, *A Scenic Tour Through the Land of Naked Infinitives*, Unpublised Manuscript

Kamp, H. and Reyle, U.: 1993, *From Discourse to Logic: Introduction to Modeltheoretic Semantics, Natural Language, Formal Logic and Discourse Representation Theory*, Vol. 2, Kluwer Academic Publishers, Dordrecht/Boston/London

Kanazawa, M.: 1993a, *Dynamic Generalized Quantifiers and Monotonicity*, Technical Report LP–93–02, ILLC

Kanazawa, M.: 1993b, Weak vs. Strong readings of Donkey Sentences and Monotonicity Inferences in a Dynamic Setting, *Linguistics and Philosophy* 16

Kanazawa, M. and Piñón, C. J.: 1994, *Dynamics, Polarity and Quantification*, No. 48 in CSLI Lecture Notes, csli, Stanford

Keenan, E. L.: 1987, A Semantic Definition of 'Indefinite NP', in E. Reuland and A. ter Meulen (eds.), *The Representation of (In)definites*, pp 286–317, The M.I.T. Press, Cambridge, Massachusetts

Keenan, E. L.: 1992, Beyond the Frege Boundary, *Linguistics and Philosophy* 15, 199–221, Revised version republished in this collection

Keenan, E. L. and Stavi, J.: 1986, A Semantic Characterization of Natural Language Determiners, *Linguistics and Philosophy* 9, 253–326

Keenan, E. L. and Westerståhl, D.: 1995, Generalized Quantifiers in Linguistics and Logic, in J. van Benthem and A. ter Meulen (eds.), *Handbook of Linguistics and Logic*, Elsevier, Amsterdam

Kibble, R.: 1994, Dynamics of Epistemic Modality and Anaphora, in *International Workshop on Computational semantics*

Kimbal (ed.): 1973, *Syntax and Semantics*, Vol. 2, New York

Krahmer, E.: 1995, *Discourse and Presuposition*, Ph.D. thesis, University of Brabant

Krahmer, E. and Muskens, R.: 1995, *Negation and Disjunction in DRT*, Unpublished Manuscript, Stockholm University. To appear in *Journal of Semantics*

Krifka, M.: 1990, Four Thousand Ships Passed Through the Lock: Object-Induced Measure Functions on Events, *Linguistics and Philosophy* 13, 487–520

Landman, F.: 1989, Groups, 1 & 2, *Linguistics and Philosophy* 12, 559–605 and 723–744

Lang, E.: 1977, *Semantik der Koordinativen Verknüpfung*, No. XIV in Studia Grammatica, Akademie-Verlag, Berlin

Langholm, T.: 1988, *Partiality, Truth, and Persistence*, Lecture Notes, CSLI Publications, Stanford, California

Lappin, S. and Francez, N.: 1994, E-type Pronouns, I-Sums, and Donkey Anaphora, *Linguistics and Philosophy* 17(4), 391–428

Lenstra, J. K., de Bakker, J. W., and Hazewinkel, M. (eds.): 1986a, *Mathematics and Computer Science*, Vol. 1, North-Holland

Lenstra, J. K., de Bakker, J. W., Hazewinkel, M., and Meertens, L. (eds.): 1986b, *Mathematics and Computer Science*, Vol. 2, North-Holland

Lewis, D.: 1975, Adverbs of Quantification, in E. L. Keenan (ed.), *Formal Semantics of Natural Language*, pp 3–15, Cambridge University Press, Cambridge

Link, G.: 1983, The Logical Analysis of Plurals and Mass Terms: A Lattice-theoretic Approach, in Baüerle et al. 1983

Link, G.: 1991, Plural, in Wunderlich and von Stechow 1984

Liu, F.: 1990, *Scope Dependency in English and Chinese*, Ph.D. thesis, UCLA

Liu, F.: 1991, Branching Quantification and Scope Independence, in van der Does and van Eijck 1996

Lønning, J. T.: 1987a, Collective Readings of Definite and Indefinite Noun Phrases, in E. Engdahl and P. Gärdenfors (eds.), *Generalized Quantifiers /Linguistic and Logical Approaches*, Studies in Linguistics and Philosophy, pp 1–29, Reidel, Dordrecht

Lønning, J. T.: 1987b, *Some Aspects of the Logic of Plural Noun Phrases*, dissertation, University of Oslo

Lønning, J. T.: 1995, Collectivity and Plurality, in J. van Benthem and A. ter Meulen (eds.), *Handbook of Linguistics and Logic*, Elsevier, Amsterdam

Muskens, R.: 1989, *Meaning and Partiality*, Ph.D. thesis, University of Amsterdam, Amsterdam

Muskens, R.: 1990, Anaphora and the Logic of Change, in J. van Eijck (ed.), *Jelia '90, European Workshop on logics in AI*, Lecture Notes, pp 414–430, Springer Verlag, Berlin

Muskens, R.: 1995, *Meaning and Partiality*, Studies in Logic, Language and Computation, CSLI Publications, Stanford, CA.

Németi, I.: 1991, *Algebraization of Quantifier Logics: an Introductory Overview*, Technical report, Mathematical Institute, Hungarian Academy of Science, Budapest

Németi, I.: 1993, *Decidability of Weakened Versions of First-Order Logic*, Lecture notes, Fifth European Summer School in Logic, Language and Information, Lisbon, Workshop on Algebraization of Logic

Ojeda, A. E.: 1992, The Semantics of Number in Arabic, in Barker and Dowty 1992, pp 303–325

Partee, B.: 1984, Nominal and Temporal Anaphora, *Linguistics and Philosophy* 7, 243–286

Polanyi, L.: 1985, A Theory of Discourse Structure and Discourse Coherence, in *Papers from the General Session of the Chicago Linguistic Society*, Vol. CLS 21, pp 306–322

Polanyi, L.: 1986, *The Linguistic Discourse Model: Towards a Formal Theory of Discourse Structure*, Technical Report 6409, BBN, Cambridge, Massachusetts

Polanyi, L.: 1988, A Formal Model of the Structure of Discourse, *Journal of Pragmatics* 12, 601–638

Polanyi, L. and Scha, R.: 1983, On the Recursive Structure of Discourse, in Ehlich and van Riemsdijk 1983, pp 141–178

Polanyi, L. and Scha, R.: 1984, A Syntactic Approach to Discourse Semantics, in *Proceedings of COLING*, pp 413–419, Stanford, California

Prüst, H.: 1992, *On Discourse Structuring, VP Anaphora and Gapping*, Ph.D. thesis, University of Amsterdam

Prüst, H. and van den Berg, M. H.: 1991, A Coherent Approach to Underspecification in Natural Language Discourse', in Caenepeel et al. 1991

Prüst, H., van den Berg, M. H., and Scha, R.: 1994, A Formal Discourse Grammar and Verb Phrase Anaphora, *Linguistics and Philosophy* 17, 261–327

Prüst, H. and Scha, R.: 1990, A Discourse Perspective on VP Anaphora, in Stokhof and Torenvliet 1990

Ranta, A.: 1990, Anaphora in Game-theoretical Semantics and in Intuitionistic Type Theory, in Haanparanta 1990

Roberts, C.: 1983, *Modal Subordination, Anaphora and Distributivity*, Ph.D. thesis, University of Massachusetts, Amherst

Scha, R.: 1981, Distributive, Collective and Cumulative Quantification, in Groenendijk et al. 1981a

Scha, R. and Stallard, D.: 1988, Multi-level Plurals and Distributivity, in *Proceedings of the 24th Annual Meeting of the ACL*

Schein, B.: 1993, *Plurals and Events*, The M.I.T. Press

Schwarzschild, R.: 1990, Against Groups, in Stokhof and Torenvliet 1990, pp 475–493

Schwarzschild, R.: 1992a, Together as a Non-distributivity Marker, in Dekker and Stokhof 1991, pp 475–493

Schwarzschild, R.: 1992b, Types of Plural Individuals, *Linguistics and Philosophy* 15, 641–675

Sher, G.: 1990, Ways of Branching Quantifiers, *Linguistics and Philosophy* 13, 393–422

Spaan, M.: 1993, *Parallel Quantification*, Prepublication Series LP–93–01, ILLC

Stokhof, M., Groenendijk, J., and Beaver, D. (eds.): 1991, *Quantification and Anaphora I*, DYANA Edinburgh, deliverable R2.2.A

Stokhof, M. and Torenvliet, L. (eds.): 1990, *Proceedings of the Seventh Amsterdam Colloquium*, Amsterdam, ITLI

Szabolcsi, A. (ed.): 1995, *Ways of Scope Taking*, *forthcoming*, UCLA

Thijsse, E.: 1992, *Partial Logic and Knowledge Representation*, Eburon Publishers, Delft, PhD thesis

Urquhart, A.: 1986, Many Valued Logic, in Gabbay and Guenther 1984

van Lambalgen, M.: 1991, Natural Deduction for Generalized Quantifiers, in J. van der Does and J. van Eijck (eds.), *Generalized Quantifier Theory and Applications*, pp 143–154, Dutch Network for Language, Logic, and Information, Amsterdam, Also to appear with CSLI Lecture Notes.

Veltman, F.: 1990, *Defaults in Update Semantics*, Technical report, ITLI, (refer to ILLC)

Verkuyl, H.: 1993, *a Theory of Aspectuality. The Interaction of Temporal and Atemporal Structure*, Cambridge University Press

Verkuyl, H.: 1994, Distributivity and Collectivity: A Couple of Odds, in Kanazawa and Piñón 1994, pp 49–80

Visser, A.: 1985, Chapter IV.14, Gabbay and Guenther 1984, Vol IV

Visser, A.: 1995a, *Contexts in Dynamic Predicate Logic*, Technical Report 143, Utrecht Research Institute for Philosophy, Utrecht

Visser, A.: 1995b, *Relational Validity and Dynamic Predicate Logic*, Logic Group Preprint series 144, Department of Philosophy, Utrecht Univrsity

Westerståhl, D.: 1984, Some Results on Quantifiers, *Notre Dame Journal of Formal Logik* 25, 152–170

Westerståhl, D.: 1985, Determiners and Context Sets, in van Benthem and ter Meulen 1985

Westerståhl, D.: 1987, Branching Generalized Quantifiers and Natural Language, in P. Gärdenfors (ed.), *Generalized Quantifiers: Linguistic and Logical Approaches*, pp 109–150, D. Reidel Publishing Company, Dordrecht

Westerståhl, D.: 1989, Quantifiers in Formal and Natural Language, in Gabbay and Guenther 1984, pp 1–131

Westerståhl, D.: 1995, Quantifiers in Natural Language: A survey of some recent work, in M. Krynicki and M. Mostowski (eds.), *Quantifiers*, D. Reidel Publishing Company, Dordrecht

Winter, Y.: 1995a, *Choice Functions and the Scopal Semantics of Indefinites*, manuscript, OTS, Utrecht

Winter, Y.: 1995b, *A Unified Semanitic Treatment of Singular NP Coordination*, manuscript, OTS, Utrecht, to appear in Linguistics and Philosophy

Wunderlich, F. and von Stechow, A. (eds.): 1984, *Semantics, An International Handbook of Contemporary Research*, de Gruyter, Berlin and New York

Zeevat, H.: 1989, A Compositional Approach to Discourse Representation Theory, *Linguistics and Philosophy* 12, 95–131, (also in: Zeevat 1991)

Zeevat, H.: 1991, *Aspects of Discourse Semantics and Unification Grammar*, Ph.D. thesis, University of Amsterdam

Samenvatting

Het onderwerp van dit proefschrift is het vinden van een beschrijving in logische formules van de betekenis van teksten. In teksten hangt de betekenis van een groot aantal uitdrukkingen samen met de betekenis van eerdere uitdrukkingen in die tekst. Om dit formeel te beschrijven zijn er in de laatste 15 jaar een aantal theorieën ontwikkeld die de betekenis van zinnen uitdrukt als een vermogen om informatie-toestanden te veranderen. De betekenis van een zin bestaat uit een uitdrukking die oude informatie oppikt en gebruikt, en nieuwe informatie introduceert. Dit zijn de zogenaamde dynamische betekenis theorieën. Als een simpel voorbeeld, in de zinnen

(2) *Een man kwam de bar binnen. Hij bestelde een biertje.*

introduceert de eerste zin de informatie dat er een man is die de bar binnen komt, en de tweede zin gebruikt die informatie om de uitdrukking *hij* een waarde te geven. Op deze manier kan informatie van een zin naar volgende zinnen. worden doorgegeven. Let op: de informatie hoeft niet uniek te zijn, er kan meer dan een mogelijke man zijn.

In dit proefschrift ga ik uit van een theorie ontwikkeld door Groenendijk en Stokhof: dynamische predicaten logica (DPL). In deze logica wordt een formule geïnterpreteerd als een relatie tussen een input toestand (de oude informatie) en een output toestand (de nieuwe informatie). Een formule verandert oude informatie in nieuwe. Het is een relatie, geen functie, omdat er geen unieke nieuwe toestand hoeft te zijn. De relatie is *waar* voor een input en output, als de formule inderdaad de input in de output kan veranderen. In voorbeeld (2), is de formule corresponderende met de eerste zin waar voor een input en output als de output hetzelfde is als de input, met als uitzondering dat de output nu ook informatie over de man bevat. Als de input en output niet de juiste relatie met elkaar hebben, is de relatie *onwaar*.

In hoofdstuk 2 stel ik een aantal verandering van deze theorie voor. De belangrijkste verandering bestaat uit een overgang naar drie-waardige logica. Waar in DPL uitdrukkingen alleen *waar* of *onwaar* kunnen zijn, kunnen ze daarnaast in deze nieuwe theorie ook *irrelevant* zijn. Dit is nodig voor het volgende

(3) *Een vrouw kwam de bar binnen. Toen kwam een man de bar binnen. Hij bestelde een biertje voor haar.*

Neem de tweede zin. De input van die zin zal informatie bevatten over de vrouw, de output zal extra informatie bevatten over de man. Noem de input van de tweede zin I en de output O. de relatie die met de tweede zin correspondeert zal waar zijn voor I en O als O hetzelfde is als I, behalve dat nu ook de man is geïntroduceerd. Dat ze voor de rest hetzelfde zijn betekent onder andere dat de vrouw nog steeds bekend is. Belangrijk, omdat *haar* in de laatste zin die informatie nodig heeft om geïnterpreteerd te worden.

We hebben gezien dat in DPL de relatie die correspondeert met een formule waar is als de formule O uit I kan maken door er informatie aan toe te voegen, en onwaar als O niet op die manier gemaakt kan worden. Vergelijk nu de volgende twee redenen om een onware waarde te krijgen voor de tweede zin: (1) I en O zijn hetzelfde, behalve dat O een man introduceert die niet de bar binnen kom; (2) O bevat geen informatie meer over de vrouw. In beide gevallen kan O niet uit I gemaakt worden door de formule. In het eerste geval is de onware waarde heel redelijk. In het tweede geval is dat veel minder voor de hand liggend. De output wordt verandert op een manier die niets met de zin te maken heeft. Een dergelijke output is irrelevant.

Het blijkt dat een dergelijke drie-waardige interpretatie het gedrag van formules onder negatie sterk verbetert: als ik *niet* voor een ware formule zet is het resultaat onwaar, voor een onware formule geeft waar, maar voor een irrelevante formule blijft irrelevant.

Het hoofdstuk eindigt met een aantal wiskundige overwegingen, waaruit blijkt dat ook vanuit een formeel perspectief de gedefiniëerde logica een "natuurlijke" is.

In hoofdstuk 3 bespreek ik een ander onderwerp: het onderscheid tussen enkelvoud en meervoud en de wijze waarop taaluitdrukkingen daarvan gebruik maken. ik geef hier een herformulering van wat de standaard theorie voor meervoud en quantificatie genoemd kan worden. De belangrijkste referentie hier is het proefschrift van Jaap van der Does uit 1992. Mijn formulering resulteert in een bijna identieke theorie, en heet als voordeel dat de vorm van de uitdrukkingen hetzelfde is als de dynamische uitdrukkingen van hoofdstuk 2, wat samenvoegen van de twee theorieën vergemakkelijkt.

In het tweede deel van dit hoofdstuk bespreek ik een variant waarin het mogelijk is om niet alleen over objecten te praten die uit meer dan een element bestaan, maar ook relaties tussen elementen van die meervoudige objecten kunt uitdrukken. Dit klinkt ingewikkelder dan het is. Neem de volgende zin:

(4) *Elke vrouw houdt van een man.*

In deze zin is er niet alleen sprake van de verzameling vrouwen, en van de verzameling mannen van wie ze houden (twee meervoudige objecten), maar er is ook sprake van een relatie tussen de individuele vrouwen en de man van wie ze houden. De variant van meervoudslogica die ik invoer kan deze interne verbanden uitdrukken. Enigzins verrassend is, dat de nieuwe logica geen extra uitdrukkingskracht geeft: ze is equivalent met de meer standaard logica in het eerste deel van het hoofdstuk. Pas als we in het volgende hoofdstuk overgaan op een dynamische logica, is er sprake van verschil.

In hoofdstuk 4 combineer ik de resultaten van hoofdstukken 2 en 3.

In hoofdstuk 5 pas ik de theorie toe. Eerst laat ik zien dat de voorspellingen van eerdere theorieën, die alleen over enkelvoud gaan, behouden blijven. Daarna bespreek ik een aantal gevallen die met meervoud te maken hebben.

Veel van de toepassingen worden hier alleen geschetst. Een preciese uitwerking moet worden uitgesteld tot een latere datum. De combinatie met discourse grammatica, de theorie die de structuur van natuurlijke taal teksten beschrijft, zal in een apart artikel beschreven worden (In de proceedings van het tiende Amsterdams Colloquium).

Titles in the ILLC Dissertation Series

Transsentential Meditations; Ups and downs in dynamic semantics
Paul Dekker
ILLC Dissertation series 1993-01

Resource Bounded Reductions
Harry Buhrman
ILLC Dissertation series 1993-02

Efficient Metamathematics
Rineke Verbrugge
ILLC Dissertation series 1993-03

Extending Modal Logic
Maarten de Rijke
ILLC Dissertation series 1993-04

Studied Flexibility
Herman Hendriks
ILLC Dissertation series 1993-05

Aspects of Algorithms and Complexity
John Tromp
ILLC Dissertation series 1993-06

The Noble Art of Linear Decorating
Harold Schellinx
ILLC Dissertation series 1994-01

Generating Uniform User-Interfaces for Interactive Programming Environments
Jan Willem Cornelis Koorn
ILLC Dissertation series 1994-02

Process Theory and Equation Solving
Nicoline Johanna Drost
ILLC Dissertation series 1994-03

Calculi for Constructive Communication, a Study of the Dynamics of Partial States
Jan Jaspars
ILLC Dissertation series 1994-04

Executable Language Definitions, Case Studies and Origin Tracking Techniques
Arie van Deursen
ILLC Dissertation series 1994-05

Chapters on Bounded Arithmetic & on Provability Logic
Domenico Zambella
ILLC Dissertation series 1994-06

Adventures in Diagonalizable Algebras
V. Yu. Shavrukov
ILLC Dissertation series 1994-07

Learnable Classes of Categorial Grammars
Makoto Kanazawa
ILLC Dissertation series 1994-08

Clocks, Trees and Stars in Process Theory
Wan Fokkink
ILLC Dissertation series 1994-09

Logics for Agents with Bounded Rationality
Zhisheng Huang
ILLC Dissertation series 1994-10

On Modular Algebraic Protocol Specification
Jacob Brunekreef
ILLC Dissertation series 1995-01

Investigating Bounded Contraction
Andreja Prijatelj
ILLC Dissertation series 1995-02

Algebraic Relativization and Arrow Logic
Maarten Marx
ILLC Dissertation series 1995-03

Study on the Formal Semantics of Pictures
Dejuan Wang
ILLC Dissertation series 1995-04

Generation of Program Analysis Tools
Frank Tip
ILLC Dissertation series 1995-05

Verification Techniques for Elementary Data Types and Retransmission Protocols
Jos van Wamel
ILLC Dissertation series 1995-06

Transformation and Analysis of (Constraint) Logic Programs
Sandro Etalle
ILLC Dissertation series 1995-07

Frames and Labels. A Modal Analysis of Categorial Inference
Natasha Kurtonina
ILLC Dissertation series 1995-08

Tools for PSF
G.J. Veltink
ILLC Dissertation series 1995-09

Studies in Dynamic Logic
Giovanna Cepparello
ILLC Dissertation series 1995-10

Instantial Logic. An Investigation into Reasoning with Instances
W.P.M. Meyer Viol
ILLC Dissertation series 1995-11

Taming Logics
Szabolcs Mikulás
ILLC Dissertation series 1995-12

Meta-Logics for Logic Programming
Marianne Kalsbeek
ILLC Dissertation series 1995-13

Enriching Linguistics with Statistics: Performance Models of Natural Language
Rens Bod
ILLC Dissertation series 1995-14

Computational Pitfalls in Tractable Grammatical Formalisms
Marten Trautwein
ILLC Dissertation series 1995-15

The Solution Sets of Local Search Problems
Sophie Fischer
ILLC Dissertation series 1995-16

Contexts of Metaphor
Michiel Leezenberg
ILLC Dissertation series 1995-17

Logical Investigations into Dynamic Semantics
Willem Groeneveld
ILLC Dissertation series 1995-18

Investigations in Logic, Language and Computation
Erik Aarts
ILLC Dissertation series 1995-19

Modal Quantifiers
Natasha Alechina
ILLC Dissertation series 1995-20

Computations in Propositional Logic
Lex Hendriks
ILLC Dissertation series 1996-01

Categories for Profit
Erik de Haas
ILLC Dissertation series 1996-02

Some Aspects of the Internal Structure of Discourse: the Dynamics of Nominal Anaphora
Martin H. van den Berg
ILLC Dissertation series 1996-03