TRANSSENTENTIAL MEDITATIONS
Ups and downs in dynamic semantics

TRANSSENTENTIAL MEDITATIONS
Ups and downs in dynamic semantics

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
in het openbaar te verdedigen in de Aula der Universiteit
(Oude Lutherse Kerk, ingang Singel 411, hoek Spui),
op donderdag 6 mei 1993 te 10.30 uur

door

Paul Jacques Edgar Dekker
geboren te Zwijndrecht

Promotor: Prof. Dr. R.I. Bartsch
Faculteit der Wijsbegeerte
Co-promotor: Dr. J.A.G. Groenendijk
Faculteit der Wijsbegeerte

Printed by Febodruk, Enschede

Acknowledgements

Writing this dissertation would not have been possible without the support, encouragement, and, most of all, expertise of my teachers Renate Bartsch, Jeroen Groenendijk and Martin Stokhof. They not only offered an excellent education and caused substantial improvements in this work, they also learned me the joy of an enterprise like the one reported here. I am deeply indebted to them.

Very special thanks go to my tutor and pal Herman Hendriks, with whom I experienced the ups and downs, and who showed me some of the tops and bottoms. Furthermore, I would like to thank Frank Veltman, Gennaro Chierchia, David Beaver and Martin van den Berg, for the stimulating and instructive discussions we had.

For reasons not having to do with the contents of this dissertation, I would like to thank –the order is arbitrary– Marjorie, Ria, Karel and Berrie. Of my friends, I would like to thank, in particular, Harry, Rob, Emiel and Lennart.

Of course I am also grateful for the grants I received. The work was financially supported by the University of Amsterdam, the Netherlans Organization for Scientific Research (NWO) and the ESPRIT Basic Research Project Dynamic Interpretation of Natural Language (DYANA).

This dissertation is dedicated to my mother and to my father. Their contribution to this work is as irretracable as significant. I also thank my sister Marieke and my brother and personal zwifzwaf Kees. I very much appreciated the new mail from my sister alerts. Typing errors may be attributed to them.

Finally I want to thank my best friends Arien and Romanie. I think I am indebted to you two most of all for the delight found in your company, and the talks not about this thesis.

Contents

Chapter 1

Introduction

This thesis deals with several aspects of the dynamic interpretation of natural language, a topic that, within the area of formal semantics, has become an established issue the last five years or so. With this work I want to contribute to the transformation of referential theories of meaning into a dynamic, information based theory of interpretation.

Background and scope

It may be useful to start with stressing the referential (truth-oriented, model-theoretic) roots of the present work. In referential theories, meaning is explained in terms of reference (or denotation) relations which hold between linguistic expressions and independent entities. Key notions are those of reference and truth, and the primary goal of an adequate semantic theory is taken to be that of presenting a theory of truth, or, rather, the characterization of the truth conditions of (indicative) sentences. (With regards to imperatives and interrogatives truth and entailment conditions are inappropriate, of course, and will have to be replaced by, for instance, fulfilment and answerhood conditions.)

Referential theories of meaning have been regarded as opposed to mentalistic theories of meaning. In the mentalistic view, meaning is primarily related to the mental representations that accompany linguistic expressions. The meanings of expressions are conceived to be, first and foremost, the concepts and thoughts language users associate with the expressions. The basic difference between the two types of theories is that in the former meaning relates language to the outside world, and in the latter to 'the mental'.

In the area of model-theoretic semantics, the past ten years witness the emergence of dynamic theories of meaning. The focus of attention shifts to the way in which (expressions in) one sentence may change the context of interpretation of (expressions in) other sentences. With this dynamic twist the perspective upon meaning changes. Contexts of interpretation are viewed as information states, and the goal of semantics is taken to be that of giving a characterization of the changes (utterances of) sentences bring about in information states. Thus, the theory of meaning evolves into a theory of information, of information update and information exchange.

This development not only exhibits the evolution of a more procedural outlook upon meaning and interpretation, it also involves a move towards more mentalistic, or representationalistic theories of meaning. The latter move is most evident in Hans Kamp's *discourse representation theory* where the meanings of sentences are primarily defined in terms of the changes they bring about in so-called discourse representation structures. In this theory truth is a notion which is not directly related to sentences, but to discourse representation structures.

However, subsequent work of Groenendijk and Stokhof shows that the dynamic twist in semantics need not necessarily involve such a departure from the more referential paradigm. It is shown that empirical results of discourse representation theory can be obtained without employing intermediary representations. The upgrade of the static referential notion of meaning into a dynamic referential one appears to be sufficient for an account of intersentential (anaphoric) links in discourse.

In the different versions of dynamic semantics which come up in this thesis, the referential view remains the point of departure and return. With Groenendijk and Stokhof's systems of *dynamic predicate logic* and *dynamic Montague grammar*, the dynamics of interpretation remains confined to the enabling and disabling of anaphoric coreference. The contents of sentences remain associated with truth conditions. Also with Veltman's *update semantics* the notion of information that is changed and exchanged remains, basically, information about the world. Information is characterized in terms of what are considered to be intersubjective conditions on what the world is like, i.e., in fact, in terms of truth conditions.

One of the basic assumptions adopted is that language users are able to exchange information on the basis of acknowledged conventional links obtaining between descriptive lexical expressions and entities in the world 'out there'. Roughly speaking, what is considered to be dynamic is the use that is made of these expressions, with their static denotations, in situations of information exchange.

This thesis belongs to the tradition in semantics in which natural language meaning is approached from a formal perspective. Mathematical tools are employed in order to provide precise definitions of truth, entailment, information and the like. One of the key notions in such formal approaches is that of the principle of compositionality of meaning. This principle is, if it can not be literally attributed to Frege, at least inspired by Frege, and can be stated in the following way:

The meaning of a compound expression is built up from the meanings of its parts.

Although some have understood this principle as an empirical statement about (natural language) semantics, it is better conceived of as a principle imposing restrictions on the description and formalization of the syntax and semantics of a language. Quoting from Janssen [1986, p. 39], which devotes much attention to the principle and to the formal statement of it, "(...) the principle of compositionality is not a principle of languages. It is a principle concerning the organization of grammars dealing both with syntax and semantics."

Janssen addresses several arguments for adopting the principle of compositionality, among which we find the benefit of working within a mathematically precise framework, formal elegance and power of a compositional framework, general applicability, heuristic value of the principle, and the fact that it enables a finite statement of the interpretation of an infinite number of expressions. I would like to add the observation that once one has gone through a thorough training in compositional semantics like the one I have experienced, one can hardly conceive of a non-compositional semantics anymore. Of course, for some this observation may constitute an argument *against* adopting the principle, but then it serves, at least, to explain the role which the principle of compositionality plays in the proposals put forward in this book.

The structure of this work

The first chapter of this thesis provides a general introduction to two frameworks in the area of natural language semantics which have been proposed as an alternative for and modification of classical Montague grammar: discourse representation theory and dynamic Montague grammar.

Discourse representation theory and dynamic Montague grammar differ from traditional Montague grammar in that they give a formal account of the dynamics of natural language interpretation. The dynamics is concerned with the phenomenon that (expressions in) one sentence change the context of interpretation of (expressions in) another sentence. Dynamic Montague grammar is the basic framework upon which the next three chapters elaborate. For expository reasons and historical adequacy the exposition of the system of dynamic Montague grammar is preceded by a sketch of the framework of discourse representation theory and of the system of dynamic predicate logic.

The second chapter presents an extended dynamic system of interpretation. The system of dynamic Montague grammar deals with the potential of sentence meanings to 'pass on' possible antecedents for subsequent pronouns. This potential is accounted for, basically, by the adoption of dynamic interpretation of two of the three basic sentential operators: that of conjunction and that of existential quantification. In the second chapter also the remaining sentential operator, that of negation, is given a dynamic interpretation. First it is argued that a dynamic interpretation of this operator is needed. There are certain anaphoric relationships, somewhat extraordinary but systematically interrelated, that require a dynamic notion of negation, i.e., one that also allows antecedents for pronominal coreference in the scope of the negation operator to be passed on. Next an alternative dynamic Montague grammar is developed which has dynamic notions of conjunction, existential quantification and negation and which is shown to cover the data discussed.

The third chapter can be conceived of as an exercise in 'statification'. In this chapter it is shown that the results obtained in dynamic Montague grammar and in its extended version presented in chapter 2 can also be obtained from a very simple static Montague grammar if the latter is extended with a system of type change. Moreover, it is shown that such a system of type change allows one to deal with much more complicated and puzzling structures in discourse, and this in a compositional fashion. However, the conclusion of the chapter is somewhat reserved. In view of controlling the overgeneration that comes with the system of type change, it is finally argued that a basic dynamic Montague grammar, supplemented with a properly restricted system of type change, might give the right results in a more manageable fashion.

Chapter 4 again elaborates upon the original dynamic Montague grammar of Groenendijk and Stokhof. In this chapter it is shown that the dynamic view on meaning can be successfully applied in a treatment of so-called implicit arguments. More specifically, it is shown how a more procedural approach can be employed in a treatment of relational nouns, extensional adverbs and tense in texts.

The final chapter 5 is concerned with more foundational notions in dynamic semantics. The chapter sets out to give more intuitive content to the more or less technical notion of information about the values of variables as it is employed in dynamic predicate logic and dynamic Montague grammar. Dynamic predicate logic is formulated into an update logic, in order to pave the way for the development of more general systems of information update and information exchange. In this chapter it is shown, furthermore, that the resulting update logic allows for an elegant treatment of several kinds of quantifying adverbs and adnominal quantification. The chapter concludes with a short study of the notion of a partial object.

Prerequisites

The main topic in this book is the formulation of the dynamics of natural language interpretation within a formal semantic framework. Although the further aim is to contribute to a formal theory of natural language interpretation, the central occupation of the book is to find the logical/formal tools by means of which certain generalizations can be accounted for appropriately. Substantial parts of this book will therefore be of a rather technical nature. And although I will try as much as possible to motivate and explain the more technical parts, they may be difficult for the less formally oriented reader. For this reason it seems appropriate to give an indication of the presupposed skills of the reader.

It would be very convenient for the reader of this thesis to be familiar with Montague grammar. (Excellent introductions can be found in Dowty, Wall and Peters [1981] and Gamut [1991].) A basic working knowledge of (extensional) type theory is a prerequisite for understanding the first four chapters of this thesis. (Again, Gamut [1991] provides a good introduction of the basics of extensional type theory.) Acquaintance with set-theoretical notation is presupposed in the fifth chapter.

Chapter 2 presupposes that the reader is acquainted with the system of dynamic Montague grammar presented in the third section of this introductory chapter. The third chapter builds on the first two chapters. The chapters 4 and 5 can be read independently.

If one is not sure about meeting the sketched 'requirements', this introductory chapter may serve as a test case. If the reader survives this chapter, he is certainly able to survive the remainder of the book.

1 DRT and DPL

Montague grammar constitutes the paradigmatic framework for the formal approach to the semantics of natural language in the seventies. The various fragments (Montague [1970a, 1970b] and, in particular, the so-called *PTQ* fragment in [1973]) are developed by the logician Richard Montague against the background of a universal syntax and semantics within an algebraic framework, and "to comprehend the syntax and semantics of both languages [natural and artifical, PD] within a single natural and mathematically precise theory" ([1973]). The original 'static' Montagovian paradigm has some principled limitations, however, which can be overcome by adopting a more procedural view on meaning, inspired by developments in cognitive science. Such a more dynamic view can be found in Hans Kamp's discourse representation theory and Irene Heim's file change semantics, and these systems turn out to be major competitors of the Montagovian framework in the eighties. More recently, several compositional reformulations of discourse representation theory have been given which show that discourse representational results can be obtained within a Montagovian framework after all. In particular, Groenendijk and Stokhof [1990a] presents a dynamic Montague grammar which adapts classical Montague grammar in order to incorporate these results.

In this section I will give an outline of discourse representation theory (*DRT*), and of Groenendijk and Stokhof's system of dynamic predicate logic. The latter can be conceived of as a reformulation of *DRT* which figures as a first step towards the construction of a system that gives a completely compositional treatment of *DRT* results. In the third section of this chapter the fully compositional system of dynamic Montague grammar is presented.

1.1 Discourse representation theory

The term discourse representation theory (*DRT*) is associated with the theory proposed by Hans Kamp [1981] in the early eighties and the branch of theorizing that originated from that paper. However, the term has also been applied to more or less similar proposals independently developed by Irene Heim (file change semantics, [1982, 1983]) and Pieter Seuren (discourse semantics, [1985]). The following exposition of *DRT* will be based mainly on Kamp's work, but can be understood as a characterization of (some aspects of) the other theories as well.

One of the motivations for the development of *DRT* is to provide for a framework in which pronominal anaphora can be dealt with appropriately, and I will concentrate on this aspect in the sketch of *DRT* below. It may be pointed out, however, that the coverage of *DRT* is more comprehensive. The theory has also been used, with considerable success, to account for other intersentential semantic relationships, tense and aspect, and propositional attitude reports. Furthermore, the original paper by Kamp [1981] has the more general aim of bridging the gap between two dominating conceptions of meaning, the one in which meaning is primarily related to truth and reality, and the one in which it is related to the mental representations of language users.

In this section I will first sketch the problems with anaphoric relationships that arise in Montague grammar and next the representational solution to these problems that *DRT* offers. Section 2.2 sketches Groenendijk and Stokhof's system of dynamic predicate logic which gives a compositional, non-representational reformulation of *DRT* that accounts for the same phenomena.

Donkey sentences

One of the shortcomings of the *PTQ* fragment is that it only deals with isolated sentences and cannot in general interpret anaphoric pronouns correctly. More in particular, *PTQ* fails to give a proper account of certain anaphoric relationships between pronouns and indefinite noun phrases, both within and across sentence boundaries. The kind of problems with anaphoric relationships that the *PTQ* model encounters can be illustrated by the following, paradigmatic, examples:

(1) A man walks in the park. He whistles.
(2) If a farmer owns a donkey, he beats it.
(3) Every farmer who owns a donkey beats it.

In these examples we find anaphoric relationships between anaphoric pronouns in one sentential clause and indefinite noun phrases in another. The anaphoric relationship in the first example might be accounted for by Montague quantification rules, but at a high price. Such an account would require the sequence of sentences to be obtained, syntactically, by an application of the quantification rule to the noun phrase *a man* and the structure *He walks in the park. He whistles*. This, however, would imply that the meaning of example 1 is not derived from that of its constituent sentence *A man walks in the park*. The noun phrase *a man* would only be taken to contribute, semantically, to the interpretation of the whole of example 1, not to its first sentence.

This way of proceeding is at odds with the intuition that sentences have an interpretation of their own, and that the interpretation of a discourse involves the incremental, one by one, processing of the sentences in the discourse. But not only is this approach counterintuitive, it also does not work for all noun phrases, and it is misguided when we turn to the examples 2 and 3.

In the *PTQ* model the anaphoric relationships in example 2 can be dealt with by quantifying the noun phrases *a farmer* and *a donkey* in the expression *If he owns it he beats it*. The translation of example 2 that results will have as truth conditions that there exist a farmer and a donkey such that if the farmer owns the donkey, then the farmer beats the donkey. In other words, the sentence would already be verified by the existence of a farmer and a donkey which is not owned by the farmer.

Intuitively, however, the sentence is associated with the stronger statement that in any case in which a farmer owns a donkey, the farmer beats the donkey, i.e., that every farmer beats every donkey he owns.[1] The problem for *PTQ* is how to get, in a non-ad hoc way, such a universal interpretation of the noun phrases *a farmer* and *a donkey*, which are normally interpreted in an existential way.

Similarly, the indefinite *a donkey* in example 3 is preferably read with universal force. Normally, the sentence is also taken to state that every farmer beats *every* donkey he owns. Here, too, the problem is that of finding a semantic way of dealing with the indefinite terms which accounts for the fact that in one construction they have existential import, and in another construction they have universal import.

Discourse referents

Already back in 1968, Karttunen [1968a] introduced the notion of a discourse referent in order to account for anaphoric relationships. Discourse referents are neither individuals in the mind of the speaker, nor 'real' objects in the world, but they are some kind of entities which can be introduced by indefinite noun phrases in the domain of discourse and which can be referred to by means of pronouns or revived by definite descriptions. Sometimes, an indefinite introduces a discourse referent

1. Other interpretations of examples like 2 have been proposed. These will be disregarded for the moment, but they will be discussed in chapter 5.

permanently, for instance, if the indefinite is not 'flagged' by a modal or a certain non-factive verb. Sometimes, the discourse marker has a restricted 'life expectancy'.

The following example illustrates both types of life-styles of discourse referents:

(4) A girl dreamt of being married to a rich man and living with him for the rest of her life.

(4a) She always dreamt of matrimony.

(4b) *She never quarreled with him.

The indefinite *a girl* in example 4 introduces a permanent discourse referent in the discourse, which appears from the fact that the girl can be referred back to by the pronoun in the continuation with 4a. On the other hand, if the indefinite *a rich man* is not read specifically, it only introduces a discourse referent for the interpretation of *living with him for the rest of her life*. Clearly, in the context of such a girl's dreams, viz., a context in which she is married to a rich man, there is such a rich man which can be referred back to by the pronoun *him* in that clause. However, out of that context there need not be such a man, and the desires of the girl may always remain what they are, only dreams. Consequently, with a continuation of 4 with example 4b we are unable to refer back to such a man, that is, if we still assume the non-specific interpretation of that indefinite.[2]

Discourse representation theory can be regarded as a formal elaboration of Karttunen's ideas about discourse referents, an elaboration that at the same time accounts for the universal impact of indefinites in certain contexts. In *DRT*, discourse referents (or, as we will call them in the sequel, discourse markers) figure in so-called discourse representations which are built up in the processing of discourse. They mark definite and indefinite objects mentioned in the discourse and are as it were a stand in for real objects which satisfy the conditions associated with them. Discourse markers are introduced in these representations by, among others, indefinite noun phrases, and, since they are definite objects, they can be picked up by pronouns in subsequent discourse. In general it depends on the specific configuration of these representations whether the discourse markers are a stand in for some indefinite object that satisfies certain imposed conditions, or for *all* objects that do, and thus both the existential and the universal impact of indefinites is accounted for.

DRT architecture

The main characteristics of *DRT* can be deduced from its name. *DRT* focuses on the semantic interpretation of *discourse*, i.e., on sequences of sentences, and in the

2. However, as Karttunen [1968a] already noticed, the context of the girl's dreams can be 'revived' by subsequent modal operators and pronouns in the scope of such operators can happily refer to the man dreamt of. So, for instance, after uttering example 4 we can properly continue with the sentence *She would never quarrel with him*. Cf., also, Seuren [1985, Ch. 5] and Heim [1992].

process of interpretation an intermediate level of *representations* is employed. In these two respects *DRT* differs fundamentally from Montague grammar. The latter focusses on the semantic interpretation of isolated sentences. Furthermore, although in Montague's *PTQ* model also an intermediary translation language is employed, the intermediary language can in principle be dispensed with[3], whereas the level of discourse representation in *DRT* is considered to be an essential component of the grammar in between the levels of syntactic analysis and semantic interpretation.

The idea underlying the level of discourse representation in *DRT* is that discourse representations reflect the information conveyed by a discourse in the form of a partial model of reality. The interpretation of a discourse involves the incremental, sentence by sentence, construction of the representation of the discourse. Such a representation reflects the contents of the discourse and it furthermore figures as a context of interpretation for successive parts of discourse. So, the meaning of some constituent sentence S of a discourse is taken to consist in the contribution that S makes to the discourse representation which is constructed on the basis of the sentences that precede S. Typically, (sentences containing) indefinite descriptions introduce discourse markers at this intermediary level of discourse representation, and anaphoric relationships between pronominal anaphors and indefinites are established by associating the pronouns with the discourse markers introduced by the indefinites.

In their turn, discourse representations are interpreted in a more familiar fashion by means of a recursive model-theoretic truth definition for discourse representation. The truth of a discourse is defined in terms of the truth of the discourse representation that results from interpreting the discourse.

A full statement of *DRT* comprises the following three parts:

1. a syntax for a fragment of English, including a rule of discourse formation
2. a definition of the way in which discourse representations can be derived from syntactically analyzed (sequences of) sentences
3. a truth definition for discourse representations

In this informal sketch we will disregard the syntax of a *DRT*-fragment. The derivation of the discourse representation for a given sentence is governed by so-called *DRS*-construction rules. The precise statements of these rules will also largely be left implicit. (Cf., Kamp and Reyle [1993] for a full specification of such rules.) We will for the most part consider fully analyzed discourse representation structures constructed on the basis of some sample sentences.

3. Since the *PTQ* translation process and the interpretation of the language of translation both are defined in a compositional fashion, the indirect interpretation of the fragment of natural language can be replaced by a direct interpretation, viz., by the composition of the two processes. Actually, Montague adopted such a method of direct interpretation in his 'English as a formal language' [1970a].

DRS-construction

By means of the *DRS*-construction algorithm natural language discourse is converted into discourse representation structures (henceforth, *DRSs*), which give a formal representation of the content of the discourse. These *DRSs* consist of two parts: a set of discourse markers, called the universe of a *DRS*, and a set of (occurrences of) conditions. Intuitively, the universe of a *DRS* consists of the individuals introduced in the course of a discourse, and the conditions are the ones imposed on the values of these discourse markers in the discourse. A formal definition of a *DRS* is presented below.

Given a discourse D consisting of the sequence of sentences $S_1, S_2, \ldots S_n$, the construction algorithm first constructs a *DRS* K_1 for the first sentence S_1. With respect to K_1 the second sentence S_2 is processed, the result of which is an extended *DRS* K_2, and proceeding in this way the last sentence S_n will finally be analyzed with respect to the *DRS* K_{n-1} for the sequence of sentences $S_1, S_2, \ldots S_{n-1}$. The result of this is some *DRS* K_n which is the discourse representation for the discourse D.

Normally the construction algorithm works top-down on (syntactically analyzed) sentences. In order to give an idea of how the algorithm deals with sentences, consider the construction of a *DRS* for the following example with respect to some initial *DRS* K_i:

(5) John loves a girl who hates him.

When example 5 is processed, the *DRS*-construction algorithm dictates that first a 'new' discourse marker y is added to the universe of K_i, that is, a discourse marker that has not already been used in K_i. Next the condition 'y is John' is added to the conditions in K_i, and the algorithm proceeds with the phrase *y loves a girl who hates him*. The processing of this phrase involves the introduction of another new discourse marker z, and the conditions 'z is a girl' and 'y loves z'. Finally, the construction algorithm processes the clause *z hates him*. For the pronoun a 'suitable' member of the universe of the present *DRS* has to be chosen. The discourse marker z is disqualified because it is associated with the wrong gender (and also because the pronoun would have to be reflexive then). So, it will be either y, the discourse marker associated with *John*, or some discourse marker already introduced by previous discourse into the universe of K_i. If some such discourse marker x has been chosen, where either x is y or x is some other discourse marker introduced before, the condition 'z hates x' is added. Summing up, the processing of the sentence with respect to some initial *DRS* K_i involves the addition to K_i of two 'new' discourse markers y and z to the universe of K_i, and the addition of the conditions 'y is John', 'z is a girl', 'y loves z' and 'z hates x', where x is y or some other 'familiar' discourse marker.

A few remarks may be added about the 'accessibility' hierarchy, which determines which discourse markers may be used for the interpretation of pronouns. As we will see presently, *DRSs* may occur as constituents of conditions in other *DRSs*. For instance, *DRSs* may contain conditions which consist in the negation of other *DRSs*, or implicational conditions which connect an antecedent with a consequent *DRS*. In case a *DRS* K_i contains the negation of a *DRS* K_j, then K_j is said to be subordinate to K_i and to all other *DRSs* to which K_i is subordinate. And if an implicational condition with antecedent *DRS* K_j and antecedent *DRS* K_k occurs in K_i, then K_j is subordinate to K_i, K_k is subordinate to K_j and K_i, and K_j and K_k are subordinate to all *DRSs* to which K_i is subordinate.

This subordination relation is relevant for the resolution of pronouns, or, rather, the choice of discourse markers for the interpretation of pronouns. Disregarding gender and number, if a pronoun is interpreted with respect to some *DRS* K_i, then it has to be associated with a discourse marker in the universe of K_i or in the universe of any *DRS* to which K_i is subordinate. For instance, consider again example 2, *If a farmer owns a donkey, he beats it*. The evaluation of this conditional sentence introduces in the main *DRS*, the *DRS* of evaluation, an implicational condition, where the antecedent *DRS* is generated from the interpretation of the clause *a farmer owns a donkey*, and the consequent from the interpretation of the clause *he beats it*. When interpreting the pronouns *he* and *it* relative to the consequent *DRS*, they have be associated with discourse markers introduced in that *DRS*, or in a superordinate *DRS*. In the present example they must be discourse markers introduced in the antecedent *DRS* by the indefinite noun phrases *a farmer* and *a donkey*, or discourse markers which have already been introduced in the main *DRS*.

Furthermore, if example 2 has been processed with respect to some *DRS* K_i, the result of which is some *DRS* K_j which consists of K_i with the added implicational condition, then the discourse markers introduced by the indefinites *a farmer* and *a donkey* are not accessible anymore for subsequent anaphoric reference. The reason is that subsequent sentences are evaluated with respect to K_j, and that pronouns can not be associated with discourse markers in the universe of a *DRS* subordinate to the *DRS* of evaluation. Notice that, thus, the subordination relation is used to characterize the different life expectancies of discourse referents.

Discourse representation structures

In the original proposal by Kamp, *DRSs* are some kind of mixtures of natural language expressions with ingredients of a logical language. They are set-theoretical constructs built up from the lexical expressions of a natural language and discourse markers of the intermediary *DRS* language.

For the sake of convenience, conditions like 'y is John', 'z is a woman', 'y loves z' and 'z hates x' can be formulated as first order predicate logic atomic formulas

$y = j$, Wz, Lyz and Hzx, respectively. Employing such a notation, we can give the following simultaneous recursive definition of *DRSs* (terms are expressions which are either individual constants or discourse markers):

Definition 1.1 (*DRSs* and conditions)
The set of conditions and the set of *DRSs* are the smallest sets such that:

1. if R is an n-place relational constant, and $t_1, \ldots t_n$ are terms then $Rt_1, \ldots t_n$ is a condition
2. if t_1 and t_2 are terms then $t_1 = t_2$ is a condition
3. if K is a *DRS* then $\neg K$ is a condition
4. if K_1 and K_2 are *DRSs* then $K_1 \rightarrow K_2$ is a condition
5. if $x_1, \ldots x_n$ are discourse markers $(0 \leq n)$, and $c_1, \ldots c_m$ are conditions $(0 \leq m)$, then $\langle x_1, \ldots x_n \colon c_1, \ldots c_m \rangle$ is a *DRS*

Only the last clause specifies the format of a *DRS*. They must be conceived of as pairs consisting of a set of discourse markers and a set of (occurrences of) conditions. Conditions are either atomic conditions $Rt_1, \ldots t_n$ or $t_1 = t_2$, or they are conditions consisting of the negation $\neg K$ of a *DRS* K or implicational conditions $K_1 \rightarrow K_2$ relating two *DRSs* K_1 and K_2.

As for an illustration, let us consider the result of processing example 5 in the presently defined format of *DRSs*. Let K_i be a *DRS* $\langle x_1, \ldots x_n \colon c_1, \ldots c_m \rangle$ with respect to which that example is processed. Then the result is the following *DRS*:

$$\langle x_1, \ldots x_n, y, z \colon c_1, \ldots c_m, y = j, Wz, Lyz, Hzx \rangle$$

where y and z are not an element of $\{x_1, \ldots x_n\}$ and either $x \in \{x_1, \ldots x_n\}$ or x is y. The discourse markers $x_1, \ldots x_n, y, z$ in the universe of the resulting *DRS* are available as antecedents for future anaphoric coreference.

Truth in DRT

In the original paper by Kamp, [1981], *DRSs* are conceived of as partial models which embody the conditions which the world must satisfy in order for the represented discourse to be true. The satisfiability of a *DRS* K by a world or model is defined in terms of the existence of a proper embedding of the universe of K into that of the model which validates the conditions on discourse markers spelled out in K.

Like the formulas of first order predicate logic, *DRSs* are interpreted with respect to a model and an assignment function, which is called an embedding in the *DRT* framework. A model consists of a domain D of individuals and an interpretation function F which maps the individual constants of the *DRS* language onto individuals in D and n-ary relational constants onto sets of n-tuples of individuals. An assignment function maps discourse markers onto individuals.

The truth of a *DRS* is defined employing the following two notions which are given a simultaneous recursive definition:

$\models_{M,g} c$, condition c is true in model M under assignment g
$h \models_{M,g} K$, assignment h is a verifying assignment of *DRS* K in model M with respect to assignment g

The definition runs as follows (for any term t, $[\![t]\!]_{M,g}$ is $F(t)$ if t is an individual constant, and $g(t)$ if t is a discourse marker; $g[x_1, \ldots, x_n]h$ says that assignment g is like h except, possibly, for the values assigned to $x_1, \ldots, x_n$):

Definition 1.2 (DRS semantics)

1. $\models_{M,g} Rt_1 \ldots t_n$ iff $\langle [\![t_1]\!]_{M,g}, \ldots, [\![t_n]\!]_{M,g} \rangle \in F(R)$
2. $\models_{M,g} t_1 = t_2$ iff $[\![t_1]\!]_{M,g} = [\![t_2]\!]_{M,g}$
3. $\models_{M,g} \neg K$ iff for no h: $h \models_{M,g} K$
4. $\models_{M,g} K_1 \to K_2$ iff for every h: $h \models_{M,g} K_1$, there is a k: $k \models_{M,h} K_2$
5. $h \models_{M,g} \langle x_1, \ldots x_n\colon c_1, \ldots c_m \rangle$ iff $g[x_1, \ldots, x_n]h$ & $\models_{M,h} c_1$ & $\ldots \models_{M,h} c_n$

A *DRS* K is true in model M with respect to assignment g iff there exists an assignment h such that $h \models_{M,g} K$.

An atomic formula $Rx_1 \ldots x_n$ is true in model M under assignment g iff the sequence of values of $x_1, \ldots x_n$ under g is in the extension of R in M. An identity statement $x_1 = x_2$ is true under assignment g iff the values of x_1 and x_n under g are identical. A negated *DRS* $\neg K$ is true in model M under assignment g iff K is not true in M with respect to g. And an implicational condition $K_1 \to K_2$ is true in M with respect to g if K_2 is true in M with respect to *every* verifying assignment of K_1 in M with respect to g. Finally, h is a verifying assignment of $K = \langle x_1, \ldots x_n\colon c_1, \ldots c_m \rangle$ with respect to g iff h at most differs from g with respect to the values assigned to $x_1, \ldots x_n$ and every condition c_i $(0 \leq i \leq m)$ is true in M under h.

The effect of this truth definition is that the values of the discourse markers in the universe of a *DRS* $K_i = \langle x_1, \ldots x_n\colon c_1, \ldots c_m \rangle$ are quantified over in a way which depends on the context of K_i. If we are concerned with the truth of K_i itself with respect to assignment g, then these values are existentially quantified over, since the truth of K_i requires there to be possible values of these discourse markers, encoded in some assignment h: $g[x_1, \ldots, x_n]h$, with respect to which all conditions in K_i are true. However, if K_i is the antecedent representation of an implicational condition $K_i \to K_j$, then they are universally quantified over. The truth of the condition $K_i \to K_j$ under an assignment g requires that K_j be true with respect to every assignment h: $g[x_1, \ldots, x_n]h$ with respect to which the conditions in K_i are true, that is, in effect, under *all* assignments of values to $x_1, \ldots, x_n$ with respect to which these conditions are true.

Donkey sentences in DRT

I will now give an indication of how the donkey examples discussed above are treated in *DRT*. Consider the following simplification of example 1:

(1) A man walks. He whistles.

With respect to some initial *DRS* $K_i = \langle x_1, \ldots x_n$: $c_1, \ldots c_m\rangle$, the construction algorithm applies first to the first sentence of example 1, and next to the second sentence. The processing of the first sentence with respect to K_i involves the addition of a discourse marker y to the universe of K_i and the addition of the conditions that y is a man and that y walks in the park, so we arrive at the following *DRS* K_j:

$\langle x_1, \ldots, x_n, y$: $c_1, \ldots c_m, My, Wy\rangle$

The second sentence is processed with respect to this *DRS*. In this sentence we find the pronoun *he*, which must be associated with a discourse marker in an accessible universe. In the present example y is a suitable candidate, and if *he* is associated with y, then the construction algorithm yields the following representation of example 1:

$\langle x_1, \ldots, x_n, y$: $c_1, \ldots c_m, My, Wy, WHy\rangle$

The example is true with respect to some assignment g iff there is an assignment h: $g[x_1, \ldots, x_n, y]h$ such that $c_1, \ldots c_m$ are true under h, and $h(y)$ is a man who walks and whistles. Truth-conditionally, this amounts to the requirement that K_i is true and that there is man who walks and who whistles.

Next consider example 2:

(2) If a farmer owns a donkey, he beats it.

As has already been indicated, the application of the construction algorithm to conditional sentences introduces implicational conditions. If the example is evaluated with respect to some *DRS* K, K is extended with a condition $K_i \rightarrow K_j$, where K_i is a *DRS* for the antecedent clause *a farmer owns a donkey*, and K_j one for the clause *he beats it*. More precisely, K is extended with the following condition:

$\langle x, y$: $Fx, Dy, Oxy\rangle \rightarrow \langle Bxy\rangle$

If (with respect to some model and assignment) h is a verifying assignment of K, then the addition of the above condition to K imposes a supplementary requirement that the consequent *DRS* $\langle Bxy\rangle$ is true with respect to every assignment k which is a verifying assignment of the antecedent *DRS* $\langle x, y$: $Fx, Dy, Oxy\rangle$ with respect to h. More specifically, it is required that for every assignment k, if k at most differs from h with respect to the values assigned to x and y, and if $k(x)$ is a farmer and $k(y)$ a donkey which $k(x)$ owns, then $k(x)$ beats $k(y)$. In fact, this comes down to the requirement that every farmer beats every donkey he owns.

The application of the construction algorithm to example 3, finally, gives rise to the same *DRSs* as the ones produced for example 2. Hence, the truth conditions of the two examples are identical.

1.2 Dynamic predicate logic

We have seen that donkey-sentences, and intersentential anaphoric relationships, while not accounted for in a Montague grammar, are successfully dealt with in

DRT. However, the latter theory is a representational, non-compositional system of interpretation. So, one might say, the latter system does not give a solution to the problem of accounting for the phenomena in question in a compositional fashion. For this reason, the basic architecture of *DRT* is not compatible with Montague grammar and this obstructs comparison and unification.

Groenendijk and Stokhof [1991, (the original paper dates from 1987)] constitutes a first step in the improvement of this situation.[4] This paper presents a system of predicate logic, called dynamic predicate logic (*DPL*), which obtains the same results as Kamp's original formulation of *DRT* (Kamp [1981]) in a non-representational and more compositional fashion. The subsequently developed system of *DMG* (Groenendijk and Stokhof [1990a]) finally incorporates the results of *DPL* into a fully compositional Montague grammar (see section 3).

DRT has shown that certain differences between the roles of sentences in discourse cannot be appropriately regarded as a difference in truth-conditional content. For instance, compare the first sentence of the following two examples, which are truth-conditionally equivalent:

(a) A man walks in the park. He whistles.
(b) Not every man does not walk in the park. *He whistles

The first sentence in the first example licenses subsequent anaphora whereas the first sentence of the second example does not. In *DRT* this phenomenon is explained in terms of the different representations associated with the two sentences. The discourse representation for the first sentence has a discourse marker associated with the subject noun phrase which is available for future anaphoric reference. The discourse marker associated with the noun phrase *every man* in the representation of the second sentence is not, since it is introduced in a *DRS* subordinate to the main *DRS*.

Groenendijk and Stokhof accurately observe that what examples like the two above show is *not* that a level of discourse representation is essential for an account of the anaphoric relationships. What such examples show is that the meaning of a sentence is not fully characterized in terms of its truth conditions. From the definition of the interpretation of *DRSs* it appears that the most central notion is not that of the truth of a *DRS* condition or of a *DRS*, but that of a verifying assignment of a *DRS*. Simply by extending the notion of the meaning of a sentence with that of assignments verifying the sentence, Groenendijk and Stokhof succeed in getting a compositional, non-representational interpretation of the language of first order predicate logic that covers the results of *DRT*.

4. Other compositional reformulations of *DRT* have been given, for instance, by Barwise [1987], Rooth [1987], Asher and Wada [1988], Zeevat [1989], Groenendijk and Stokhof [1990a], Muskens [1990].

Relations between assignments

DPL takes as its starting point the view that the meaning of a sentence does not lie in its truth conditions, but rather in the way it changes the information of the hearer. Such information may involve, among others, truth-conditional information about what the world is like, but also information about what are possible antecedents of anaphoric pronouns. Since *DPL* in the first place aims to give a reformulation of *DRT*, it focusses attention on the latter aspects of information.

As in certain approaches to the semantics of programming languages, the meaning of a sentence (read: program) is captured in terms of a relation between states. In *DPL* the states are assignment functions. The interpretation of a sentence then constitutes a set of ordered pairs of assignments, each element of which constitutes a possible 'input–output' pair. If the pair of assignments $\langle g, h\rangle$ is in the interpretation of a sentence ϕ, then, one might say, the interpretation of ϕ in state g may have the, possibly new, state h as a result.

Adopting such a procedural notion of meaning, it is relatively obvious what the interpretation of conjunction must be. Conceiving of the interpretation of a discourse as the sequential interpretation of its successive sentences, the interpretation of a conjunction $\phi \wedge \psi$ is the composition of the interpretations of ϕ and ψ. With respect to some initial assignment g, the result of interpreting $\phi \wedge \psi$ may be some state h iff h may be the result of interpreting ψ with respect to some assignment k which may result from interpreting ϕ with respect to g.

The most important action in *DPL* originates from occurrences of existential quantifiers. An existential quantifier $\exists x$ has the effect of a random assignment of a value to the variable x, which is like in ordinary predicate logic. However, in contradistinction with predicate logic, the interpretation of an existentially quantified formula $\exists x\phi$ in *DPL* as it were 'remembers' the randomly assigned values of x which satisfy the conditions imposed upon the value of x by ϕ. Since these values are encoded in the possible output assignments, they can be referred back to.

Consider, for instance, the formula $\exists xMx$. Assuming that the extension of M is the set of men, this formula is true with respect to some assignment g in ordinary predicate logic iff for some assignment h, $g[x]h$ and $h(x)$ is a man. The same requirement must be met in order for the formula to be *true* in *DPL*. However, the *DPL interpretation* of the formula with respect to g will be any such assignment h. In other words, the *DPL* interpretation of $\exists xMx$ is that set of pairs $\langle g, h\rangle$ of assignments g and h such that $g[x]h$ and $h(x)$ is a man.

Now consider the conjunction of $\exists xMx$ with the formula Wx, where W denotes the set of walking individuals. The latter formula is true with respect to some assignment h iff $h(x)$ walks. Employing the interpretation of the conjunction of two formulas indicated above, this conjunction will be true with respect to an assignment g iff Wx is true with respect to some state h which may result from interpreting

$\exists xMx$ with respect to g. Spelling these conditions out we find that the whole conjunction is true with respect to g iff there is some assignment h such that $g[x]h$ and $h(x)$ is a man and $h(x)$ walks.

The above example illustrates the benefits of *DPL* interpretation. Employing the independent interpretations of the two conjuncts $\exists xMx$ and Wx, we can derive the interpretation of their conjunction $\exists xMx \wedge Wx$ which is associated with the truth conditions that there is a man who walks. Furthermore, the interpretation of $\exists xMx \wedge Wx$ with respect to some state g 'remembers' that the value of x is a man who walks. So, a subsequent use of the variable ('discourse marker') x can be associated with the possible values of x which are introduced by the existentially quantified formula $\exists xMx$.

DPL definitions

The language of *DPL* is that of ordinary predicate logic. It has individual constants and variables (which make up the set of terms) and n-ary relational constants. Atomic formulas are $Rt_1 \ldots t_n$ and $t_1 = t_2$, where R is an n-ary relational constant and $t_1, \ldots, t_n$ are terms. Furthermore, if ϕ and ψ are formulas and x is a variable, then $\neg\phi$, $\exists x\phi$ and $(\phi \wedge \psi)$ are formulas.

Like in ordinary predicate logic and in *DRT*, *DPL* interpretation is defined with respect to a model $M = \langle D, F\rangle$ which consists of a domain D of individuals and an interpretation function F assigning objects in D to the individual constants and sets of n-tuples of individuals to n-ary relational constants. Again $[\![t]\!]_{M,g}$ is $g(t)$ if t is a variable, and $F(t)$ if t is a constant, and $g[x]h$ says that assignment h is like assignment g except possibly with respect to the value it assigns to x.

The interpretation of *DPL* formulas is defined as follows:

Definition 1.3 (DPL semantics)

1. $[\![Rt_1 \ldots t_n]\!]_M = \{\langle g,h\rangle \mid h = g \text{ and } \langle [\![t_1]\!]_{M,g}, \ldots, [\![t_n]\!]_{M,g}\rangle \in F(R)\}$
2. $[\![t_1 = t_2]\!]_M = \{\langle g,h\rangle \mid h = g \text{ and } [\![t_1]\!]_{M,g} = [\![t_2]\!]_{M,g}\}$
3. $[\![\neg\phi]\!]_M = \{\langle g,h\rangle \mid h = g \text{ and } \neg\exists h\text{: } \langle g,h\rangle \in [\![\phi]\!]_M\}$
4. $[\![\exists x\phi]\!]_M = \{\langle g,h\rangle \mid \exists k\text{: } g[x]k \text{ and } \langle k,h\rangle \in [\![\phi]\!]_M\}$
5. $[\![\phi \wedge \psi]\!]_M = \{\langle g,h\rangle \mid \exists k\text{: } \langle g,k\rangle \in [\![\phi]\!]_M \text{ and } \langle k,h\rangle \in [\![\psi]\!]_M\}$

A formula is true with respect to a model M and assignment g iff there is an assignment h such that $\langle g,h\rangle \in [\![\phi]\!]_M$.

Atomic formulas are interpreted as so-called 'tests'. Given an input assignment g the interpretation of such formulas either produces g, if the test succeeds, or has no output, if the test fails. They accept as possible input the assignments with respect to which the formulas are true in ordinary predicate logic and with respect to which the corresponding *DRT* conditions are true. The interpretation of the negation $\neg\phi$ of a formula ϕ also involves a test. It accepts as possible input those assignments g

of which it holds that the interpretation of ϕ with respect to g has no output. The interpretation of existentially quantified formulas and of conjunctions is defined as sketched above.

As for a first comparison of *DPL* with *DRT* the following things may be observed. In the first place, *DRT* and *DPL* completely agree on the interpretation of atomic formulas and negations. The mere difference is that in *DPL* they are conceived of as formulas (or *DRSs*, for that matter) in their own right, instead of as conditions, as is the case in *DRT*. In the second place, *DRT* has implicational conditions, and a rule of *DRS* formation, where *DPL* has operations of existential quantification and conjunction.

It is relatively easy to see that a *DRT* implication $K_1 \rightarrow K_2$ corresponds to a *DPL* implication $\phi \rightarrow \psi$ where ϕ is the *DPL*-equivalent of K_1 and ψ that of K_2 and where $\phi \rightarrow \psi$ is defined, as usual, as $\neg(\phi \wedge \neg\psi)$. Furthermore, the interpretation of a *DRS* $K_i = \langle x_1, \ldots x_n \colon c_1, \ldots c_m \rangle$ is easily seen to be equivalent with the iterated existential quantification in *DPL* over the values of $x_1, \ldots x_n$ in the conjunction of the formulas $\phi_1, \ldots \phi_m$ that correspond to the conditions $c_1, \ldots c_m$.

On the other hand, *DRT* lacks an operation which corresponds to the *DPL*'s conjunction. In fact, the presence of such an operation, with a well-defined (dynamic) interpretation, enables *DPL* to give a more compositional translation of natural language than *DRT* does.

Truth conditions of DPL formulas

In order to illustrate the semantic properties of *DPL*, it is elucidating to give a truth conditions preserving translation into ordinary predicate logic. This translation of a formula ϕ yields what Groenendijk and Stokhof call the 'normal binding form' of ϕ. It is defined as follows:

Definition 1.4

1. $(Rt_1 \ldots t_n)^{\clubsuit} = Rt_1 \ldots t_n$
2. $(\neg\phi)^{\clubsuit} = \neg(\phi)^{\clubsuit}$
3. $(\exists x\phi)^{\clubsuit} = \exists x(\phi)^{\clubsuit}$
4. $(Rt_1 \ldots t_n \wedge \psi)^{\clubsuit} = (Rt_1 \ldots t_n)^{\clubsuit} \wedge (\psi)^{\clubsuit}$
   $(\neg\phi \wedge \psi)^{\clubsuit} = (\neg\phi)^{\clubsuit} \wedge (\psi)^{\clubsuit}$
   $(\exists x\phi \wedge \psi)^{\clubsuit} = (\exists x(\phi \wedge \psi))^{\clubsuit}$
   $((\phi \wedge \psi) \wedge \chi)^{\clubsuit} = (\phi \wedge (\psi \wedge \chi))^{\clubsuit}$

A formula ϕ is true with respect to M and assignment g in *DPL* iff $\phi^{\clubsuit}$ is true with respect to M and assignment g in predicate logic.

The translation $^{\clubsuit}$ clearly reveals the semantic properties of *DPL*. For as far as truth conditions are concerned, atomic formulas and sentential operators behave in exactly the same way in *DPL* as in predicate logic, with the following exception. In *DPL*, a formula $(\exists x\phi \wedge \psi)$ is equivalent with the formula $\exists x(\phi \wedge \psi)$. It is exactly this distinguishing equivalence in which we recognize the (dynamic) account of establishing of anaphoric relationships in *DPL*. For instance, consider again the sequence *A man walks. He whistles*, which is most intuitively translated as $\exists x(Mx \wedge Wx) \wedge WHx$. In *DPL* this formula is equivalent with the formula $\exists x(Mx \wedge Wx \wedge WHx)$, which is the natural translation of the sentence *There is a man who walks and whistles*. Notice, moreover, that precisely because of the last mentioned equivalence, conjunction is not in general commutative in *DPL*. The *DPL* truth conditions of $\exists xMx \wedge Wx$ equal the predicate logic truth conditions of $\exists x(Mx \wedge Wx)$, which are different from the *DPL* and predicate logic truth conditions of $Wx \wedge \exists xMx$.

DRT and DPL

DPL has been conceived of here as a reformulation of *DRT*. *DPL* provides for a meaning-preserving restatement of the *DRS* language into the more familiar language of predicate logic, which is given a recursively defined dynamic interpretation basically similar to that of the *DRS* language. However, the difference between the two is significant.

The meanings of natural language discourses can be expressed by *DPL* formulas which more closely reflect the structure of these discourses than the corresponding *DRSs* do. For instance, compare the *DPL* translation 1a of example 1 with the basic *DRS* 1b for the example in *DRT*, viz., the *DRS* constructed with respect to the 'empty' input *DRS* $\langle\ \rangle$:

(1) A man walks. He whistles.
(1a) $\exists x(Mx \wedge Wx) \wedge WHx$
(1b) $\langle x\colon Mx, Wx, WHx\rangle$

In formula 1a we can recognize the translations of the two sentences that make up example 1. The translation of example 1 simply consists of the (*DPL*-)conjunction of the translations of its constituent sentences. However, no constituent of the *DRS* 1b can be conceived of as the basic *DRT* representation of the first (or second) sentence of example 1, viz., $\langle x\colon Mx, Wx\rangle$ ($\langle WHx\rangle$). The reason is that the *DRT* interpretation of the second sentence in this example involves a modification of the representation of the first sentence and not an operation that conjoins the *DRSs* associated with the two constituent sentences.

Next, consider the *DPL* translation 3a of example 3 and the basic *DRS* 3b for the example:

(3) Every farmer who owns a donkey beats it.
(3a) $\forall x((Fx \wedge \exists y(Dy \wedge Oxy)) \rightarrow Bxy)$
(3b) $\langle\langle x, y\colon Fx, Dy, Oxy\rangle \rightarrow \langle\ \colon Bxy\rangle\rangle$

In the *DRS* 3b for example 3 we find no representation of the constituent clause *who owns a donkey*. However, the *DPL* translation 3a has a subformula which is the translation of this relative clause, viz., $\exists y(Dy \wedge Oxy)$. Examples like the above motivate the conclusion that natural language expressions can be translated in a more compositional fashion into the language of *DPL*, than in the representational language of *DRT*.

Finally, notice that *DPL* still does not allow for a fully compositional translation of, for instance, quantifying noun phrases, the reason being that *DPL*, like ordinary predicate logic, is a simple first order logic. So, in order to develop a dynamic Montague grammar, the *DPL* operations have to be lifted to a higher order logic. The next section shows how this can be done. It presents Groenendijk and Stokhof's dynamic Montague grammar, which offers a compositional and dynamic semantics of a fragment of natural language.

2 Dynamic Montague grammar

In the construction of a dynamic Montague grammar, the development of *DPL* can only be a first step. *DPL* is just a first order logic and, therefore, it can not be used to give a compositional account of the semantics of natural language at the subsentential level. So, the next step is to define a typed logic that encompasses dynamic interpretation. Groenendijk and Stokhof [1990a] define a dynamic Montague grammar (*DMG*) in terms of a variant of intensional logic due to Janssen [1986]. They show that this Montague grammar incorporates basic *DRT* results in a completely compositional way. This section gives an exposition of Groenendijk and Stokhof's system. I start with an exposition of dynamic intensional logic (*DIL*), the variant of *IL* that is used in *DMG*.

2.1 Dynamic intensional logic

In order to give a further compositionalization of the dynamic interpretation of natural language, the natural way to go about is to resort to a *typed* logic. However, not any typed logic will do. For instance, the logic must enable us to express the ways in which the interpretations of pronouns may vary from context to context, and the way in which certain expressions may change the context of interpretation for other expressions. Given the fact that we can conceive of the meaning of a pronoun as a function from contexts to individuals, a logic is preferred that has a device which indicates abstraction over contexts. Furthermore, if the logic also has a device indicating application to contexts, it may be able to express functions, which, when applied to the meaning of a pronoun, involve the evaluation of the pronoun with respect to a context different from the context with respect to which the function itself is evaluated, and, thus, express a notion of context change for the interpretation of pronouns.

A logic that serves these purposes is dynamic intensional logic (*DIL*), a variant of *IL*. In *DIL*, the set of states is made to behave like a set of assignments to so-called discourse markers, special constants of type e which are used in the translations of pronouns. Since the states behave like discourse marker assignments, the intension operator $^\wedge$ indicates abstraction over discourse marker assignments. Furthermore, the extension operator $^\vee$ in that logic involves the evaluation of an expression with respect to the current discourse marker assignment.

In this section I present the basic ingredients of *DIL*. For more details and discussion, the reader is referred to Janssen [1986] and Groenendijk and Stokhof [1990a].

Dynamic intensionality

The language of *DIL* has a distinguished set of discourse markers $d, d', \ldots$ among the constants of type e, which figure in the translation of pronouns. Since the discourse markers are *constants*, their interpretation depends on the index, or state, of evaluation, not on an assignment function. However, by means of postulates these states are made to behave like discourse marker assignments, and, thus, these discourse markers themselves are made to act like variables.

The discourse markers are even more made to look like variables, since *DIL* has so-called 'state quantifiers' $\exists d, \exists d', \ldots$ which in effect involve quantification over the values of discourse markers.[5] In fact, a state quantifier $\exists d$ is a modal operator that quantifies over states. By means of the postulates mentioned above this is related to quantification over discourse marker assignments, and the interpretation of a formula $\exists d\phi$ in effect involves a form of existential quantification over the values of the discourse marker d.

So, apart from the ordinary variable binding devices, *DIL* has a *modal* 'variable' binding device in terms of discourse markers and state quantifiers. (For motivation and discussion, see Groenendijk and Stokhof [1990a].) Because of this modal nature of discourse markers and state quantifiers, they interact in an interesting way with the intension and extension operators $^\wedge$ and $^\vee$. For instance, since states act like assignment functions, the intension $^\wedge d$ of a discourse marker d in *DIL* denotes the function f from the domain of states (discourse marker assignments) to the domain of individuals D the value of which for any state s is the value of d in s.

On the other hand, a state quantifier $\exists d$ affects occurrences of the extension operator $^\vee$ in its scope. For instance, for the formula $\exists d(\mathtt{man}(d) \wedge {^\vee p})$ to be true,

5. Here I deviate from Janssen and Groenendijk and Stokhof who use so-called state switchers $\{\alpha/d\}$, $\{\beta/d'\}$. However, Groenendijk and Stokhof only use state switchers in combination with existential quantifiers, the combined interpretation of which is expressed by the state quantifiers employed here.

the extension of p must be true, not necessarily in the state s with respect to which interpretation of the entire formula takes place, but in a state s' possibly differing from s in that the value of d is a man in s'. So, if we substitute $^\wedge\mathtt{whistle}(d)$ for p in the present example, we find that the resulting sentence is true in s iff there is state s' which differs from s at most in that the value of d is a man in s' and such that the value of d whistles in s'. In other words, in that case the formula $\mathtt{whistle}(d)$ is in effect interpreted with respect to a state (assignment) in which d is a man.

Before turning to the definitions of *DIL*, it must be noticed that *DIL*, as it stands, does not deal with *IL*'s intensionality. In the system presented below, all of the intensionality is concerned with the interpretation of discourse markers. All other constants are 'rigid', i.e., they have the same extension in each state. Intensionality in the ordinary sense can be incorporated in *DIL*. However, for the present purposes it suffices to stick to the 'extensional' system, the only intensionality of which is the modal 'variable' binding device. It should be noticed, furthermore, that only discourse markers of type e are used.[6] I now turn to the definitions of *DIL*.

DIL: types and syntax

The system of dynamic intensional logic differs from intensional logic *IL* in three respects. First, a distinguished set of discourse markers *DM* is recognized among the constants of type e. Second, instead of *IL*'s modal and temporal operators, *DIL* has state quantifiers $\exists d$, for any discourse marker d. Third, three postulates make the states behave like discourse marker assignments. I start with the recursive definition of the set of *DIL* types:

Definition 2.1 (DIL types)
The set of types T is the smallest set such that:

1. $e, t \in T$
2. If $a, b \in T$, then $\langle a, b\rangle \in T$
3. If $a \in T$, then $\langle s, a\rangle \in T$

The set T of *DIL* types contains the basic types e and t, the type of individuals (or entities) and the type of truth values. Furthermore, it contains derived functional types, $\langle a, b\rangle$ and $\langle s, a\rangle$. The first is the type of functions from objects of type a to objects of type b, and the last is the intensional type of functions from states to objects of type a.

6. The language of *DIL* can be extended with discourse markers of any extensional type without complication. However, the introduction of discourse markers of intensional types involves a foundational problem. Since states are postulated to behave as discourse marker assignments, the interpretation of a discourse marker d of an intensional type $\langle s, a\rangle$, which is the type of functions from states to objects of type a, would have to be defined for states that, by postulate, define the interpetation of d. This problem can be gotten around in various ways. See Janssen [1986] and [1990] for the introduction of discourse markers of types other than type e.

Assuming for each type a a set CON_a of constants of type a, with a distinguished set DM of discourse markers among the constants of type e, and a set VAR_a of variables of type a, the set of well-formed expressions of *DIL* is defined as follows:

Definition 2.2 (DIL syntax)
The set WE_a of well-formed expressions of type a, for any type $a \in T$, is the smallest set such that for all types $a, b \in T$:

1. If $c \in CON_a$ then $c \in WE_a$
   If $x \in VAR_a$ then $x \in WE_a$
2. If $\beta \in WE_{\langle a,b\rangle}$ and $\alpha \in WE_a$ then $\beta(\alpha) \in WE_b$
   If $\beta \in WE_b$ and $x \in WE_a$ then $\lambda x\ \beta \in WE_{\langle a,b\rangle}$
3. If $\alpha \in WE_a$ and $\beta \in WE_a$ then $(\alpha = \beta) \in WE_t$
4. If $\phi \in WE_t$ then $\neg\phi \in WE_t$
   If $\phi \in WE_t$ and $x \in VAR_a$ then $\exists x\phi \in WE_t$
   If $\phi \in WE_t$ and $d \in DM$, then $\exists d\phi \in WE_t$
   If $\phi, \psi \in WE_t$ then $(\phi \wedge \psi) \in WE_t$
5. If $\alpha \in WE_a$ then ${}^\wedge\alpha \in WE_{\langle s,a\rangle}$
   If $\alpha \in WE_{\langle s,a\rangle}$ then ${}^\vee\alpha \in WE_a$

In what follows, $\forall x\phi$, $(\phi \rightarrow \psi)$, and $(\phi \vee \psi)$ abbreviate $\neg\exists x\neg\phi$, $\neg(\phi \wedge \neg\psi)$, and $(\neg\phi \rightarrow \psi)$ respectively.

So, the language of *DIL* consists of expressions built up from constants and variables by means of functional application and λ-abstraction, identity, the usual connectives and quantifiers, the intension and extension operators $^\wedge$ and $^\vee$, and the state quantifier.

DIL: domains and postulates

I now turn to the semantics of *DIL*. First the interpretation domains for expressions of the various types are defined, on the basis of a set D of individuals and a set I of states. If D and I are such sets, then for any type a the set D_a of possible denotations of (expressions of) type a is defined as follows:

Definition 2.3 (DIL domains)

1. $D_e = D$
   $D_t = \{0,1\}$
2. $D_{\langle a,b\rangle} = D_b^{D_a}$
3. $D_{\langle s,a\rangle} = D_a^I$

Expressions of type e have a denotation in the domain of individuals and the possible denotations of expressions of type t are the truth values 1 (true) and 0 (false). The domain $D_{\langle a,b\rangle}$ is the set of functions with domain D_a and values in D_b. The domain $D_{\langle s,a\rangle}$ is the set of functions from states to objects in D_a.

The semantics of *DIL* is defined with respect to a model $M = \langle D, I, F\rangle$ where D and I are as above, and F is an interpretation function for the constants of the language such that for any constant c of type a, $F(c)$ is a function from I to D_a. A model M is required to satisfy the following three postulates:

Postulate 1 (Rigidness postulate)

- For all $c \in CON$, if $c \notin DM$ then for all $s, s' \in I$: $F(c)(s) = F(c)(s')$

This postulate guarantees that all of the intensionality of *DIL* resides in the interpretation of discourse markers. As has been indicated above, the intensionality phenomena that *IL* deals with are not dealt with here. (However, dropping this postulate evidently gives us the resources of (classical) intensionality back again.)

Postulate 2 (Distinctness postulate)

- For all $s, s' \in I$, if for all $c \in CON$: $F(c)(s) = F(c)(s')$ then $s = s'$

Postulate 3 (Update postulate)

- For all $s \in I$, $d \in DM$, $z \in D_e$ there is an $s' \in I$ such that:
  1. for all $c \in CON$, if $c \neq d$ then $F(c)(s) = F(c)(s')$, and
  2. $F(d)(s') = z$

The last two postulates make the set of states correspond to discourse marker assignments. Together with the first postulate they guarantee that every state uniquely corresponds to a discourse marker assignment. A state s corresponds to the discourse marker assignment g such that $\forall d \in DM$: $g(d) = F(d)(s)$.[7] The postulates imply, furthermore, that for very state s, discourse marker d and object z, there is a unique state s', such that in s' the value of all constants except d is the same as in s and such that the value of d in s' is z. This state will be indicated by $s[d/z]$, which reflects the usual notation $g[x/z]$ for the assignment h which at most differs from g in that the value of x with respect to h is z.

DIL: interpretation

I now turn to the interpretation of well-formed *DIL* expressions. The interpretation $[\![\alpha]\!]_{M,s,g}$ of an expression α with respect to a model M, state $s \in I$ and assignment g is defined as follows:

Definition 2.4 (DIL semantics)

1. $[\![c]\!]_{M,s,g} = F(c)(s)$, for all constants c
   $[\![x]\!]_{M,s,g} = g(x)$, for all variables x
2. $[\![\beta(\alpha)]\!]_{M,s,g} = [\![\beta]\!]_{M,s,g}([\![\alpha]\!]_{M,s,g})$
   $[\![\lambda x_a \beta_b]\!]_{M,s,g}$ = the function $h \in D_b^{D_a}$: $h(z) = [\![\beta]\!]_{M,s,g[x/z]}$ for all $z \in D_a$

7. If, in order to re-introduce ordinary intensionality in *DIL*, the rigidness postulate is dropped, then this correspondence will have to be relativized.

3. $[\![\alpha = \beta]\!]_{M,s,g} = 1$ iff $[\![\alpha]\!]_{M,s,g} = [\![\beta]\!]_{M,s,g}$
4. $[\![\neg\phi]\!]_{M,s,g} = 1$ iff $[\![\phi]\!]_{M,s,g} = 0$
   $[\![\exists x_a \phi]\!]_{M,s,g} = 1$ iff there is a $z \in D_a$: $[\![\phi]\!]_{M,s,g[x/z]} = 1$
   $[\![\exists d\phi]\!]_{M,s,g} = 1$ iff there is a $z \in D_e$: $[\![\phi]\!]_{M,s[d/z],g} = 1$
   $[\![\phi \wedge \psi]\!]_{M,s,g} = 1$ iff $[\![\phi]\!]_{M,s,g} = [\![\psi]\!]_{M,s,g} = 1$
5. $[\![{}^\wedge\alpha_a]\!]_{M,s,g}$ = the function $h \in D_a^I$: $h(s') = [\![\alpha]\!]_{M,s',g}$ for all $s' \in I$
   $[\![{}^\vee\alpha]\!]_{M,s,g} = [\![\alpha]\!]_{M,s,g}(s)$

Like in *IL*, constants are assigned an interpretation relative to the state of evaluation. However, all constants which are not discourse markers receive the same interpretation in every state. By the rigidness postulate, only the interpretation of discourse markers may vary from state to state. Except for the clause dealing with state quantifiers, all clauses in the definition of $[\![\ ]\!]$ are as in *IL*.

A formula $\exists d\phi$ is defined to be true in a state s iff ϕ is true in a state s' that is like s except, possibly, for the value that d has in s'. Since the postulates guarantee that for any object z there is an alternative state s' in which the value of d is z, the state quantifier in effect quantifies over possible values of d. In this respect, the state quantifier behaves exactly like the ordinary existential quantifier.

Truth and entailment are defined as in *IL*:

Definition 2.5 (Truth, entailment and equivalence)
For all expressions ϕ and ψ of type t, and α and β of any type a,
- ϕ is true in M with respect to s and g, $M \models_{s,g} \phi$, iff $[\![\phi]\!]_{M,s,g} = 1$
- ϕ entails ψ, $\phi \models \psi$, iff for all M, s and g, if $M \models_{s,g} \phi$ then $M \models_{s,g} \psi$
- α and β are equivalent, $\alpha \Leftrightarrow \beta$, iff for all M, s and g: $[\![\alpha]\!]_{M,s,g} = [\![\beta]\!]_{M,s,g}$

Intensionality and λ-conversion in DIL

The intension and extension operators $^\wedge$ and $^\vee$ in *DIL* behave the same as in *IL*. These operators involve abstraction over, and application onto, states of evaluation, respectively. Of course, $^{\vee\wedge}$-elimination remains valid:

Fact 2.1 ($^{\vee\wedge}$-elimination)
- $^{\vee\wedge}\alpha \Leftrightarrow \alpha$

However, since states play the part of discourse marker assignments in *DIL*, the meaning of $^\wedge$ and $^\vee$ in *DIL* is different from that in *IL*. The meaning of an expression $^\wedge\phi$ is no longer considered to be the set of worlds, or situations, in which ϕ is true, but the set of (discourse marker) assignments with respect to which ϕ is true. Furthermore, as was indicated above, an occurrence of the extension operator $^\vee$ in the scope of a state quantifier $\exists d$ comes down to application to a state quantified over by means of $\exists d$. These two facts are crucial for *DIL*'s account of the dynamic binding of 'free' discourse markers, as we will see shortly. Before we see how such dynamic binding is realized, we first have to consider the conditions for λ-conversion in *DIL*.

The conditions for λ-conversion in *DIL* are virtually the same as in *IL*, but the definition of the relevant set of intensionally closed expressions is different:

Definition 2.6 (Intensionally closed expressions)
ICE, the set of intensionally closed expressions, is the smallest set such that:

1. $c, x \in ICE$ for any constant $c \notin DM$ and variable x
2. ${}^\wedge\alpha \in ICE$ for every well-formed expression α
3. $\beta \in ICE$ if β is constructed from elements of *ICE* by means of application, abstraction, identity, negation, quantification (both forms) and conjunction

Clearly, since the interpretation of constants that are not discourse markers is state independent, by the rigidness postulate, these constants can be included among the set of intensionally closed expressions. Discourse markers can only be part of intensionally closed expressions if they are in the scope of the ${}^\wedge$-operator. Expressions that are excluded from *ICE* are discourse markers and expressions fronted by the ${}^\vee$-operator.

The following two facts are as in *IL*:

Fact 2.2

- If β is intensionally closed, then $[\![\beta]\!]_{M,s,g} = [\![\beta]\!]_{M,s',g}$ for all $s, s' \in I$

Fact 2.3 (λ-conversion)

- $(\lambda x\beta)(\alpha) \Leftrightarrow [\alpha/x]\beta$ if:
  1. all free variables in α are free for x in β and
  2. α is intensionally closed

Notice, however, that in fact 2.3 no conditions on discourse markers in α are imposed. If all free variables in α are free for x in β, then the requirement that α is intensionally closed is a sufficient (not necessary) condition for λ-conversion. So, if α is of the form ${}^\wedge\phi$ and ϕ contains free discourse markers, λ-conversion is allowed, since the discourse markers in α are semantically bound by the operator ${}^\wedge$.

Mediated binding in DIL

The interplay between the λ-operator, the intension and extension operators and the state quantifier gives rise to what may be called the 'dynamic' or 'mediated' binding of discourse markers. Consider the following example, where p is a variable of type $\langle s, t\rangle$:

(6) $(\lambda p\ \exists d(\texttt{man}(d) \wedge {}^\vee p))({}^\wedge(\texttt{whistle}(d))) \Leftrightarrow$ (fact 2.3)
$\exists d(\texttt{man}(d) \wedge {}^{\vee\wedge}(\texttt{whistle}(d))) \Leftrightarrow$ (fact 2.1)
$\exists d(\texttt{man}(d) \wedge \texttt{whistle}(d))$

Clearly, the argument expression $^\wedge(\texttt{whistle}(d))$ in this example is intensionally closed. The argument denotes the function f from states to truth values such that $f(s) = 1$ iff the value of d in s whistles. The abstract $\lambda p\ \exists d(\texttt{man}(d) \wedge {}^\vee p)$ denotes, with respect to a state s, the function g from propositions to truth values such that $g(p) = 1$ iff there is a state s' such that s' at most differs from s with respect to the value that d has in s', the value of d is a man in s' and p holds of s'. Application of the functional expression to the argument expression yields true if there is a state s' such that s' at most differs from s with respect to the value that d has in s', the value of d is a man in s' and the value of d whistles in s'. This is equivalent to the statement that there is a man who whistles.

The above example shows a form of mediated binding, since the state quantifier $\exists d$ turns out to bind, semantically, a discourse marker d in $\texttt{whistle}(d)$ that does not occur in the syntactic scope of the quantifier. This form of binding is essentially mediated by the abstraction over the variable p in the scope of the state quantifier $\exists d$. The evaluation of the extension of p in the scope of $\exists d$, and the application of the whole λ-term to the intension of $\texttt{whistle}(d)$, makes that $\texttt{whistle}(d)$ in this example turns out to be evaluated as if it were in the scope of $\exists d$.

This example lays bare *DIL*'s potential to deal with anaphora in a completely compositional way. In *DIL*, two expressions can be assigned a meaning of their own and still, in the composition of the two, indefinites (state quantifiers) in the first may bind pronouns (discourse markers) in the second by mediation. Groenendijk and Stokhof elaborate such a treatment of anaphora in the framework of a dynamic Montague grammar, which is presented in the next subsection. Before we turn to their treatment, it is convenient to reflect a moment on the relation between the state quantifier and the ordinary existential quantifier.

The state quantifier behaves more or less like the existential quantifier, but, as we have seen, it behaves differently in interaction with the intension and extension operators. I will now show in which cases a formula $\exists d\phi$ can be equated with an ordinary existentially quantified formula. Let us first decide upon some terminology. An occurrence of a discourse marker d in ϕ is free iff it is not in the scope of an intension operator $^\wedge$ or of a state quantifier $\exists d$ in ϕ. Furthermore, an occurrence of $^\vee$ in ϕ is called free iff that occurrence is not in the scope of an intension operator $^\wedge$ in ϕ. Finally, $[x/d]\phi$ indicates the formula obtained from ϕ by substituting all free occurrences of d in ϕ by x. The following fact tells us when a state quantifier can be reduced to an ordinary existential quantifier:

Fact 2.4 ($\exists d$-reduction)

- $\exists d\phi \Leftrightarrow \exists x[x/d]\phi$ if
  1. x does not occur free in ϕ
  2. x is free for d in ϕ
  3. $^\vee$ does not occur free in ϕ

The first two conditions in this fact are standard conditions on substitution. The third condition is needed because of the modal nature of the state quantifier. For instance, the expression $\lambda p\ \exists d(\texttt{man}(d) \wedge {}^{\vee}p)$ in the above example can *not* be reduced to $\lambda p\ \exists x(\texttt{man}(x) \wedge {}^{\vee}p)$, since the state quantifier $\exists d$ affects the interpretation of the subformula ${}^{\vee}p$, which the existential quantifier $\exists x$ does not. On the other hand, since there is no free occurrence of ${}^{\vee}$ in $\exists d(\texttt{man}(d) \wedge \texttt{whistle}(d))$, this formula can be reduced, namely to $\exists x(\texttt{man}(x) \wedge \texttt{whistle}(x))$.

2.2 Dynamic interpretation in DMG

As was said above, *DIL* takes the part of *IL* in the dynamic Montague grammar proposed by Groenendijk and Stokhof. *DIL* has the expressive power that is required for a formulation of the dynamics of *DPL* within a completely compositional setting. However, *DIL* does not simply replace *IL* in *DMG*. In order to capture the dynamics of natural language that *DPL* deals with, the sentences of natural language should not be associated with the type t of truth values, as in *MG* (at least, not to start out with). The dynamics of natural language sentences is dealt with at a different level of types. In *DMG*, sentences are associated with the functional type of so-called context change potentials, the type of expressions that may affect the interpretation of other expressions.

In this section, I will first discuss the type of objects that serve as the denotations of natural language sentences in *DMG*, i.e., the type of *DMG*'s context change potentials. Then I discuss *DMG*'s (dynamic) counterparts of the usual (static) sentential operations at this higher level of types. These operators will be conceived of as belonging to another, fully typed, language, the semantics of which is stated in terms of *DIL* expressions. This intermediary language, *DFL*, greatly facilitates the study and exposition of *DMG*.[8] After that, in section 3.3, Groenendijk and Stokhof's *DMG*, a small dynamic Montague-style fragment of English, is defined and illustrated with some examples.

The type of context change potentials

DMG is a version of Montague grammar which accounts for the fact that the interpretation of a sentence may change the context of interpretation of sentences that follow it. Like in *DPL*, this is achieved by associating a sentence with a context change potential which is realized by the conjunction of the sentence with subsequent sentences.

8. In fact, Groenendijk and Stokhof do not use such an intermediary language. They do use derived notions of negation, existential quantification and conjunction (cf., below) but these are conceived of as notational devices which abbreviate complex *DIL* expressions. I have reinterpreted these abbreviatory devices as operators of a distinguished language for the purpose of generalization and comparison. Eventually, this is only a shift of perspective which is not a principled one, since the intermediary *DFL* language can be dispensed with, as the language of *DIL* can.

In the exposition of *DIL*, we have already encountered an example of so-called 'dynamic' or 'mediated' binding. When we apply the expression $\lambda p\ \exists d(\mathtt{man}(d) \wedge {}^{\vee}p)$ to the intension of the *DIL* formula $\mathtt{whistle}(d)$, the state quantifier $\exists d$ turns out to bind the discourse marker d in $\mathtt{whistle}(d)$. In fact, the intension of the expression $\lambda p\ \exists d(\mathtt{man}(d) \wedge {}^{\vee}p)$ can be taken to denote a context change potential. It denotes the function f from contexts (states) and propositions (sets of states) to truth values which holds of a state s and a proposition p iff p is true in a state s' which at most differs from s with respect to the value d has in s and such that the value of d in s' is a man. Hence, such a function can be conceived of as changing the context of interpretation for propositional expressions.

More in general, *DIL* expressions of the form ${}^{\wedge}\lambda p\ (\ldots {}^{\vee}p \ldots)$ may involve a change in the context of evaluation of propositional expressions to which they are applied. The denoted functions, properties of propositions of type $\langle s, \langle\langle s,t\rangle, t\rangle\rangle$, will be called contexts change potentials from now on.[9]

In *DMG*, the sentences of a fragment of natural language are translated into expressions which denote such context change potentials and these context change potentials are realized by applying their extensions to propositional expressions. For example, the sentence *A man walks* is associated with the (reduced) translation ${}^{\wedge}\lambda p\ \exists d(\mathtt{man}(d) \wedge \mathtt{walks}(d) \wedge {}^{\vee}p)$ and the sentence *He whistles*, with the translation ${}^{\wedge}\lambda p\ (\mathtt{whistle}(d) \wedge {}^{\vee}p)$. When the two sentences are conjoined, we take some form of intensional functional composition of the extensions of the translations of both sentences. This composition comes down to the application of $\lambda p\ \exists d(\mathtt{man}(d) \wedge \mathtt{walks}(d) \wedge {}^{\vee}p)$ to the intension of $(\mathtt{whistle}(d) \wedge {}^{\vee}p)$ and a subsequent abstraction over the value of p. The result can be reduced to the expression $\lambda p\ \exists d(\mathtt{man}(d) \wedge \mathtt{walks}(d) \wedge \mathtt{whistle}(d) \wedge {}^{\vee}p)$. Here we see, first, that the occurrence of d in the formula $\mathtt{whistle}(d)$ is bound by the state quantifier $\exists d$, and, second, that the result, as well, denotes a context change potential.

DFL: type shift

Like I said, the dynamics of natural language is formulated in *DMG* by translating expressions of a fragment into expressions of an intermediary language, *DFL*, the semantics of which is defined in terms of *DIL*. So, the organization of *DMG* can be

9. It must be noted that such functions do not literally change the context of interpretation of their argument expressions. The interpretation of the application a function f to some argument expression a with respect to M, s and g, of course, remains the application of the interpretation of f with respect to M, s and g to the interpretation of a with respect to M, s and g. However, the particular functions which figure as the denotations of sentences in *DMG* can all be phrased as functions which hold of a state s and a proposition p iff p is true in some state s' possibly different from s and such that so and so. For this reason it is appropriate to call them context change potentials, and the type $\langle s, \langle\langle s,t\rangle, t\rangle\rangle$ the type of context change potentials. (I just note that context change potentials can be modeled in other types as well, as we will see in chapter 3.)

pictured as follows:

$$\boxed{\text{AT}} \rightsquigarrow \boxed{\text{DFL}} \Longrightarrow \boxed{\text{DIL}} \rightarrow \boxed{\text{Models}}$$

Natural language expressions are translated into *DFL* expressions, the interpretations of which are defined in terms of *DIL* expressions which, in turn, are assigned a model-theoretic interpretation.[10] The intermediary system of *DFL* is some kind of an extensional dynamic logic. The formulas of *DFL* denote context change potentials and the system has notions of dynamic conjunction and dynamic existential quantification which are operations on expressions of that type. In fact, the type of context change potentials constitutes a basic type in *DFL*.

The set of *DFL* types T_D, is constructed from the type of individual concepts and from the type of context change potentials (i.e., the type of properties of propositions)[11]:

Definition 2.7 (DFL types)
The set of *DFL* types, T_D, is the smallest set such that:

1. $\langle s, e\rangle \in T_D$
2. $\langle s, \langle\langle s, t\rangle, t\rangle\rangle \in T_D$
3. if $a, b \in T_D$ then $\langle a, b\rangle \in T_D$

In *DFL*, the types $\langle s, e\rangle$ and $\langle s, \langle\langle s, t\rangle, t\rangle\rangle$ take the part of the types e and t from extensional logics. In fact, the *DFL* types stand in a one-to-one corespondence with the extensional *DIL* types. The following recursively defined type shift $^{\uparrow}$ relates the extensional types in T to T_D[12]:

Definition 2.8 (DFL type shift)

1. $^{\uparrow}e \quad = \langle s, e\rangle$
   $^{\uparrow}t \quad = \langle s, \langle\langle s, t\rangle, t\rangle\rangle$
2. $^{\uparrow}\langle a, b\rangle = \langle {}^{\uparrow}a, {}^{\uparrow}b\rangle$, for a and b extensional types

So, in *DFL*, only types are used which are the lift $^{\uparrow}a$ of an extensional type a.

Associated with the above type shift are two operations $\uparrow$ and $\downarrow$ that mediate between the extensional *DIL* types and the corresponding *DFL* types. For any expression ϕ of an extensional type a in *DIL*, $\uparrow\phi$ is an expression of type $^{\uparrow}a$, and for any expression Φ of a *DFL* type $^{\uparrow}a$, $\downarrow\Phi$ is an expression of type a again. The interpre-

10. Notice, however, that, as much as *IL* is dispensible in *MG*, *DIL* is dispensible in *DMG*, and so is *DFL*.
11. With the present definition I deviate from Groenendijk and Stokhof, who use functional types derived from the types e and $\langle\langle s, t\rangle, t\rangle$, and, furthermore, employ derived types $\langle s, a\rangle$. The reason for this deviation is that *DFL*, thus, can be conceived of as an *extensional* dynamic logic.
12. An extensional type is a type in which the type s does not occur. The set of extensional types is defined as the set of types of extensional type theory, i.e., as the smallest set which contains the types e and t and such that if a and b are in that set, then also is $\langle a, b\rangle$.

tation of the type shifting operations is specified in terms of *DIL* expressions in the following simultaneous recursive definition:

Definition 2.9 (DFL type shift (interpretation))

1. $\uparrow\phi_e = {}^\wedge\phi$
   $\downarrow\Phi_{\uparrow e} = {}^\vee\Phi$
2. $\uparrow\phi_t = {}^\wedge\lambda p\ (\phi \wedge {}^\vee p)$ (p not free in ϕ)
   $\downarrow\Phi_{\uparrow t} = {}^\vee\Phi({}^\wedge\mathtt{true})$
3. $\uparrow\phi_{\langle a,b\rangle} = \lambda x_{\uparrow a}\ \uparrow(\phi(\downarrow x))$ (x not free in ϕ)
   $\downarrow\Phi_{\uparrow\langle a,b\rangle} = \lambda x_a\ \downarrow(\Phi(\uparrow x))$ (x not free in Φ)

where p is a variable of type $\langle s,t\rangle$, and `true` is a constant of type t that is assigned the value 1 (in every state s).

According to this definition, the lift of an expression of type e denotes the intension of that expression. So, for any discourse marker d, $[\![\uparrow d]\!]_M = F(d)$. The lowering of an expression of type ${}^\uparrow t$ gives the expression's extension in the state of evaluation.

The most important clause in the above definition is the second one, which sends a *DIL* formula, an expression of type t, to *DFL* formulas, expressions of the type of context change potentials. The lift of a *DIL* formula ϕ denotes a property of propositions, viz., the property of being true in conjunction with ϕ. The application of the extension of $\uparrow\phi$ to the intension of a *DIL* formula ψ is equivalent with the conjunction of ϕ with ψ:

$$({}^\vee\uparrow\phi)({}^\wedge\psi) \Leftrightarrow ({}^\vee{}^\wedge\lambda p\ (\phi \wedge {}^\vee p))({}^\wedge\psi) \Leftrightarrow \phi \wedge \psi$$

Clearly, the lift of a *DIL* formula ϕ denotes a vacuous context change potential. This function holds of a state s and a proposition p iff ϕ is true in s and p holds of s. So, this function involves no change in the context of interpretation of propositional argument expressions. The operator $\uparrow$ merely serves to lift static expressions to the level of types at which *DMG* context change obtains.

The lowering of an expression Φ of type ${}^\uparrow t$ consists in the application of its extension to the necessarily true proposition. Since the kind of context change potentials we will be dealing with all express the property of propositions of being true in a state s' satisfying certain further conditions, the closure $\downarrow\Phi$ of Φ simply asserts that ${}^\wedge\mathtt{true}$ is true in such a state s' and this comes down to the statement that a state s' of that kind exists. The closure thus gives us Φ's truth conditions, and it robs Φ of any dynamic impact it may have. More generally, if Φ is of any type ${}^\uparrow a$, then $\downarrow\Phi$ specifies the static content of Φ, and if Φ is of type ${}^\uparrow t$ and $\downarrow\Phi$ is true (with respect to M, s, and g), then also Φ is said to be true (with respect to M, s and g).

An easy induction proves the following fact:

Fact 2.5 ($\downarrow\uparrow$-elimination)

- $\downarrow\uparrow\phi \Leftrightarrow \phi$

Compare this with $^{\vee\wedge}$-elimination. What does not hold in general is that $\uparrow\downarrow\Phi \Leftrightarrow \Phi$, as similarly $^{\wedge\vee}\alpha \Leftrightarrow \alpha$ does not hold in general. The lift $\uparrow\downarrow\Phi$ of the lowering of Φ deprives Φ of its context change potential, and it will therefore be referred to as the *static closure* of Φ.

DFL: syntax and semantics

I now turn to the language of *DFL*, which is specified by the following definition[13]:

Definition 2.10 (DFL syntax)
For any extensional type a, the set $WE_{D,\uparrow a}$ of well-formed *DFL*-expressions of type $\uparrow a$ is the smallest set such that for all extensional types a and b:

1. If $\alpha \in CON_a^{IL}$ then $\uparrow\alpha \in WE_{D,\uparrow a}$
   If $\alpha \in VAR_{\uparrow a}^{IL}$ then $\alpha \in WE_{D,\uparrow a}$, where a is an extensional type
2. If $\beta \in WE_{D,\uparrow\langle a,b\rangle}$ and $\alpha \in WE_{D,\uparrow a}$ then $\beta(\alpha) \in WE_{D,\uparrow b}$
   If $\beta \in WE_{D,\uparrow b}$ and $x \in VAR_{\uparrow a}$ then $\lambda x\ \beta \in WE_{D,\uparrow\langle a,b\rangle}$
3. If $\alpha, \beta \in WE_{D,\uparrow a}$ then $(\alpha \,\tilde{=}\, \beta) \in WE_{D,\uparrow t}$
4. If $\Phi \in WE_{D,\uparrow t}$ then $\sim\Phi \in WE_{D,\uparrow t}$
   If $\Phi \in WE_{D,\uparrow t}$ and $d \in DM$ then $\mathcal{E}d\beta \in WE_{D,\uparrow t}$ (similarly for $d \in VAR_{\uparrow a}$)
   If $\Phi, \Psi \in WE_{D,\uparrow t}$ then $[\Phi\ ;\ \Psi] \in WE_{D,\uparrow t}$

In what follows, $\mathcal{A}d\Phi$, $[\Phi \Rightarrow \Psi]$ and $[\Phi$ **or** $\Psi]$ abbreviate $\sim\mathcal{E}d\sim\Phi$, $\sim[\Phi\ ;\ \sim\Psi]$ and $[\sim\Phi \Rightarrow \Psi]$, respectively.)

The *DFL* language is built up from the lift of extensional *DIL* constants and from variables of the *DFL* types by means of application, abstraction, and dynamic counterparts of identity, negation, existential quantification and conjunction. So, virtually this is a language of an extensional logic. The difference resides in the association of the expressions of this language with raised types, and in the possibility to quantify over the values of discourse markers.

As was already said above, the semantics of the *DFL* language is stated in terms of the language of *DIL*.[14] Variables, application structures and λ-terms are interpreted as in *DIL*, and the interpretation of lifted *DIL* constants is specified in terms of *DIL* expressions in definition 2.9. The interpretation of the *DFL* counterparts of identity, negation, existential quantification and conjunction is defined as follows (again, p is a variable of type $\langle s, t\rangle$):

Definition 2.11 (DFL semantics)

1. $\alpha \,\tilde{=}\, \beta\ = \uparrow(\downarrow\alpha = \downarrow\beta)$

---

13. As was indicated above, in Groenendijk and Stokhof's presentation of *DMG* the operators $\sim$, $\mathcal{E}d$ and ; merely serve to abbreviate *DIL* expressions.
14. In the sequel I will also refer to the interpretation of *DFL* expressions by means of *DIL* terms. This is not a strictly correct way to go about, but this way of talking about things simplifies matters greatly, and it is not likely to give rise to confusion.

2. $\sim\Phi \quad = \uparrow\neg\downarrow\Phi$
3. $\mathcal{E}d\Phi \quad = {^\wedge}\lambda p\ \exists d {^\vee}\Phi(p)$ (p not free in Φ)
4. $[\Phi\,;\Psi] = {^\wedge}\lambda p\ ({^\vee}\Phi({^\wedge}({^\vee}\Psi(p))))$ (p not free in Φ or Ψ)

In *DFL* $\tilde{=}$ does not denote strict identity. The truth of a *DFL* identity statement in terms of $\tilde{=}$ merely requires identity of the static contents of the identified expressions, not of their context change potential. The reason is that *DMG* (or *DFL*, for that matter) is not concerned with statements about dynamic denotations, but with an account of the realization of context change potentials in discourse. An identity statement in *DFL* therefore states static identity, a statement which by means of the lifting operation $\uparrow$ is transposed to the level of context change potentials.

Something similar goes for negation. The *DFL* negation of a *DFL* formula Φ involves the classical negation $\neg\downarrow\Phi$ of the truth-conditional content of Φ. This classical negation is lifted to the level of context change potentials by means of $\uparrow$, in order for $\sim\Phi$ to denote an object of the type of context change potentials again. Notice that the *DFL* negation of Φ, thus, deprives Φ of any dynamic impact it may have. This seems to accord with the observation that, in general, anaphoric pronouns can not refer back to indefinite noun phrases (translated by means of dynamic existential quantifiers), when the noun phrases stand in the scope of a negation. For instance, consider the following examples:

$^?$It is not the case that a man walks in the park. He whistles.
$^?$No man walks in the park. He whistles.

It is hardly possible to interpret the pronouns in the second sentences of these two examples as being anaphorically related to the noun phrases *a man* and *no man* in the first sentences of the examples. (However, in chapter 2 we will come across some examples in which indefinites in the scope of a negation remain available for subsequent anaphoric reference.)

With the existential quantifier the dynamics comes in. A dynamic existentially quantified formula involves abstraction over a propositional variable the value of which can be evaluated in a context different from the context of evaluation. For instance, consider the *DFL* formula $\mathcal{E}d\uparrow\texttt{man}(\uparrow d)$. The embedded formula $\uparrow\texttt{man}(\uparrow d)$, which is equivalent with $\uparrow(\texttt{man}(d)) \Leftrightarrow {^\wedge}\lambda p\ (\texttt{man}(d) \wedge {^\vee}p)$, denotes the (vacuous) context change potential which holds of a state s and a proposition p iff the value of d in s is a man and p holds of s. By means of the dynamic existential quantifier, this function can be turned into a genuine context change potential. The interpretation of the formula $\mathcal{E}d\uparrow\texttt{man}(\uparrow d)$ is defined to be ${^\wedge}\lambda p\ \exists d({^\vee}\uparrow\texttt{man}(\uparrow d)(p))$ which is equivalent with ${^\wedge}\lambda p\ \exists d(\texttt{man}(d)\wedge{^\vee}p)$. This expression denotes the context change potential which holds of a state s and a proposition p iff p holds of a state s' which (at most) differs from s with respect to the value of d and such that the value d in s' is a man.

The dynamic conjunction of Φ and Ψ involves the intension of the intensional composition of the extensions of the denoted context change potentials. So, consider the conjunction $[\mathcal{E}d{\uparrow}\texttt{man}({\uparrow}d)\,;{\uparrow}\texttt{whistle}({\uparrow}d)]$ of the existentially quantified formula above with the atomic formula ${\uparrow}\texttt{whistle}({\uparrow}d)$ (which equals ${\uparrow}(\texttt{whistle}(d))$). The interpretation of this *DFL* conjunction is given by the following *DIL* formula, and subsequent reductions of it:

$$
\begin{aligned}
&{}^\wedge\lambda p\ {}^\vee\mathcal{E}d{\uparrow}(\texttt{man}(d))({}^\wedge({}^\vee{\uparrow}(\texttt{whistle}(d))(p)))\\
&{}^\wedge\lambda p\ {}^\vee\mathcal{E}d{\uparrow}(\texttt{man}(d))({}^\wedge({}^{\vee\wedge}\lambda p\ (\texttt{whistle}(d) \wedge {}^\vee p)(p)))\\
&{}^\wedge\lambda p\ {}^\vee\mathcal{E}d{\uparrow}(\texttt{man}(d))({}^\wedge(\texttt{whistle}(d) \wedge {}^\vee p))\\
&{}^\wedge\lambda p\ ({}^{\vee\wedge}\lambda p\ \exists d(\texttt{man}(d) \wedge {}^\vee p))({}^\wedge(\texttt{whistle}(d) \wedge {}^\vee p))\\
&{}^\wedge\lambda p\ \exists d(\texttt{man}(d) \wedge \texttt{whistle}(d) \wedge {}^\vee p)
\end{aligned}
$$

This formula denotes the context change potential which holds of a state s and a proposition p iff p holds of a state s' which (at most) differs from s with respect to the value of d and such that the value d in s' is a man who whistles.

DFL: semantic properties

The notion of dynamic conjunction defined above is associative. Furthermore, the dynamic existential quantifier 'associates' with a conjunction sign to the right of it:

Fact 2.6 (Associativity (1))

- $[[\Phi\,;\Psi]\,;\Upsilon] \Leftrightarrow [\Phi\,;[\Psi\,;\Upsilon]]$
- $[\mathcal{E}d\Phi\,;\Psi] \quad \Leftrightarrow \mathcal{E}d[\Phi\,;\Psi]$

The fact that the existential quantifier in *DFL* associates with conjunction lies at the heart of *DMG*'s treatment of intersentential anaphora. For instance (as we will see in more detail below) the sequence of sentences *A man walks. He talks* is associated with the following (reduced) translation:

$$[\mathcal{E}d[{\uparrow}(\texttt{man}(d))\,;{\uparrow}(\texttt{walk}(d))]\,;{\uparrow}(\texttt{whistles}(d))]$$

Using fact 2.6, this *DFL* formula is equivalent with the following formula:

$$\mathcal{E}d[{\uparrow}(\texttt{man}(d))\,;[{\uparrow}(\texttt{walk}(d))\,;{\uparrow}(\texttt{whistles}(d))]]$$

In other words, the sequence turns out equivalent with the sentence *There is a man who walks and whistles.*

Another typically dynamic fact is that dynamic conjunction is not commutative:

Fact 2.7 (Non-commutativity)

- $[\Phi\,;\Psi] \not\Leftrightarrow [\Psi\,;\Phi]$

For instance, the *DMG* translation $[\mathcal{E}d[{\uparrow}(\texttt{man}(d))\ ;\ {\uparrow}(\texttt{walk}(d))]\ ;\ {\uparrow}(\texttt{whistles}(d))]$ of a sequence *A man walks. He whistles* is not equivalent with the translation $[{\uparrow}(\texttt{whistles}(d))\ ;\ \mathcal{E}d[{\uparrow}(\texttt{man}(d))\ ;\ {\uparrow}(\texttt{walk}(d))]]$ of the sequence *He whistles. A man walks*. The reason may be clear. In the first sequence the pronoun *he* is bound by the indefinite noun phrase *a man*, whereas in the second it is not.

By the definition of $\mathcal{A}d$ and $\Rightarrow$, and employing fact 2.6, the following facts are easily proved:

Fact 2.8 (Associativity (2))
- $[[\Phi\,;\Psi] \Rightarrow \Upsilon] \Leftrightarrow [\Phi \Rightarrow [\Psi \Rightarrow \Upsilon]]$
- $[\mathcal{E}d\Phi \Rightarrow \Psi] \quad \Leftrightarrow \mathcal{A}d[\Phi \Rightarrow \Psi]$

This fact shows that the *DFL* notion of implication is, what is adequately called, internally dynamic. An existential quantifier in the antecedent of an implication may bind discourse markers in the consequent, this, with universal force. Thus, *DMG* is able to account for donkey sentences of the conditional variety. For instance, the sentence *If a farmer owns a donkey, he beats it* has the following (reduced) translation in *DMG*:

$$[\mathcal{E}d[{\uparrow}(\texttt{farm}(d))\,;\mathcal{E}d'[{\uparrow}(\texttt{donk}(d'))\,;{\uparrow}(\texttt{own}(d)(d'))]] \Rightarrow {\uparrow}(\texttt{beat}(d)(d'))]$$

Employing fact 2.8, this formula can be seen to be equivalent with the following formula:

$$\mathcal{A}d[{\uparrow}(\texttt{farm}(d)) \Rightarrow \mathcal{A}d'[{\uparrow}(\texttt{donk}(d')) \Rightarrow [{\uparrow}(\texttt{own}(d)(d')) \Rightarrow {\uparrow}(\texttt{beat}(d)(d'))]]]$$

This formula is true iff every farmer beats every donkey he owns. In fact, these are the truth conditions associated with the donkey sentence within the framework of *DRT*.

Implication in *DFL*, although internally dynamic, is externally static, as is *DFL*'s universal quantifier. This is a consequence of the fact that $[\Phi \Rightarrow \Psi]$ and $\mathcal{A}d\Phi$ are defined as $\sim[\Phi\,;\sim\Psi]$ and $\sim\mathcal{E}d\sim\Phi$, respectively, and the fact that negation is static. For this reason, indefinite noun phrases which figure in a conditional sentence or which occur in a sentence in the scope of a universal quantifier are unable to bind pronouns in subsequent sentences. Again, see chapter 2 for examples for which one does need (externally) dynamic notions of implication and universal quantification. Notice that the *DFL* disjunction $[\Phi\ \textbf{or}\ \Psi]$, which is defined as $[\sim\Phi \Rightarrow \Psi]$, is both internally and externally static, so no noun phrases in Φ or Ψ are available for subsequent anaphoric coreference, nor can pronouns in Ψ be anaphorically related to noun phrases in Φ.

The static nature of negation also blocks some standard equivalences. For instance, $\mathcal{E}d\Phi$ and $\sim\mathcal{A}d\sim\Phi$ are not equivalent, neither are $[\Phi\,;\Psi]$ and $\sim[\Phi \Rightarrow \sim\Psi]$. However, as is to be expected, these formulas do have the same truth-conditional content, i.e., ${\downarrow}\mathcal{E}d\sim\Phi \Leftrightarrow {\downarrow}\sim\mathcal{A}d\Phi$ and ${\downarrow}[\Phi\,;\Psi] \Leftrightarrow {\downarrow}\sim[\Phi \Rightarrow \sim\Psi]$. So, although the formulas have different dynamic properties, as far as truth conditions are concerned they are related in the usual way.

All *DFL* formulas denote properties of propositions, which, equivalently, can be conceived of as functions from states to sets of sets of states. Adopting the terminology of the theory of generalized quantifiers they denote functions from states to generalized quantifiers over states. An important property of *DFL* is that the extensions of all of its sentences (formulas without free variables) are so-called *upward monotonic* quantifiers over states. A quantifier is called upward monotonic if it contains all supersets of any set it contains:

$$\mathcal{Q} \text{ is upward monotonic iff } \forall P, Q \text{ if } P \subseteq Q \text{ and } P \in \mathcal{Q} \text{ then } Q \in \mathcal{Q}$$

For any *DFL* sentence Φ, the extension of Φ is always upward monotonic. So, if $[\![p]\!] \subseteq [\![q]\!]$, that is, if $\forall s'$ if $[\![p]\!](s') = 1$ then $[\![q]\!](s') = 1$, then if $[\![{}^\vee\Phi]\!]([\![p]\!]) = 1$ then $[\![{}^\vee\Phi]\!]([\![q]\!]) = 1$.

Since the extension of any *DFL* formula Φ is upward monotonic, sentence sequencing always involves a strengthening of truth-conditions:

Fact 2.9
- $\downarrow[\Phi\,;\Psi]$ entails $\downarrow\Phi$ if Φ is upward monotonic

The upward monotonicity of *DFL* accords with the intuition that the processing of subsequent sentences in a discourse constitutes an ongoing process of information update. This property will turn out to be of great importance when we turn to the issue of dynamic negation in chapter 2.

DFL reductions

Before we turn to the *DMG* fragment of natural language, it is useful to state some reduction facts. In the first place, λ-conversion is allowed under the conditions for λ-conversion in an *extensional* logic:

Fact 2.10 (λ-conversion)
- $(\lambda x\ \beta)(\alpha) \Leftrightarrow [\alpha/x]\beta$ if all free variables in α are free for x in β

For λ-conversion to be allowed, it need not be separately required that α is intensionally closed, since, as is easily seen, all *DFL* expressions *are* intensionally closed.

The following equivalences enable us to replace dynamic operators by their static counterparts when we determine the static contents of (dynamically interpreted) *DFL* expressions by means of $\downarrow$:

Fact 2.11 (Arrow elimination)
- $\downarrow\uparrow\phi \quad \Leftrightarrow \phi$
- $(\uparrow\phi)(\Psi) \quad \Leftrightarrow \uparrow(\phi(\downarrow\Psi))$
- $\downarrow(\alpha \doteq \beta) \quad \Leftrightarrow (\downarrow\alpha = \downarrow\beta)$
  $\downarrow{\sim}\Phi \quad \Leftrightarrow \neg\downarrow\Phi$
  $\downarrow\mathcal{E}d\Phi \quad \Leftrightarrow \exists d\downarrow\Phi$
  $\downarrow\mathcal{A}d\Phi \quad \Leftrightarrow \forall d\downarrow\Phi$
- $\downarrow[\uparrow\phi\,;\Psi] \quad \Leftrightarrow \phi \wedge \downarrow\Psi$
  $\downarrow[\uparrow\phi \Rightarrow \Psi] \quad \Leftrightarrow \phi \rightarrow \downarrow\Psi$

$\downarrow[\uparrow\phi$ **or** $\Psi] \quad \Leftrightarrow \phi \vee \downarrow\Psi$

The equivalences in fact 2.11 can be used to transform the closure of most dynamic *DMG* formulas into *DIL* formulas. In many cases we may push the closure operator $\downarrow$ over *DMG* operators, which are replaced by their static counterparts then. The closure operator collapses when it confronts a lifted atomic formula. (Notice that the reduction rules in 2.11 do not enable us to reduce all *DMG* expressions. In section 3 of chapter 2 a complete reduction system is given.)

2.3 A DMG fragment of natural language

We may now turn to the construction of a dynamic Montague style fragment of natural language. Compared to the original *PTQ* model (Montague [1973]) it is a rather poor fragment, since the aim is only to show how dynamic interpetation fits in.[15] Furthermore, the following things change. First, of all (constituent-)expressions the type is raised in accordance with the dynamic type shift. Second, the constants are replaced by their raised counterparts and sentential operators by their dynamic counterparts. Finally, in the translation of pronouns and quantifying noun phrases I use indexed discourse markers. Indices are used to indicate anaphoric relationships among constituents.

DMG syntax

The *DMG* syntax of a fragment of natural language is, basically, a categorial syntax with has an additional set of syntactic operations. The syntax consists of the following three components:

1. a set of categories, which is constructed from a limited set of basic categories
2. a specification of the basic expressions of each category
3. a set of construction rules by means of which compound expressions can be formed

The set of categories is defined as follows:

Definition 2.12 (DMG categories)
The set of categories *CAT* is the smallest set such that:

1. $S, CN, IV \in CAT$
2. If $B, A \in CAT$ then $B/A \in CAT$

15. The *PTQ* model deals with many interesting features of English, such as its apparatus of quantification, intensional verbs and intensional prepositions, the function of the definite article, the nature of ambiguity, and the role of adjectives and adverbs. Therse are disreagrded here. However, some of these features will come up for discussion in due course. In chapter 3, for instance, we will find an (alternative) treatment of quantifier scope due to Hendriks [1988, 1992], in chapter 4 adverbs and tense will be addressed from the dynamic perspective on meaning, and in chapter 5 intensionality is addressed from an epistemic perspective.

In this definition S stands for the category of sentences, CN for that of common noun phrases, and IV for that of intransitive verb phrases. Derived categories are of the form B/A, where A and B can be any category. The category B/A is the category of expressions which together with an expression of category A to their right make up a (compound) expression of category B (modulo some possible syntactic modifications which are neglected here). Some frequently employed derived categories are abbreviated as follows: the category of noun phrases $NP = S/IV$, the category of determiners $Det = NP/CN$, and the category of transitive verb phrases $TV = IV/NP$.

Some categories A are associated with a non-empty set of basic expressions B_A. The following expressions will be employed:

Definition 2.13 (DMG basic expressions)

- B_{CN} = {*man, woman, donkey, house,* ...}
- B_{IV} = {*walk, whistle,* ...}
- B_{TV} = {*own, love,* ...}
- B_{Det} = {a_i, $every_j$, ...}
- B_{NP} = {$John_i$, $Mary_j$, ..., he_i, he_j, ...}

Other expressions can be easily added, as long as they can, and are, treated in the same way as the above expressions are treated below.

Compound expressions can be formed by means of the following construction rules:[16]

Definition 2.14 (DMG construction rules)
The set P_A of compound expressions of category A, for any category $A \in CAT$, is the smallest set such that for all categories $A, B \in CAT$:

- $B_A \subseteq P_A$
- if α is an expression of category A and β is an expression of category B/A, then $[\beta\ \alpha]$ is an expression of category B (functional application)
- if ϕ is an expression of category S, then $[Not\ \phi]$ is an expression of category S (sentence negation)
- if ϕ and ψ are expressions of category S, then $[\phi\ and\ \psi]$ is an expression of category S (conjunction)
- if ϕ and ψ are expressions of category S, then $[If\ \phi,\ then\ \psi]$ is an expression of category S (conditionals)
- if α is an expression of category CN and β is an expression of category IV, then $[\alpha\ who\ \beta]$ is an expression of category CN (relative clause formation)

16. With this definition all syntactic clauses having to do with gender, case, tense, and the like are neglected. Thus, all Montague's rules of application can be phrased in one rule.

Apart from the rule of application, there are rules by means of which we can construct negated sentences, conjunctions, conditionals and relative clauses.[17] The brackets enclosing compound expressions will be omitted whenever irrelevant (which is most of the time).

DMG translation

I now turn to the semantics of the *DMG* fragment. As has already been said, the interpretation of expressions of the fragment is given by their translation into expressions of the language of *DFL*. The semantics is spelled out in analogy with the set-up of the syntax. I will define:

1. a function f mapping *DMG* categories to *DFL* types
2. the translation (of type $f(A)$) of the basic expressions of category A
3. the translation of compound expressions in terms of the translations of their constituent expressions

The *DMG* category to type assignment differs from that in the *PTQ* model:

Definition 2.15 (DMG category to type assignment)

1. $f(S) = {\uparrow}t$
   $f(CN) = {\uparrow}\langle e,t\rangle$
   $f(IV) = {\uparrow}\langle e,t\rangle$
2. $f(B/A) = \langle f(A), f(B)\rangle$

The basic difference is that the type associated with the category of sentences is the type of context change potentials.

I now give some examples of the translations of the basic expressions of the *DMG* fragment above. The main difference with the *PTQ* model resides in the types of the variables used, and in the fact that instead of *IL* operators, corresponding *DFL* operators are used. In the following definition x and y are variables of type ${\uparrow}e$, P and Q variables of type ${\uparrow}\langle e,t\rangle$, and T of type ${\uparrow}\langle\langle e,t\rangle,t\rangle$; $\mathtt{j}$ is a constant of type e, $\mathtt{man}$ and $\mathtt{walk}$ are constants of type $\langle e,t\rangle$, and $\mathtt{love}$ of type $\langle e,\langle e,t\rangle\rangle$; the d_i are discourse markers:

Definition 2.16 (DMG basic expressions)

- $man_{CN} \rightsquigarrow {\uparrow}\mathtt{man}$
- $walk_{IV} \rightsquigarrow {\uparrow}\mathtt{walk}$
- $own_{TV} \rightsquigarrow \lambda T \lambda x\ T(\lambda y\ {\uparrow}\mathtt{own}(y)(x))$
- $a_i \rightsquigarrow \lambda P \lambda Q\ \mathcal{E} d_i[P({\uparrow}d_i)\ ;\ Q({\uparrow}d_i)]$
- $every_j \rightsquigarrow \lambda P \lambda Q\ \mathcal{A} d_i[P({\uparrow}d_i) \Rightarrow Q({\uparrow}d_i)]$
- $he_i \rightsquigarrow \lambda Q\ Q({\uparrow}d_i)$

17. It may be noticed that the rule of relative clause formation does not generate relative clauses like *man whom Mary loves*. This is not a principled limitation, but one for simplicity's sake only.

- $John_i \rightsquigarrow \lambda Q\ \mathcal{E}d_i[(\uparrow \mathtt{j} \doteq \uparrow d_i)\,;Q(\uparrow d_i)]$

I now turn to the definition of the translation of compound constructions. The translation of compound expressions is completely determined by the construction rules by means of which the compounds are constructed and the translations of the constituent expressions from which they are formed. The translations are also like the *PTQ* counterparts but for the use of T_D-types and the occurrence of *DFL* operators. In the following definition I have labelled the expressions with subscripts indicating the category to which they belong. Furthermore, for any expression α I use α' to indicate the translation of α.

Definition 2.17 (DMG construction rules)

1. Functional application: $(\beta_{B/A}\ \alpha_A)_B \rightsquigarrow \beta'(\alpha')$
2. Sentence negation: $(Not\ \sigma_S)_S \rightsquigarrow {\sim}\sigma'$
3. Sequencing: $(\sigma_S\ .\ \tau_S)_S \rightsquigarrow [\sigma'\,;\tau']$
4. Conditionals: $(If\ \sigma_S,\ then\ \tau_S)_S \rightsquigarrow [\sigma' \Rightarrow \tau']$
5. Relative clauses: $(\alpha_{CN}\ who\ \beta_{IV})_{CN} \rightsquigarrow \lambda x\ [\alpha'(x)\,;\beta'(x)]$

Interpretation in dynamic Montague grammar is illustrated by the *DMG* treatment of three examples. I review three examples discussed in Groenendijk and Stokhof [1990a]. As is said above, indices indicate intended anaphoric relationships.

A_i man walks. He_i talks.

This is a slightly simplified version of our earlier example. By means of functional application we get the following translation of the first sentence:

$(\lambda P \lambda Q\ \mathcal{E}d_i[P(\uparrow d_i)\,;Q(\uparrow d_i)])(\uparrow\mathtt{man})(\uparrow\mathtt{walk}) \Leftrightarrow$ (fact 2.10)
$\mathcal{E}d_i[\uparrow\mathtt{man}(\uparrow d_i)\,;\uparrow\mathtt{walk}(\uparrow d_i)] \Leftrightarrow$ (fact 2.11)
$\mathcal{E}d_i[\uparrow(\mathtt{man}(d_i))\,;\uparrow(\mathtt{walk}(d_i))]$

The second sentence is associated with the following translation:

$(\lambda Q\ Q(\uparrow d_i))(\uparrow\mathtt{talk}) \Leftrightarrow$ (fact 2.10)
$\uparrow\mathtt{talk}(\uparrow d_i) \Leftrightarrow$ (fact 2.11)
$\uparrow(\mathtt{talk}(d_i))$

Employing the above translations, the sequence of the two sentences is translated as follows:

$[\mathcal{E}d_i[\uparrow(\mathtt{man}(d_i))\,;\uparrow(\mathtt{walk}(d_i))]\,;\uparrow(\mathtt{talk}(d_i))]$

This formula has the following truth conditions:

$\downarrow[\mathcal{E}d_i[\uparrow(\mathtt{man}(d_i))\,;\uparrow(\mathtt{walk}(d_i))]\,;\uparrow(\mathtt{talk}(d_i))] \Leftrightarrow$ (fact 2.6)
$\downarrow\mathcal{E}d_i[\uparrow(\mathtt{man}(d_i))\,;[\uparrow(\mathtt{walk}(d_i))\,;\uparrow(\mathtt{talk}(d_i))]] \Leftrightarrow$ (fact 2.11)
$\exists d_i(\mathtt{man}(d_i) \wedge \mathtt{walk}(d_i) \wedge \mathtt{talk}(d_i)) \Leftrightarrow$ (fact 2.4)

$$\exists x(\mathtt{man}(x) \wedge \mathtt{walk}(x) \wedge \mathtt{talk}(x))$$

Here we see that the pronoun *he*$_i$ is bound by the quantifying noun phrase a$_i$ *man*, even though its translation does not occur in the immediate syntactic scope of the translation of the noun phrase.

If a$_i$ *man walks, he*$_i$ *talks*

The second example is an implication built up from the two sentences the conjunction of which constitute the first example. Using the translation rule for conditional sentences, we get the following translation of this example:

$$\mathcal{E}d_i[{\uparrow}(\mathtt{man}(d_i))\,;{\uparrow}(\mathtt{walk}(d_i))] \Rightarrow {\uparrow}(\mathtt{talk}(d_i))$$

The truth-conditions of this formula can be computed as follows:

$${\downarrow}[\mathcal{E}d_i[{\uparrow}(\mathtt{man}(d_i))\,;{\uparrow}(\mathtt{walk}(d_i))] \Rightarrow {\uparrow}(\mathtt{talk}(d_i))] \Leftrightarrow \text{(fact 2.8)}$$
$${\downarrow}\mathcal{A}d_i[[{\uparrow}(\mathtt{man}(d_i))\,;{\uparrow}(\mathtt{walk}(d_i))] \Rightarrow {\uparrow}(\mathtt{talk}(d_i))] \Leftrightarrow \text{(fact 2.8)}$$
$${\downarrow}\mathcal{A}d_i[{\uparrow}(\mathtt{man}(d_i)) \Rightarrow [{\uparrow}(\mathtt{walk}(d_i)) \Rightarrow {\uparrow}(\mathtt{talk}(d_i))]] \Leftrightarrow \text{(fact 2.11)}$$
$$\forall d_i(\mathtt{man}(d_i) \to (\mathtt{walk}(d_i) \to \mathtt{talk}(d_i))) \Leftrightarrow \text{(fact 2.4)}$$
$$\forall x(\mathtt{man}(x) \to (\mathtt{walk}(x) \to \mathtt{talk}(x)))$$

By combining the two sentential clauses in a conditional sentence, the existential quantifier in the antecedent turns out to bind the pronoun in the consequent, although the two clauses are assigned a meaning of their own. Notice that the existential quantifier in this example has gained universal force.

Every$_i$ *farmer who owns* a$_j$ *donkey beats it*$_j$

Although the previous example has already shown how donkey-type anaphora are dealt with in *DMG*, I shall also show how the classical donkey sentence *Every farmer who owns a donkey, beats it* is derived. By means of application and λ-conversion, the following translation of the intransitive verb phrase *owns* a$_j$ *donkey* results:

$$(\lambda T\lambda x\ T(\lambda y{\uparrow}\mathtt{own}(y)(x)))(\lambda P\lambda Q\ \mathcal{E}d_j[P({\uparrow}d_j)\,;Q({\uparrow}d_j)]({\uparrow}\mathtt{donk})) \Leftrightarrow$$
$$(\lambda T\lambda x\ T(\lambda y{\uparrow}\mathtt{own}(y)(x)))(\lambda Q\ \mathcal{E}d_j[{\uparrow}\mathtt{donk}({\uparrow}d_j)\,;Q({\uparrow}d_j)]) \Leftrightarrow$$
$$\lambda x\ (\lambda Q\ \mathcal{E}d_j[{\uparrow}\mathtt{donk}({\uparrow}d_j)\,;Q({\uparrow}d_j)])(\lambda y{\uparrow}\mathtt{own}(y)(x)) \Leftrightarrow$$
$$\lambda x\ \mathcal{E}d_j[{\uparrow}\mathtt{donk}({\uparrow}d_j)\,;{\uparrow}\mathtt{own}({\uparrow}d_j)(x)]$$

The compound common noun *farmer who owns* a$_j$ *donkey* has the following (reduced) translation:

$$\lambda x\ [{\uparrow}\mathtt{farm}(x)\,;\mathcal{E}d_j[{\uparrow}\mathtt{donk}({\uparrow}d_j)\,;{\uparrow}\mathtt{own}({\uparrow}d_j)(x)]]$$

The intransitive verb phrase *beats it*$_j$ gets the following translation, which can be reduced again by means of λ-conversion:

$$(\lambda T\lambda x\ T(\lambda y{\uparrow}\mathtt{beat}(y)(x)))(\lambda Q\ Q({\uparrow}d_j)) \Leftrightarrow$$
$$\lambda x\ (\lambda Q\ Q({\uparrow}d_j))(\lambda y{\uparrow}\mathtt{beat}(y)(x)) \Leftrightarrow$$
$$\lambda x\ {\uparrow}\mathtt{beat}({\uparrow}d_j)(x)$$

The determiner *every*$_i$ combines, first, with the common noun phrase *farmer who owns* a$_j$ *donkey*, and, next, with the intransitive verb phrase *beats it*$_j$. Two applications and some λ-conversions then give us the following translation of the whole sentence:

$$\mathcal{A}d_i[[\uparrow\mathtt{farm}(\uparrow d_i)\,;\mathcal{E}d_j[\uparrow\mathtt{donk}(\uparrow d_j)\,;\uparrow\mathtt{own}(\uparrow d_j)(\uparrow d_i)]] \Rightarrow \uparrow\mathtt{beat}(\uparrow d_j)(\uparrow d_i)]$$

The sentence is assigned the following truth-conditions:

$$\begin{array}{l}
\downarrow\mathcal{A}d_i[[\uparrow\mathtt{farm}(\uparrow d_i)\,;\mathcal{E}d_j[\uparrow\mathtt{donk}(\uparrow d_j)\,;\uparrow\mathtt{own}(\uparrow d_j)(\uparrow d_i)]] \Rightarrow \uparrow\mathtt{beat}(\uparrow d_j)(\uparrow d_i)] \Leftrightarrow \\
\downarrow\mathcal{A}d_i[[\uparrow(\mathtt{farm}(d_i))\,;\mathcal{E}d_j[\uparrow(\mathtt{donk}(d_j))\,;\uparrow(\mathtt{own}(d_j)(d_i))]] \Rightarrow \uparrow(\mathtt{beat}(d_j)(d_i))] \Leftrightarrow \\
\downarrow\mathcal{A}d_i[\uparrow(\mathtt{farm}(d_i)) \Rightarrow [\mathcal{E}d_j[\uparrow(\mathtt{donk}(d_j))\,;\uparrow(\mathtt{own}(d_j)(d_i))] \Rightarrow \uparrow(\mathtt{beat}(d_j)(d_i))]] \Leftrightarrow \\
\downarrow\mathcal{A}d_i[\uparrow(\mathtt{farm}(d_i)) \Rightarrow \mathcal{A}d_j[[\uparrow(\mathtt{donk}(d_j))\,;\uparrow(\mathtt{own}(d_j)(d_i))] \Rightarrow \uparrow(\mathtt{beat}(d_j)(d_i))]] \Leftrightarrow \\
\downarrow\mathcal{A}d_i[\uparrow(\mathtt{farm}(d_i)) \Rightarrow \mathcal{A}d_j[\uparrow(\mathtt{donk}(d_j)) \Rightarrow [\uparrow(\mathtt{own}(d_j)(d_i)) \Rightarrow \uparrow(\mathtt{beat}(d_j)(d_i))]]] \\
\Leftrightarrow \forall d_i(\mathtt{farm}(d_i) \rightarrow \forall d_i(\mathtt{donk}(d_j) \rightarrow (\mathtt{own}(d_j)(d_i) \rightarrow \mathtt{beat}(d_j)(d_i)))) \\
\Leftrightarrow \forall x(\mathtt{farm}(x) \rightarrow \forall y(\mathtt{donk}(y) \rightarrow (\mathtt{own}(y)(x) \rightarrow \mathtt{beat}(y)(x))))
\end{array}$$

(By means of the facts 2.11, 2.8 (three times), 2.11 and 2.4, respectively.) The example is in fact assigned the same truth conditions as in *DRT* but in a fully compositional fashion. Notice that the indefinite noun phrase *a donkey* in the restriction of the universal quantifier *every* turns out to bind the pronoun *it* in the quantifier's nuclear scope, this with universal force.

The last example concludes the exposition of the *DMG* fragment. We have seen that the phenomenon of cross-sentential anaphora as we find in donkey sentences can be treated in *DMG* in an adequate and completely compositional way. In fact, what distinguishes *DMG* from *MG* is, basically, the associativity of the dynamic existential quantifier and the non-commutativity of conjunction. This means that *DMG* is indeed a semantic theory that unifies important insights from *MG* and *DRT*.

Chapter 2

Dynamic negation

In [1990a] Groenendijk and Stokhof reformulate basic aspects of discourse representation theory within dynamic Montague grammar and give a fully compositional treatment of anaphoric relationships holding between pronouns and indefinite noun phrases. Characteristic feature of the logic they use is that two of its three primitive operators are dynamic. The existential quantifier is dynamic in the sense that it may bind variables occurring beyond its syntactic scope and conjunction is dynamic since it is associative (as is usual) but not commutative (which is unusual). On the other hand, the third primitive operator, negation, is 'static'. The negation of a formula ϕ closes off the dynamic potential of quantifiers in ϕ and this seems to be the usual effect of natural language negation.

Still, there is a class of examples that involve anaphoric relationships between quantifying noun phrases within the scope of a negation and pronouns occurring outside of it. Therefore, the question naturally arises what a system would or should be like in which all three operators are dynamic.

At the end of [1990a] Groenendijk and Stokhof propose an alternative notion of *dynamic* negation which serves to account for (some of) the anaphoric relationships which obtain across the scope of a negation. In this chapter I discuss Groenendijk and Stokhof's proposal and argue that their notion of dynamic negation is not completely satisfactory. Next I present an alternative system of dynamic Montague grammar with a notion of dynamic negation and I show that the resulting system of dynamic interpretation accounts for the anaphoric relationships studied.

I will proceed in the following way. First, in section 1.1, I discuss Groenendijk and Stokhof's motivation for introducing dynamic negation and the proposal they actually make. Next, in the sections 1.2 and 1.3, I discuss a fundamental problem with the dynamic negation they propose and I formulate three plausible requirements that a notion of dynamic negation must satisfy. In section 2.1 I give an alternative version of *DMG*, referred to as *DMG(2)*, and extend it in section 2.2 with a notion of dynamic negation that satisfies the requirements formulated in section 1.3. I study the logical behaviour of this notion of dynamic negation in some detail and show how it can be applied successfully to some natural language examples. Section 2.3, finally, gives an overview of the distinctive features of *DMG*, *DMG* with dynamic negation, and *DMG(2)* with dynamic negation.

1 Dynamic negation in DMG

In the last section of [1990a] Groenendijk and Stokhof point out that the framework of *DMG* allows for a straightforward extension which enables them to cover a special class of anaphoric dependencies. On the face of it, these dependencies involve a kind of dynamic implication, dynamic universal quantification, or dynamic disjunction, and in *DMG* a quite acceptable definition of these notions can be given in terms of a notion of dynamic negation. The mere possibility of such an extension of *DMG* serves to indicate, they claim, "... that, even restricting ourselves to the first-order level of quantification and anaphoric reference, *DMG* is potentially more than just the sum of *MG* and *DRT*" ([1990a, p. 33]). So much will remain beyond doubt.

However, I will argue in this section that a simple extension of *DMG* with dynamic negation does not provide a principled account of some examples which do not seem to be essentially different from the examples that Groenendijk and Stokhof account for. For a proper analysis of all the examples something more is needed, viz., a slight, but structural, modification of *DMG*. In section 2 I present this modification of *DMG*, and give the required definition of dynamic negation. Here I first consider the special examples that *DMG* can handle and the way in which *DMG* accounts for them.

1.1 Extended dynamics in DMG

Groenendijk and Stokhof discuss the following examples:

(1) It is not the case that John does not own a car. It is red and it is parked in front of the house.
(2) John owns a car. It is red and it is parked in front of the house.
(3) If a client comes in, you treat him politely. You offer him a cup of coffee.
(4) Every player chooses a pawn. He puts it on square one.
(5) Either there is no bathroom here, or it is in a funny place. In any case, it is not on the ground floor.
(6) If there is a bathroom here, it is in a funny place. In any case, it is not on the ground floor.

(Examples like these date back to the seventies, and have been discussed by Gareth Evans, Lauri Karttunen, Barbara Partee, Craige Roberts, Peter Sells, to name a few.)

In example 1 we find a double negation of the sentence *John owns a car*. As appears from the continuation *It is red etc.*, the double negation not only preserves the truth-conditional content of the sentence *John owns a car*, but also its context change potential. The pronoun *it* seems to be bound by the quantifying noun phrase *a car*, even though this noun phrase is in the scope of the double negation. So, disregarding non-truth-conditional effects having to do with topic/focus, etc., the most likely reading of 1 is equivalent to that of 2, without double negation. But, as we saw in the introduction, *DMG*'s negation does not license the law of double negation. The double negation of a formula Φ in *DMG* has the truth-conditions of Φ, but not its context change potential.

Example 3 exhibits a dynamic implication. In this example, the quantifying noun phrase *a client* in the antecedent of the implication not only binds a pronoun in the consequent, with universal force, but also a pronoun in the sentence that follows the implication, again with universal force. However, the *DMG* notion of implication licenses the first kind of binding only. The static character of the negation in terms of which the implication is defined blocks possible anaphoric relationships between noun phrases inside an implication and pronouns outside of it.

Something similar goes for example 4. Since the universal quantifier is defined in terms of the negation of the existential quantifier, it is (externally) static and the noun phrase *every player* therefore has no dynamic potential in *DMG*. Furthermore, for the very same reason the noun phrase *every player* closes off the dynamic potential of the indefinite *a pawn* in its scope. Still, the pronouns *he* and *it* appear to be bound by these two noun phrases.

In example 5 the two occurrences of the pronoun *it* are anaphorically related to the noun phrase *no bathroom*, and these are unaccounted for as yet. In the first place, the noun phrase *no bathroom* itself is static since it is the negation of the noun phrase *a bathroom*. In the second place, even if the noun phrase *no bathroom* had binding potential, the disjunction would block both anaphoric relationships. Since the disjunction is defined in terms of the negation of both disjuncts, it blocks anaphoric relationships between noun phrases in the first disjunct and pronouns in the second, and between noun phrases figuring in any of the disjuncts and pronouns in sentences following the disjunction. So, both anaphors in example 5 remain unexplained.

The last example, 6, is intuitively equivalent to example 5 and also classically equivalent according to the law: $(\phi \rightarrow \psi) \Leftrightarrow (\neg\phi \vee \psi)$. In *DMG* this example is not equivalent to example 5. The difference is that in *DMG* indefinites in the antecedent of a conditional sentence may bind pronouns in the consequent, whereas a disjunction allows no internal bindings. Therefore *DMG* accounts for the first anaphoric relationship in 6, the one between the indefinite *a bathroom* and the pronoun in the consequent of the first sentence. However, as in example 3, the pronoun in the second sentence of 6 remains unaccounted for.

The observations concerning the examples 1–6 suggest that it is the notion of negation that is in need of revision. In the first place, if, as examples 1 and 2 suggest, the double negation of a formula can be completely equivalent to the formula itself, then the mere negation of the formula must retain the formula's context change potential, in some way or other, in order for the double negation of a formula to have the same context change potential as the original formula. In the second place, having a notion of dynamic negation automatically entails having dynamic implication, disjunction and implication as well, that is, as long as these notions remain defined in the usual way in terms of negation, conjunction and existential quantification. In other words, with a notion of dynamic negation we might account for the anaphoric dependencies in examples 3–6, although it remains to be seen, of course, whether the examples are assigned proper truth-conditions in that case. In the third place, if we have a notion of dynamic negation that obeys the law of double negation, the (classical) equivalence of $[\Phi \Rightarrow \Psi]$ and $[\sim\Phi \ \mathbf{or} \ \Psi]$ is restored again, which seems to be required in view of examples 5 and 6.[1]

Dynamic negation

Groenendijk and Stokhof propose an alternative notion of negation that meets the desiderata we found so far. Interestingly, all that seems required is the *standard* definition of negation as complementation:

Definition 1.1 (Dynamic negation)

- $\sim\Phi = {}^\wedge\lambda p \ \neg{}^\vee\Phi(p)$

The old definition of static negation can be obtained from the dynamic one, by taking the static closure of the dynamic negation: $\uparrow\downarrow\sim\Phi \Leftrightarrow \uparrow\neg\downarrow\Phi = \sim_s\Phi$, where $\sim_s$ indicates static negation.

Being complementation, dynamic negation of course obeys the law of double negation:

Fact 1.1 (Double negation)

- $\sim\sim\Phi \Leftrightarrow \Phi$

From this fact it immediately follows that the notion of dynamic negation accounts for the equivalence of the examples 1 and 2. Replacing *DMG*'s static notion of negation by this dynamic one, example 1 is assigned in a completely compositional way the truth-conditions that John does have a car, which is red and parked in front of the house.

1. If Φ equals $\sim\sim\Phi$, $[\Phi \Rightarrow \Psi]$ equals $[\sim\sim\Phi \Rightarrow \Psi]$ which, by definition, equals $[\sim\Phi \ \mathbf{or} \ \Psi]$.

Since the three primitive operations of negation, existential quantification and conjunction are now dynamic, the derived operations of implication, disjunction and universal quantification turn out (internally and externally) dynamic as well. This has the following consequences. In the first place we have recovered the full interdefinability of the quantifiers and of the implication and the disjunction:

Fact 1.2 (Interdefinability)

- $\mathcal{E}d{\sim}\Phi \quad \Leftrightarrow {\sim}\mathcal{A}d\Phi$
- $[\Phi \Rightarrow \Psi] \Leftrightarrow [{\sim}\Phi$ **or** $\Psi]$

In the second place, the universal quantifier, implication, and disjunction associate with conjunction:

Fact 1.3 (Extended associativity)

- $[\mathcal{A}d\Phi\,;\Psi] \qquad \Leftrightarrow \mathcal{A}d[\Phi\,;\Psi]$
- $[[\Phi \Rightarrow \Psi]\,;\Upsilon] \quad \Leftrightarrow [\Phi \Rightarrow [\Psi\,;\Upsilon]]$
- $[[\Phi$ **or** $\Psi]\,;\Upsilon] \quad \Leftrightarrow [\Phi$ **or** $[\Psi\,;\Upsilon]]$

The above facts in fact express what is at issue in the examples 3–6. Since the dynamic implication associates with conjunction, example 3 turns out to be equivalent with the sentence *If a client comes in, you treat him politely and offer him coffee*, and this sentence is interpreted appropriately in *DMG*. Likewise, given the dynamics of the implication and of the universal quantificier, example 4 is equivalent with *Every player chooses a pawn which he puts on square one*. Finally, the full interdefinability of **or** and ⇒ makes examples 5 and 6 equivalent and by the dynamics of the implication they get the same meaning as the sentence *If there is a bathroom here, it is in a funny place and not on the ground floor*.

Some examples

I will now show, in a more detailed way, what truth-conditions are associated with (slight simplifications of) the examples 3–5 (the treatment of 6 runs parallel to that of 3).

If a client comes in, you pamper him. You offer him coffee

Example 3 is associated with the following translation:

$$[\mathcal{E}d_i[\uparrow\mathtt{client}(d_i)\,;\uparrow\mathtt{come}(d_i)] \Rightarrow \uparrow\mathtt{pamper}(d_i)(\mathtt{y})]\,;\uparrow\mathtt{offer}(\mathtt{c})(d_i)(\mathtt{y})$$

This expression has the following truth-conditions:

$$\begin{array}{l}
\downarrow[[\mathcal{E}d_i[\uparrow\mathtt{client}(d_i)\,;\uparrow\mathtt{come}(d_i)] \Rightarrow \uparrow\mathtt{pamper}(d_i)(\mathtt{y})]\,;\uparrow\mathtt{offer}(\mathtt{c})(d_i)(\mathtt{y})] \Leftrightarrow \\
\downarrow[\mathcal{E}d_i[\uparrow\mathtt{client}(d_i)\,;\uparrow\mathtt{come}(d_i)] \Rightarrow [\uparrow\mathtt{pamper}(d_i)(\mathtt{y})\,;\uparrow\mathtt{offer}(\mathtt{c})(d_i)(\mathtt{y})]] \Leftrightarrow \\
\downarrow\mathcal{A}d_i[[\uparrow\mathtt{client}(d_i)\,;\uparrow\mathtt{come}(d_i)] \Rightarrow [\uparrow\mathtt{pamper}(d_i)(\mathtt{y})\,;\uparrow\mathtt{offer}(\mathtt{c})(d_i)(\mathtt{y})]] \Leftrightarrow \\
\downarrow\mathcal{A}d_i[\uparrow\mathtt{client}(d_i) \Rightarrow [\uparrow\mathtt{come}(d_i) \Rightarrow [\uparrow\mathtt{pamper}(d_i)(\mathtt{y})\,;\uparrow\mathtt{offer}(\mathtt{c})(d_i)(\mathtt{y})]]] \Leftrightarrow \\
\forall d_i(\mathtt{client}(d_i) \rightarrow (\mathtt{come}(d_i) \rightarrow (\mathtt{pamper}(d_i)(\mathtt{y}) \wedge \mathtt{offer}(\mathtt{c})(d_i)(\mathtt{y})))) \Leftrightarrow
\end{array}$$

$\forall x(\mathtt{client}(x) \rightarrow (\mathtt{come}(x) \rightarrow (\mathtt{pamper}(x)(\mathtt{y}) \wedge \mathtt{offer}(\mathtt{c})(x)(\mathtt{y}))))$

(By means of the associativity fact 1.3 and the associativity and reduction facts 2.8 (twice), 2.11 and 2.4 from chapter 1 respectively.) In other words, 3 will be true (or satisfied) if any client that comes in is pampered and offered coffee (by you).[2]

Every player chooses a pawn. He puts it on square one

Functional application and some reductions yield the following translation of 4:

$\mathcal{A}d_i[\uparrow\mathtt{player}(d_i) \Rightarrow \mathcal{E}d_j[\uparrow\mathtt{pawn}(d_j)\,;\uparrow\mathtt{choose}(d_j)(d_i)]]\,;\uparrow\mathtt{put}(\mathtt{1})(d_j)(d_i)$

The truth-conditions can be determined in the following way:

$\downarrow[\mathcal{A}d_i[\uparrow\mathtt{player}(d_i) \Rightarrow \mathcal{E}d_j[\uparrow\mathtt{pawn}(d_j)\,;\uparrow\mathtt{choose}(d_j)(d_i)]]\,;\uparrow\mathtt{put}(\mathtt{1})(d_j)(d_i)] \Leftrightarrow$
$\downarrow\mathcal{A}d_i[[\uparrow\mathtt{player}(d_i) \Rightarrow \mathcal{E}d_j[\uparrow\mathtt{pawn}(d_j)\,;\uparrow\mathtt{choose}(d_j)(d_i)]]\,;\uparrow\mathtt{put}(\mathtt{1})(d_j)(d_i)] \Leftrightarrow$
$\downarrow\mathcal{A}d_i[\uparrow\mathtt{player}(d_i) \Rightarrow [\mathcal{E}d_j[\uparrow\mathtt{pawn}(d_j)\,;\uparrow\mathtt{choose}(d_j)(d_i)]\,;\uparrow\mathtt{put}(\mathtt{1})(d_j)(d_i)]] \Leftrightarrow$
$\downarrow\mathcal{A}d_i[\uparrow\mathtt{player}(d_i) \Rightarrow \mathcal{E}d_j[\uparrow\mathtt{pawn}(d_j)\,;[\uparrow\mathtt{choose}(d_j)(d_i)\,;\uparrow\mathtt{put}(\mathtt{1})(d_j)(d_i)]]] \Leftrightarrow$
$\forall d_i(\mathtt{player}(d_i) \rightarrow \exists d_j(\mathtt{pawn}(d_j) \wedge (\mathtt{choose}(d_j)(d_i) \wedge \mathtt{put}(\mathtt{1})(d_j)(d_i)))) \Leftrightarrow$
$\forall x(\mathtt{player}(x) \rightarrow \exists y(\mathtt{pawn}(y) \wedge (\mathtt{choose}(y)(x) \wedge \mathtt{put}(\mathtt{1})(y)(x))))$

(By means of the associativity fact 1.3 (twice), and the associativity and reduction facts 2.6, 2.11 and 2.4 from chapter 1 respectively.) So, this example is true in *DMG* (or, again: satisfied) if every player chooses a pawn which he puts on square one. Observe, once more, that these truth-conditions are the result of a completely compositional interpretation procedure.

Either there is no bathroom here or it is upstairs. It is not downstairs

The translation of 5 reduces to the following formula:

$[\sim\mathcal{E}d_i[\uparrow\mathtt{bathroom}(d_i)\,;\uparrow\mathtt{here}(d_i)]\ \mathbf{or}\ \uparrow\mathtt{up}(d_i)]\,;\sim\uparrow\mathtt{down}(d_i)$

This formula has the following truth-conditions:

$\downarrow[[\sim\mathcal{E}d_i[\uparrow\mathtt{bathroom}(d_i)\,;\uparrow\mathtt{here}(d_i)]\ \mathbf{or}\ \uparrow\mathtt{up}(d_i)]\,;\sim\uparrow\mathtt{down}(d_i)] \Leftrightarrow$
$\downarrow[\sim\mathcal{E}d_i[\uparrow\mathtt{bathroom}(d_i)\,;\uparrow\mathtt{here}(d_i)]\ \mathbf{or}\ [\uparrow\mathtt{up}(d_i)\,;\sim\uparrow\mathtt{down}(d_i)]] \Leftrightarrow$
$\downarrow[\mathcal{E}d_i[\uparrow\mathtt{bathroom}(d_i)\,;\uparrow\mathtt{here}(d_i)] \Rightarrow [\uparrow\mathtt{up}(d_i)\,;\sim\uparrow\mathtt{down}(d_i)]] \Leftrightarrow$
$\downarrow\mathcal{A}d_i[\uparrow\mathtt{bathroom}(d_i) \Rightarrow [\uparrow\mathtt{here}(d_i) \Rightarrow [\uparrow\mathtt{up}(d_i)\,;\sim\uparrow\mathtt{down}(d_i)]]] \Leftrightarrow$
$\forall d_i(\mathtt{bathroom}(d_i) \rightarrow (\mathtt{here}(d_i) \rightarrow (\mathtt{up}(d_i) \wedge \neg\mathtt{down}(d_i)))) \Leftrightarrow$
$\forall x(\mathtt{bathroom}(x) \rightarrow (\mathtt{here}(x) \rightarrow (\mathtt{up}(x) \wedge \neg\mathtt{down}(x))))$

(By means of the associativity and interdefinability facts 1.3 and 1.2 and the associativity and reduction facts 2.8, 2.11 and 2.4 from chapter 1 respectively.) In other words, for example 5 to be true any bathroom that happens to be in this place should be upstairs and not downstairs.

---

2. Instead of the truth-conditions, it might be more appropriate to speak of the satisfiability-conditions of this example. As Groenendijk and Stokhof remark, this example (and also example 4) is most naturally interpreted as an instructive discourse. However, we are only concerned with the external dynamics of a conditional here, be it a declarative conditional or an imperative one.

1.2 Constraints on the extended dynamics

So far, it seems there is only reason to be glad. By just reformulating *MG* in order to capture *DRT* results in a compositional way, we get a system that can be extended in a natural way to a system that accounts for phenomena which it was not designed to account for in the first place.

The *DMG* analysis of the anaphoric relationships at issue is reminiscent of what Roberts [1987, 1989], in *DRT*, labels the "insertion approach". On the insertion approach, one gets results which are comparable to the extended dynamic equivalences in fact 1.3 by inserting conditions expressed by sentences in a discourse into representations which are *embedded* in the discourse representation structure that constitutes the context. For instance, on the insertive approach a *DRS* for example 3 can be constructed by adding the condition expressed by *You offer him coffee* not, as is usual, to the context's main *DRS*, but to the embedded *DRS* for the clause *you pamper him* which is a constituent of the *DRS* for *If a client comes in, you pamper him*. The discourse representation that results from this is identical to the one associated with the sentence *If a client comes in you pamper him and offer him coffee*.

The major difference between the insertion approach and the extended dynamic analysis presented above is that the last gives a purely semantic account of the phenomena, solely in terms of dynamic negation, existential quantification and conjunction. The gain is that the semantic analysis of extended *DMG* lays bare systematic semantic relationships and that it offers a uniform explanation of the equivalence of the examples 1 and 2 and that of the examples 5 and 6. In particular, the successful treatment of Partee's bathroom disjunction in example 5 (*Either there is no bathroom here or it is in a funny place*) speaks in favour of the semantic approach, since it remains unclear how one can cope with this sentence by means of insertion only.[3]

It must be said that neither the extended dynamics of *DMG*, nor the insertion approach in *DRT*, enables a treatment of all possible anaphoric relationships. Groenendijk and Stokhof propose the analyses as "illustrations of the possibilities inherent in *DMG* rather than as final analyses of the phenomena in question" and they also indicate that it is a "restricted set of facts" being discussed. Not all pronouns can be treated along these lines, by stretching the scope of the relevant operators so to speak. Evans [1977] presents examples that do not seem to fit into the extended dynamic approach, nor into the insertion approach, and more examples can be found in Roberts [1989] and Heim [1990]. However, extended *DMG does* give a systematic account of the possible dynamics of the quantifiers and operators of first

---

3. Analyses of the bathroom disjunction have been offered within the framework of *DRT*, but not in terms of insertion. For instance, Roberts [1987] uses the technique of accommodation in order to account for example 5.

order predicate logic in a uniform fashion. I take it that this constitutes motivation enough for a further study.

A problem with dynamic negation

Groenendijk and Stokhof's treatment of the extended dynamics raises two more issues. In the first place it remains to be investigated what (contextual) factors trigger a dynamic, rather than a static interpretation of the various operators. It is certainly beyond doubt that it is undesirable to always use the dynamic variant of, say, the implication (cf., also, the examples motivating the *static* negation in section 3 of chapter 1). I will not go into this issue here, but only point out that the static interpretations of the operators are still available. They can be derived from the dynamic interpretations using the closure operation $\uparrow\downarrow$.

The second issue is much more pressing. As Groenendijk and Stokhof point out themselves, the extended dynamics of *DMG* is based on a peculiar property of dynamic negation, viz., that it associates with conjunction:

Fact 1.4 (Associative negation)

- $[\sim\Phi\ ;\Psi] \Leftrightarrow \sim[\Phi\ ;\Psi]$

Notice, first, that it is this property of dynamic negation that makes it possible that pronouns in Ψ get bound by quantifiers in Φ. However, it is a problematic property. Fact 1.4 says that the scope of the negation of a sentence extends to sentences that follow it in the discourse. This implies that, for instance, if John walks and is not riding a horse, then, although *John does not walk* is false, the conjunction *John does not walk. He is riding a horse* is true, since the conjunction is equivalent with the (true) sentence *It is not true that John walks and is riding a horse*. This is absurd. So, here we face a fundamental problem with dynamic negation and the remainder of this chapter is devoted to solving it.[4]

Notice that the negation of a formula Φ, all by itself, is not that problematic. In the first place, $\sim\Phi$ is true iff Φ is false, since $\downarrow\sim\Phi \Leftrightarrow \neg\downarrow\Phi$. In the second place, dynamic negation gives rise to the classical equivalences $\sim\sim\Phi \Leftrightarrow \Phi$, $\sim\mathcal{A}d\Phi \Leftrightarrow \mathcal{E}d\sim\Phi$ and $\sim[\Phi \Rightarrow \Psi] \Leftrightarrow [\Phi\ ;\sim\Psi]$. So much seems fine. Problems only show up when the dynamics of $\sim\Phi$ is applied, that is, when a formula $\sim\Phi$ is combined with another formula Ψ by means of a sentential connective. Since dynamic conjunction is the only primitive sentential connective, only (sub)formulas of the form $[\sim\Phi\ ;\Psi]$ turn out to be problematic, cf., fact 1.4.

Still, not every conjunction with a first conjunct of the form $\sim\Phi$ is analyzed

4. It should be noted that the insertion approach faces the same problem. If it is allowed to add the conditions expressed by a sentence to a representation that is embedded in the main representation that constitutes the context, then it is allowed to add them to a representation preceded by the negation sign in the main representation. The semantic result is the same as in fact 1.4.

inappropriately. For instance, the conjunction of the negation of a negated formula $\sim\Phi$ with Ψ turns out equivalent with the conjunction of Φ itself with Ψ, and this is what we want in view of the examples 1 and 2. Furthermore, by the definition of $\Rightarrow$, $\mathcal{A}d$ and **or**, the *DFL* analysis of the examples 3–5 also involves conjunctions of the form $[\sim\Phi\ ;\ \Psi]$. As we saw, these examples are analyzed to our satisfaction. So, one may wonder how a notion of negation that seems so ill-behaved in view of fact 1.4, lies at the heart of the well-behaved extended dynamic interpretations of the universal quantifier, implication and disjunction. As Groenendijk and Stokhof show, this fact has a relatively straightforward semantic explanation in terms of the monotonicity properties of *DFL* formulas.

Negation and monotonicity

In chapter 1 it was shown that all *DFL* sentences denote upward monotonic quantifiers over states, and this, it was argued, appears to be in correspondence with the intuition that sentence sequencing produces information update.

With the present notion of dynamic negation things change, since it introduces *downward* monotonic quantifiers in *DFL*. A quantifier is called downward monotonic if it contains all subsets of any set it contains:

$$\mathcal{Q} \text{ is downward monotonic iff } \forall P, Q \text{ if } P \supseteq Q \text{ and } \mathcal{Q}(P) = 1 \text{ then } \mathcal{Q}(Q) = 1$$

A well-known fact from the theory of generalized quantifiers is that the negation of an upward monotonic quantifier, defined as its complementation, returns a downward monotonic quantifier, and vice versa. So, since dynamic negation *is* complementation, it turns an upward monotonic *DFL* formula into a downward monotonic one.[5]

Since dynamic negation introduces downward monotonic formulas in *DFL*, not all conjunctions are updates anymore. Consider the following fact, intimately related to fact 1.4:

Fact 1.5 (Downdate)

- $\downarrow\Phi$ entails $\downarrow[\Phi\ ;\ \Psi]$ if Φ is downward monotonic[6]

This fact tells us that if Φ is downward monotonic, then its conjunction with Ψ is less informative then Φ itself is and, hence, conjunction with Ψ does not produce information *update*, but information *downdate*. This, of course, is at odds with intuition, and should not be allowed.

---

5. Again, this applies, literally, to the *extensions* of *DFL* formulas. So, suppose a formula Φ is upward monotonic, i.e., if $[\![p]\!] \subseteq [\![q]\!]$ and $[\![{}^\vee\Phi]\!]([\![p]\!]) = 1$ then $[\![{}^\vee\Phi]\!]([\![q]\!]) = 1$. Then $\sim\Phi$ is downward monotonic, i.e., if $[\![p]\!] \supseteq [\![q]\!]$ and $[\![{}^\vee{\sim}\Phi]\!]([\![p]\!]) = 1$ then $[\![{}^\vee{\sim}\Phi]\!]([\![q]\!]) = 1$.

6. Suppose $[\![\downarrow\Phi]\!] = 1$, i.e., $[\![{}^\vee\Phi({}^\wedge\mathtt{true})]\!] = [\![{}^\vee\Phi]\!]([\![{}^\wedge\mathtt{true}]\!]) = 1$. Clearly, $[\![{}^\wedge({}^\vee\Psi({}^\wedge\mathtt{true}))]\!] \subseteq [\![{}^\wedge\mathtt{true}]\!]$. So, if Φ is downward monotonic, then also $[\![{}^\vee\Phi]\!]([\![{}^\wedge({}^\vee\Psi({}^\wedge\mathtt{true}))]\!]) = [\![{}^\vee[\Phi\ ;\ \Psi]({}^\wedge\mathtt{true})]\!] = 1$, i.e., $[\![\downarrow[\Phi\ ;\ \Psi]]\!] = 1$.

The reason why a monotonicity reversing notion of dynamic negation does not produce wrong results for the examples 1–6 is that the respective analyses, even though they are cast, among others, in terms of dynamic negation, do not involve downdates. In the translations of the examples, each occurrence of a conjunction $[\Phi\,;\Psi]$ has an upward monotonic formula Φ as its first conjunct and, therefore, each conjunction of such a formula Φ with a formula Ψ produces an update of Φ. So, the suspicious fact 1.4, or the terrible fact 1.5, are completely irrelevant for the analyses of these examples. It is expedient to substantiate this claim by inspecting two of the examples in terms of their monotonicity properties.

First consider example 1, *It is not the case that John does not own a car*. The translation of this sentence has the form $\sim\sim\Phi$, where Φ says that John owns a car. In this example the subformula Φ is upward monotonic, and, since dynamic negation reverses monotonicity, $\sim\Phi$ is downward monotonic. However, since $\sim\Phi$ is downward monotonic, $\sim\sim\Phi$ is upward monotonic again, and it is this upward monotonic formula $\sim\sim\Phi$ that is conjoined with the translation of the sentence *It is red etc.* We see that, although the translation of this example contains a downward monotonic subformula, viz., $\sim\Phi$, the conjunction as a whole does not contain a downward monotonic formula as a first conjunct, i.e., it involves no downdate.

As for a second example, consider the extended bathroom disjunction 5, *Either there is no bathroom here, or it is in a funny place. In any case, it is not on the ground floor*. This example has a translation of the following form:

$$[[\sim\mathcal{E}d\Phi\ \mathbf{or}\ \Psi]\,;\Upsilon]$$

(Where Φ says that d is a bathroom here, Ψ that d is in a funny place, and Υ that d is not on the ground floor.) Writing out the disjunction, we get the following formula:

$$[\sim[\sim\sim\mathcal{E}d\Phi\,;\sim\Psi]\,;\Upsilon]$$

We find two conjunctions here. First, there is the conjunction of $\sim\sim\mathcal{E}d\Phi$ and $\sim\Psi$. Since $\mathcal{E}d\Phi$ is upward monotonic, its double negation $\sim\sim\mathcal{E}d\Phi$ is upward monotonic as well. So, the conjunction of $\sim\sim\mathcal{E}d\Phi$ with $\sim\Psi$ involves a genuine update of $\sim\sim\mathcal{E}d\Phi$. Notice, however, that the result of this conjunction itself is downward monotonic. For, Ψ being upward monotonic, the second conjunct $\sim\Psi$ is downward monotonic and therefore its conjunction with the upward monotonic first conjunct $\sim\sim\mathcal{E}d\Phi$ is also downward monotonic.[7] The second conjunction has as its first conjunct the

7. The conjunction of a downward monotonic and an upward monotonic formula is downward monotonic. Suppose Φ is upward monotonic and Ψ is downward monotonic and let $[\![p]\!] \supseteq [\![q]\!]$. Since Ψ is downward monotonic, always if $[\![{}^\vee\Psi]\!]([\![p]\!]) = 1$ then $[\![{}^\vee\Psi]\!]([\![q]\!]) = 1$, so $[\![{}^\wedge({}^\vee\Psi(p))]\!] \subseteq [\![{}^\wedge({}^\vee\Psi(q))]\!]$. Now suppose $[\![{}^\vee[\Phi\,;\Psi]]\!]([\![p]\!]) = 1$, i.e., $[\![{}^\vee\Phi({}^\wedge({}^\vee\Psi(p)))]\!] = [\![{}^\vee\Phi]\!]([\![{}^\wedge({}^\vee\Psi(p))]\!]) = 1$. Since Φ is upward monotonic, we find that $[\![{}^\vee\Phi]\!]([\![{}^\wedge({}^\vee\Psi(q))]\!]) = 1$, and, hence, that $[\![{}^\vee[\Phi;\Psi]]\!]([\![q]\!]) = 1$. By a completely similar proof we find that if Φ is downward monotonic and Ψ is upward monotonic, then if $[\![p]\!] \supseteq [\![q]\!]$ and $[\![{}^\vee[\Phi\,;\Psi]]\!]([\![p]\!]) = 1$, then $[\![{}^\vee\Phi\,;\Psi]]\!]([\![q]\!]) = 1$.

negation of the first conjunction. Since the first conjunction is downward monotonic, its negation is upward monotonic, and therefore the conjunction of $\sim[\sim\sim\mathcal{E}d\Phi\,;\sim\Psi]$ with Υ also produces a real update of $\sim[\sim\sim\mathcal{E}d\Phi\,;\sim\Psi]$. In other words, in the processing of this example we encounter three downward monotonic subformulas, viz., $\sim\mathcal{E}d\Phi$, $\sim\Psi$ and $[\sim\sim\mathcal{E}d\Phi\,;\sim\Psi]$, but no (local) downdates.

A solution to the problem

The above considerations seem to motivate the conclusion that it is not (necessarily) the presence of downward monotonic formulas in extended *DFL* that is problematic, but their behaviour in conjunctions, as exemplified in fact 1.5 (but, cf. below). In other words, the conclusion might be that the problem with negation can be solved, not by adopting an alternative notion of negation, but simply by restricting the use of conjunction. We can retain the notion of dynamic negation defined above, and with that save the analysis of the examples 1–6, if only we exclude, in some way or other, that generated downward monotonic sentence interpretations get conjoined, and thus expel the possibility of downdate. This solution is the one proposed by Groenendijk and Stokhof.

Groenendijk and Stokhof propose to exclude the possibility of downdate by means of constraints on translations: "the translation of each sentence which constitutes a separate step in the discourse should denote an upward monotonic quantifier over states." This constraint allows the presence of downward monotonic sentences, except in certain constructions. Groenendijk and Stokhof mention discourse sequencing, relative clause formation and the formation of conditional sentences as constructions to which the constraint should apply. So, the translation of a sequence $\sigma.\tau$ of two sentences σ and τ can be $[\sigma'\,;\tau']$ only if σ' is upward monotonic. If σ' is downward monotonic, then the translation $[\uparrow\downarrow\sigma'\,;\tau']$ has to be used, since, for any formula Φ, $\uparrow\downarrow\Phi$ is upward monotonic. In a similar way the antecedent of a conditional is required to be upward monotonic, as is the restriction in a restrictive relative clause.

The net effect of this general constraint is that every conjunction that occurs in the translation of any *DMG* sentence has an upward monotonic first conjunct.[8]

8. Groenendijk and Stokhof's constraint does not mention all conjunctions occurring in the *DMG* fragment, but the following observation shows that also the conjunctions which are not mentioned observe the monotonicitiy constraint. The constraint on the translation of relative clauses makes that all common noun phrases in *DMG* denote functions from individuals to upward monotonic quantifiers over states: for atomic common noun phrases this holds by definition, and it holds for compound common nouns since they only allow upward monotonic restrictions, this by the constraint. From this it follows that the conjunction in the translation of an existentially quantified structure $an_{DET}(A_{CN})(B_{IV})$ $(\mathcal{E}d[A'(\uparrow d)\,;B'(\uparrow d)])$ and the one implicit in the translation of a structure $every_{DET}(A_{CN})(B_{IV})$ $(\mathcal{A}d[A'(\uparrow d) \Rightarrow B'(\uparrow d)] = \sim\mathcal{E}d[A'(\uparrow d)\,;\sim B'(\uparrow d)])$ also have an upward monotonic first conjunct (viz., $A'(\uparrow d)$). This exhausts the conjunctions not directly mentioned in Groenendijk and Stokhof's constraint.

Groenendijk and Stokhof's constraint thus preserves the good of the dynamic negation, the extended dynamics, and it excludes the bad, its misconduct as a first conjunct.

However, the use of such constraints is not a very satisfactory solution to the problem with dynamic negation. In medical terms, constraints do not combat the disease but only its symptoms. Or, as Groenendijk and Stokhof remark: "one would prefer to restrict the logical system itself rather than the use that is made of it" (p. 33). So, it would be preferable to have a more principled solution of the problem in the form of a system in which the operators themselves exclude the unwanted downdates, i.e., a system in which negation is dynamic and preserves (upward) monotonicity.

In section 2 I present such a solution, but first, in the remaining part of this section, I argue that the present solution is also empirically unsatisfactory. I will discuss some examples which are structurally related to the examples which extended *DMG* accounts for but which violate the stated monotonicity constraint. We will see that in order to cope with these examples the constrained extended dynamics can, again, be improved upon by associating with sentences both an upward and a downward monotonic interpretation. However, since the required adjustment involves complicating interpretation at a local level in order to be able to satisfy the monotonicity constraints at a global level, it, in fact, illustrates the importance of having a notion of dynamic negation that preserves monotonicity and that, hence, enables us to do without constraints.

1.3 Limits to the constraints

Groenendijk and Stokhof's 'constrained extended dynamic approach' to anaphora excludes downward monotonic formulas from conjunction with a second conjunct. So, if we want to conjoin a sentence, and if it has a downward monotonic translation, we have to make it upward monotonic first. This is accomplished by taking the static closure $\uparrow\downarrow$ of the translation of the sentence, since the static closure $\uparrow\downarrow\Phi$ of any formula Φ is upward monotonic. (The same goes for the antecedent of a conditional sentence and for the sentence that restricts a relative clause.) Notice that, although the static closure of Φ preserves Φ's truth-conditions, since $\downarrow(\uparrow\downarrow\Phi) \Leftrightarrow \downarrow\Phi$, it deprives Φ of its dynamic potential.

However, this is only part of the story. In order to treat the bathroom disjunction, the extended *DMG* fragment must have a rule of disjunction formation. In the statement of this rule too, of course, a monotonicity constraint has to apply, be it a different one this time. Since $[\Phi\ \mathbf{or}\ \Psi] = [\sim\Phi \Rightarrow \Psi]$, since the antecedent of an implication must be upward monotonic, and since dynamic negation reverses monotonicity, a disjunction's first disjunct is required to be *downward* monotonic: the translation of a disjunction σ *or* τ of two sentences σ and τ is $[\sigma'\ \mathbf{or}\ \tau']$ only if σ' is downward monotonic.

The question now arises what to do if a first disjunct is upward monotonic. In the case of sentence conjunction, we saw, a downward monotonic first conjunct should be turned into a truth-conditionally equivalent upward monotonic one, by taking the static closure of the formula. In the case of sentence disjunction we now have to turn an upward monotonic first disjunct Φ into a truth-conditionally equivalent downward monotonic one. For this we need the 'dual' of the static closure of Φ: $\sim\uparrow\downarrow\sim\Phi$. Clearly, since for any formula Φ, $\uparrow\downarrow\Phi$ is upward monotonic, $\sim\uparrow\downarrow\Phi$ is downward monotonic and so is $\sim\uparrow\downarrow\sim\Phi$. Furthermore, the truth-conditions of the dual of the static closure of Φ are those of Φ: $\downarrow(\sim\uparrow\downarrow\sim\Phi) \Leftrightarrow \neg\downarrow\uparrow\neg\downarrow\Phi \Leftrightarrow \neg\neg\downarrow\Phi \Leftrightarrow \downarrow\Phi$. Notice that also the dual of the static closure of Φ, like the static closure of Φ, deprives Φ of its dynamic potential.

The above observations imply that in the construction of compound sentences in extended *DMG*, (sub)sentences that do not immediately comply with the monotonicity contraint are deprived of their context change potential. This, however, is not empirically motivated. Consider the following examples 7 and 8, which are simple variants of 3 and 4:

(7) No player leaves the room. He stays were he is.
(8) No client that comes in is offered coffee. He is directly sent up to me.

On the expected interpretation of the determiner *no*, which is the dynamic negation of the determiner *a*, the first sentences of these examples are downward monotonic. So, in the respective conjunctions they have to be made upward monotonic by means of the closure operator. However, this closure precludes the apparent anaphoric relationships which obtain between the noun phrases in the first sentences and the pronouns in the second ones.

Groenendijk and Stokhof remark that we get an adequate interpretation of the examples 7 and 8, if we use the following alternative interpretation of the determiner *no*:

$$\lambda P \lambda Q \sim\mathcal{E}d[P(\uparrow d)\,;\sim\uparrow\downarrow\sim Q(\uparrow d)]$$

This interpretation of the determiner *no* is dynamic, and it guarantees upward monotonicity.[9] So, if the determiner *no* is interpreted as indicated, the examples 7 and 8 comply with the monotonicity constraint and they turn out true iff every player does not leave the room but stays where he is, and every client that comes in is not offered coffee but sent up to me, respectively. We see that, using the dynamic interpretation of the determiner *no* above, the examples 7 and 8 are analyzed appropriately.

9. This translation involves an embedded dual static closure over the second argument of *no*. This closure makes the embedded clause $Q(\uparrow d)$ downward monotonic, but, since this clause is in the scope of the main negation, the whole is upward monotonic again.

However, notice that the determiner should not always be interpreted like that. The *upward* monotonic interpretation of the determiner *no* is not appropriate for a treatment of the bathroom disjunction. We saw above that a disjunction requires a downward monotonic first disjunct. So, if we use the above interpretation of the determiner *no*, the first disjunct in the bathroom disjunction *there is no bathroom here* turns out upward monotonic and its dual static closure has to be used. But then the noun phrase *no bathroom* can not bind the pronoun in the second disjunct anymore.

It appears, then, that we need the upward monotonic interpretation of the determiner *no* in some cases, viz., for a treatment of the examples 7 and 8, and in other cases we need the downward monotonic interpretation, viz., for a treatment of example 5. In other words, the determiner should be assumed to be ambiguous, and, depending on the construction the determiner figures in, its downward or its upward monotonic interpretation has to be selected.

Similar observations can be made with respect to disjunctions and conditional sentences. We can find disjunctions and conditionals of which the first clause does not have the required monotonicity properties, but nevertheless contains a noun phrase that binds a pronoun in the second. The following examples are again simple variants of examples discussed above (example 9 is also discussed by Craige Roberts and attributed to Steve Berman):

(9) Either there is a bathroom downstairs, or it is upstairs.
(10) If it is not the case that there is a bathroom downstairs, then it is upstairs.

Extended *DMG* assigns the first disjunct of example 9 an upward monotonic interpretation. Since a first disjunct in extended *DMG* is required to be downward monotonic, the dual static closure has to be used in example 9, but that precludes the anaphoric relationship between the indefinite *a bathroom* in the first disjunct and the pronoun in the second. Example 10 is the implicational counterpart of the disjunction 9. The antecedent of this implication is the (dynamic) negation of the first disjunct of 9. Since the monotonicity constraint requires the antecedent of an implication to be upward monotonic, we have to use the static closure of the negation (or, equivalently, the static negation), but that leaves the pronominal anaphor in the consequent unexplained.

In order to cope with the examples 9 and 10 we need a dynamic but downward monotonic interpretation of the clause *there is a bathroom downstairs*. This can be achieved by the following alternative interpretation of the determiner *a* in which, as in the alternative *upward* monotonic interpretation of *no*, the dual static closure of the determiner's second argument is employed:

$$\lambda P \lambda Q\ \mathcal{E}d[P(\uparrow d)\,;\sim\uparrow\downarrow\sim Q(\uparrow d)]$$

Using this interpretation of the determiner *a*, both examples 9 and 10 turn out true iff there is a bathroom which is either downstairs or upstairs, which seems correct. Notice that the downward monotonic interpretation of *a* and the upward monotonic interpretation of *no* above are each other's dynamic negation.

Of course, the downward monotonic interpretation of the determiner *a* should not be used in all cases. We need a downward monotonic interpretation for a proper analysis of examples like 9 and 10, and an upward monotonic interpretation for a proper analysis of, for instance, example 6. So, like the determiner *no*, the determiner *a* must be assumed to be ambiguous and its downward or upward monotonic interpretation has to be selected according to the monotonicity requirements of the compound structures in which the determiner occurs.

It appears that for a uniform treatment of the examples 1–6, which extends to the similar examples 7–10, the *DMG* quantifiers should be assigned both an upward and a downward monotonic interpretation. True, it need not necessarily be the quantifiers from which the ambiguity originates, but what seems required at least is that sentences which contain quantifiers get assigned both an upward and a downward monotonic dynamic interpretation. The constraints on the construction of compound sentences then can be taken to *select* those interpretations of its (quantified) constituent sentences that have the required monotonicity. Thus, such constructions can always be interpreted dynamically *and* in compliance with the monotonicity constraints.

In pursuit of an alternative negation

We see that *DMG* can be extended so as to cover the extended dynamics of the first order quantifiers and operators by supplying it with a monotonicity reversing notion of dynamic negation, a monotonicity constraint on dynamic conjunction and a mechanism by means of which (quantified) sentences are assigned both upward and downward monotonic dynamic interpretations. It must be noticed that the need to be supplied with both up- and downward monotonic readings of sentences here in fact follows from (i) the (intuitive) requirement that every first conjunct in a conjunction is upward monotonic and (ii) the fact that negation reverses monotonicity. Only by positing this ambiguity it is guaranteed that both $[\Phi\ ;\ \Psi]$ and $[\sim\Phi\ ;\ \Psi]$ can be interpreted dynamically and in compliance with the monotonicity constraint. (Recall that the constraints on implication and disjunction follow from the constraint on conjunction by their definition.)

The need to posit such an ambiguity stresses, once more, the desirability of having a monotonicity *preserving* notion of dynamic negation. If we have such a monotonicity preserving notion of dynamic negation then all *DFL* formulas are guaranteed to be upward monotonic, the constraints on conjunctions will always be satisfied, and, hence, no ambiguity needs to be posited. So, it appears, matters can be greatly simplified if we find such a notion of negation.

Now let us ask ourselves more specifically what a really adequate notion of dynamic negation should be like. In the first place, it should involve negation of truth-conditional content, otherwise it could hardly be called a form of negation. In the second place, it should be dynamic, of course. In the third place it should preserve monotonicity. We can formalize these demands as follows:

1. $\downarrow\sim\Phi \Leftrightarrow \neg\downarrow\Phi$
2. $\sim\sim\Phi \Leftrightarrow \Phi$
3. If Φ is upward monotonic, then $\sim\Phi$ is upward monotonic

According to the first requirement the dynamic negation of Φ must be true if Φ is false, and vice versa. If this requirement is met then dynamic negation implies *negation* of truth-conditional content. The second requirement, that negation satisfies the law of double negation, lends negation a *dynamic* character. The third requirement speaks for itself.

So we have three requirements that a notion of dynamic negation must satisfy. The static negation of *DMG* obeys the first and the third requirement, but not the second, and the dynamic negation of extended *DMG* obeys the first and the second requirement, but not the third. The question now is whether there is an operation in the framework of *DMG* that satisfies all three requirements, and the straightforward answer to that question is that there is not:

Fact 1.6
In the framework of *DMG* no operation satisfies the above three requirements.

This fact is proved by contraposition. Suppose some function, denoted by $\sim$, satisfies the three requirements. Then take two closed formulas Φ and Ψ, both upward monotonic, true with respect to all states s and such that $[\![\Phi]\!] \neq [\![\Psi]\!]$. (For instance, let Φ and Ψ be $\mathcal{E}d(\uparrow d \mathrel{\tilde{=}} \uparrow d)$ and $\mathcal{E}d'(\uparrow d' \mathrel{\tilde{=}} \uparrow d')$, respectively.) Then for all s: $[\![\downarrow\Phi]\!]_s = [\![\downarrow\Psi]\!]_s = 1$ and, hence, $[\![\neg\downarrow\Phi]\!]_s = [\![\neg\downarrow\Psi]\!]_s = 0$. So, by 1, for all s: $[\![\downarrow\sim\Phi]\!]_s = [\![\downarrow\sim\Psi]\!]_s = 0$. Since Φ is upward monotonic, by 3, $\sim\Phi$ is upward monotonic, and since $[\![\downarrow\sim\Phi]\!]_s = [\![\sim\Phi]\!]_s(s)(I) = 0$, we find that for all s and p: $[\![\sim\Phi]\!]_s(s)(p) = 0$, and, Φ being intensionally closed, for all s, s' and p: $[\![\sim\Phi]\!]_{s'}(s)(p) = 0$. Completely similarly we find that for all s, s' and p: $[\![\sim\Psi]\!]_{s'}(s)(p) = 0$. So, $[\![\sim\Phi]\!] = [\![\sim\Psi]\!]$, and, hence, $[\![\sim\sim\Phi]\!] = [\![\sim\sim\Psi]\!]$. But then, by 2, $[\![\Phi]\!] = [\![\Psi]\!]$. This contradicts the assumption that $[\![\Phi]\!]$ and $[\![\Psi]\!]$ are different.

Where does that leave us? We have seen how *DMG* gives a truly compositional reformulation of *DRT*, and we have observed *DMG*'s inherent possibility to extend the dynamics by adopting a notion of dynamic negation. However, we have also seen that *DMG*'s notion of dynamic negation undermines a most welcome property of *DMG*, viz., that all *DMG* formulas are upward monotonic. Moreover, we have seen that it is not possible to define an adequate alternative monotonicity preserving notion of dynamic negation in the framework of *DMG*.

A *DMG* treatment of all the examples discussed then seems to require us to assign sentences both upward and downward monotonic interpretations. In case of conjunction, implication or disjunction, constraints have to be used to select an interpretation of the first constituent sentence of the connective which has the required monotonicity (upward, upward or downward, respectively). In fact, the cost of having a monotonicity reversing notion of negation is that we have to assume ambiguity at a local level in order to arrive at an appropriate dynamic interpetation at a global level.

In the next section I will show that we can improve upon this situation if we resort to a higher level of types. In that section I present a system *DMG(2)* which is equivalent with *DMG*, but in which it is possible to define a notion of dynamic negation that satisfies the requirements that were set out in this section. Hence, in *DMG(2)* no constraints need to apply, nor does it require us to assume ambiguous interpretation.

2 Dynamic Montague grammar(2)

In this section I present an alternative dynamic Montague grammar, *DMG(2)*, which mainly differs from *DMG* with respect to the type assigned to *DFL* formulas. The whole organization of *DMG(2)* is the same as that of *DMG*, and its first version, the one without dynamic negation, is shown to be equivalent with *DMG*. However, the higher typed system of interpretation used in *DMG(2)* leaves us just the amount of elbow room needed to define an adequate notion of dynamic negation. This notion of negation is presented and discussed in the section 2.2.

2.1 DMG(2) interpretation

Like *DPL* and *DRT*, *DMG* gives an account of the fact that (indefinites in) a sentence of natural language may bring about a change in the context of interpretation that affects the interpretation of (pronouns in) successive sentences. In *DMG* this is accounted for by interpreting sentences as context change potentials which have the type of properties of propositions. In this section I will first give some motivation for the adoption of a higher type of context change potentials and next I give an alternative formulation of *DMG* in that higher type.

Context change potentials reconsidered

In *DMG* a sentence is interpreted as a function from contexts (states) and propositions (sets of states) to truth values. Such a function f can be conceived of as a context change potential since f holds of a state s and a proposition p iff p is true in a state s' which may be different from s and such that certain further conditions associated with f are satisfied, cf., the remarks about context change potentials in chapter 1, section 3. Notice that the realization of the context change potential of a *DMG* sentence comes about by the application of its extension to some propositional expression, typically the intension of a *DIL* formula.

The realization of the context change potential of a *DFL* formula is intimately related to the *DFL* conjunction of that formula with another formula. Basically, the dynamic conjunction of Φ and Ψ involves the application of the extension of Φ to the proposition expressed by Ψ, this, with a supplementary application and abstraction device that ensures that the result of the conjunction is a context change potential again. As we just saw, the realization of Φ's context change potential also consists in the application of Φ's extension to some proposition expression, which shows that dynamic context change and conjunction go hand in hand in *DMG*.[10]

Clearly, such a link between context change and conjunction is anticipated by the definition of the lift of *DIL* formulas to the level of context change potentials. This lift turns a *DIL* formula ϕ into the function $^\wedge\lambda p\ (\phi \wedge {}^\vee p)$ which is associated with the property of propositions of being true *in conjunction with* ϕ. Therefore, the *application* of the extension of $\uparrow\phi$ to the intension of a formula ψ comes down to the static conjunction of ϕ with ψ, and, consequently, the application of the extension of *DFL* formulas more in general, i.e., the realization of their context change potentials, always involves some form of conjunction of static content.

It seems relatively natural to assume a tight connection between (dynamic) context change and (dynamic) conjunction if, first, we are dealing with the changes that (expressions in) one sentence may bring about in the context of interpretation of (expressions in) another sentence, and, second, if conjunction is the only (primitive) sentential connective. But, of course, in natural language the context change potential of one sentence may be applied to that of another sentence which is related to the first in a way other than can be defined in terms of conjunction (for instance, in dialogues). And also if we keep to conjunction as the only primitive connective, the tight connection between context change and conjunction becomes an obstacle when we take into account the changes in the context brought about by the *negation* of a formula Φ in the conjunction of $\sim\Phi$ with Ψ.

If in the dynamic conjunction of $\sim\Phi$ with Ψ the negation of Φ is dynamic, it must be an operation that allows Φ's context change potential to affect the interpretation of the second conjunct Ψ. However, if Ψ is subject to the context change

10. We can indicate the link between context change and conjunction more precisely. Being subject to the context change potential of a formula Φ in *DMG* involves, basically, occurring in a sentential expression to the intension of which the extension of Φ gets applied, possibly after some λ-conversions. So, if an expression α is subject to the context change potential of a formula Φ, it, eventually, occurs in a sentential expression β to the intension of which the extension of Φ applies, as in ${}^\vee\Phi({}^\wedge\beta)$. Observe that ${}^\vee\Phi({}^\wedge\beta)$ is equivalent with ${}^\vee[\Phi\,;{}^\wedge\lambda p\ \beta](p)$. Here we see that α shows up as part of an expression $(\lambda p\ \beta)$ which can be conceived to be (dynamically) conjoined with Φ.

potential of Φ, and if, as appears to be the case in *DMG*, the realization of a context change potential is tied up with some form of conjunction, then, it appears, the conjunction of $\sim\Phi$ with Ψ involves, in some or other way, the negation of the conjunction of Φ with Ψ. As we have already seen (cf., fact 1.4), on the most straightforward definition of dynamic negation in *DMG* this is precisely what happens, that is, that the conjunction of the negation of Φ with Ψ equals the negation of the conjunction of Φ with Ψ.

These observations suggest that, for a generalization of the *DMG* dynamics of interpretation, and for an extension of it with a proper notion of dynamic negation, we have to loosen the too intimate relation between context change and conjunction in the first place. That is, it seems a prerequisite to assign sentences a kind of context change potentials the realization of which is not as tied up with conjunction as it is in *DMG*. We then may see afterwards whether it is possible to define an adequate notion of dynamic negation for context change potentials of such an alternative kind. This is the strategy adopted in the sequel.

Aiming at dissociating context change from conjunction, it is important to observe that the *DMG* type of context change potentials is not the only conceivable one. In any *DIL* type a, there are context change potentials γ of type $\langle\langle s, a\rangle, t\rangle$, which, when applied to the intension of an expression α of type a, involve the evaluation of α in a state possibly different from the state with respect to which γ is evaluated.[11] So, if we want sentences to denote context change potentials, a very legitimate question is what kind of objects we want these context change potentials to apply to, or, which is essentially the same question, for which type of expressions we want them to change the context of interpretation.

The *DMG* answer to this question is that a sentence Φ denotes a context change potential that gets applied to propositions, typically, the propositions expressed by the sentences with which Φ gets conjoined. We have just seen that, for our present concerns, this kind of context change potentials seems to involve too tight a link between context change and conjunction. In order to loosen this link, we have to generalize over the several ways in which a sentence Φ may get combined, and in the case of the conjunction of Φ with a propositional expression ψ, we have to apply the context change potential of Φ, not to the proposition expressed by ψ, but to the function expressed by the conjunction with ψ. Clearly, if context change potentials apply to objects of this type then they may as well apply to functions expressed by other coordinations, such as, for instance, a *disjunction* with Ψ.

The function expressed by the conjunction with a subsequent sentence ψ is here conceived of as the property of propositions of being true in conjunction with ψ, $^\wedge\lambda p\ (^\vee p \wedge \psi)$ of type $\langle s, \langle\langle s, t\rangle, t\rangle\rangle$.[12] Since we want context change potentials

11. For instance, if R is a variable of type $\langle s, a\rangle$, let γ be an expression of the form $\lambda R\ \exists d(\ldots {}^\vee R \ldots)$.
12. Notice that this is the function denoted by the *DFL* lift $\uparrow\psi$ of a *DIL* formula ψ. There is a slight

to be objects the extensions of which get applied to these kinds of functions, they will have to be properties of properties of propositions, i.e., objects of the type $\langle s, \langle\langle s, \langle\langle s, t\rangle, t\rangle\rangle, t\rangle\rangle$.

In the remainder of this section I will give a reformulation of *DMG* into a system *DMG(2)* in which sentences are associated with context change potentials of this type. In its first presentation, the system is shown to be equivalent to *DMG*.

DFL(2) type shift

The formulation of an alternative dynamic Montague grammar, *DMG(2)*, in which sentences are assigned meanings which are properties of properties of propositions, requires only a reformulation of the type and interpretation of the *DFL* language. The reinterpreted system will be referred to as *DFL(2)*. I start with the definition of the types of *DFL(2)*.

The set of *DFL(2)* types $T_{D(2)}$ is defined as follows:

Definition 2.1 (DFL(2) types)
The set of *DFL(2)* types, $T_{D(2)}$, is the smallest set such that:
1. $\langle s, e\rangle \in T_{D(2)}$
2. $\langle s, \langle\langle s, \langle\langle s, t\rangle, t\rangle\rangle, t\rangle\rangle \in T_{D(2)}$
3. if $a, b \in T_{D(2)}$ then $\langle a, b\rangle \in T_{D(2)}$

The following type shift relates the extensional types in T to $T_{D(2)}$:

Definition 2.2 (DFL(2) type shift)
1. ${\uparrow}e \quad = \langle s, e\rangle$
2. ${\uparrow}t \quad = \langle s, \langle\langle s, \langle\langle s, t\rangle, t\rangle\rangle, t\rangle\rangle$
3. ${\uparrow}\langle a, b\rangle = \langle {\uparrow}a, {\uparrow}b\rangle$

So, whereas in *DFL* every (sub)type t is replaced by $\langle s, \langle\langle s, t\rangle, t\rangle\rangle$, the type of properties of propositions, in *DFL(2)* it is replaced by $\langle s, \langle\langle s, \langle\langle s, t\rangle, t\rangle\rangle, t\rangle\rangle$, the type of properties of properties of propositions.

Like the *DFL* type shift, the *DFL(2)* type shift ↑ is associated with an interpretation. The following definition simultaneously defines ↑, the interpretation of type shifts into *DFL(2)* types, and ↓, the interpretation of shifts back into the original *DIL* types:

Definition 2.3 (DFL(2) type shift (interpretation))
1. $\uparrow\phi_e \quad = {}^\wedge\phi$
   $\downarrow\Phi_{\uparrow e} \quad = {}^\vee\Phi$

difference in the conception of the two, since in the formulation of the update with ψ, ψ, suggestively, figures as the second conjunct, and in the lift $\uparrow\psi$, ψ figures as the first conjunct. However, since the conjunction involved is classical, commutative, conjunction, this makes no difference semantically.

2. $\uparrow\phi_t \quad = {}^\wedge\lambda R\ {}^\vee R({}^\wedge\phi)$ (R not free in ϕ)
   $\downarrow\Phi_{\uparrow t} \quad = {}^\vee\Phi({}^\wedge\lambda p\ {}^\vee p)$
3. $\uparrow\phi_{\langle a,b\rangle} = \lambda x_{\uparrow a}\ \uparrow(\phi(\downarrow x))$ (x not free in ϕ)
   $\downarrow\Phi_{\uparrow\langle a,b\rangle} = \lambda x_a\ \downarrow(\Phi(\uparrow x))$ (x not free in Φ)

where R is a variable of type $\langle s, \langle\langle s,t\rangle, t\rangle\rangle$ and p a variable of type $\langle s, t\rangle$

The crucial clause in this definition is again the interpretation of the lift from expressions of type t. The lift of a *DIL* formula ϕ denotes the property of properties of propositions of being a property that the proposition that ϕ has. So, a property of propositions R has the property of properties of propositions denoted by $\uparrow\phi$ iff the proposition that ϕ has the property R. This type shift corresponds to a familiar 'raising' type shift which lifts an object z to the set of sets of objects Z such that a set of objects P is an element of Z iff z is an element of P. (Cf., Montague [1973], Rooth and Partee [1983] and Hendriks [1992] among others. See also chapter 3).

When a property of propositions R is ascribed the property of properties of propositions denoted by a *DFL(2)* formula Φ, i.e., when the extension of Φ is applied to R, I will also say that Φ is ascribed the property R. In the lowering $\downarrow\Phi$ of Φ, then, Φ is ascribed such a property, viz., the property ${}^\wedge\lambda p\ {}^\vee p$, which is the property of being a true proposition. So, the lowering of Φ ascribes Φ the property of being true. As we will see below, this is one of two kinds of properties of propositions which, in *DFL(2)*, a formula Φ may get ascribed. The other property that Φ may get ascribed is that of being true in conjunction with some other proposition, typically, the proposition expressed by the sentence with which Φ gets conjoined.

As is the case in *DFL*, the lowering $\downarrow\Phi$ of a formula Φ gives us Φ's truth conditional content. More in general, the lowering of any *DFL(2)* expression Φ of type ${}^\uparrow a$ returns the static contents of Φ, of type a.

Like in *DFL*, the lowering of the lift of ϕ is equivalent with ϕ:

Fact 2.1 (↓↑-elimination(2))

- $\downarrow\uparrow\phi \Leftrightarrow \phi$

In case ϕ is of type t, then $\downarrow\uparrow\phi$ says that the property of being a true proposition has the property of being a property of the proposition that ϕ, i.e., that the property of being a true proposition is a property that the proposition that ϕ has. In other words, $\downarrow\uparrow\phi$ simply says that the proposition *that* ϕ has the property of being true.

What still does not hold in general is that $\uparrow\downarrow\Phi \Leftrightarrow \Phi$. The formula $\uparrow\downarrow\Phi$ denotes the property of properties of propositions of being a property of the proposition that Φ has the property of being true. Like in *DFL* the lift of the lowering of Φ annihilates the dynamic potential of Φ.

DFL(2) semantics

The language of *DFL(2)* is the same as that of *DFL*, except for the associated types and the fact that *DFL(2)* employs variables of types ${}^{\uparrow}a$ according to the *DFL(2)* definition of the type shift $^{\uparrow}$, where *DFL* employs variables of types ${}^{\uparrow}a$ according to the *DFL* definition of the type shift $^{\uparrow}$.

In the semantics, only the clauses for identity, negation, existential quantification and conjunction have to be adjusted. In *DFL(2)* their interpretation is defined as follows (again, R is a variable of type $\langle s, \langle\langle s,t\rangle, t\rangle\rangle$, and p a variable of type $\langle s,t\rangle$):

Definition 2.4 (DFL(2) semantics)

1. $\alpha \mathrel{\tilde{=}} \beta \;= \uparrow(\downarrow\alpha = \downarrow\beta)$
2. $\sim\Phi \quad = \uparrow\neg\downarrow\Phi$
3. $\mathcal{E}d\Phi \quad = {}^{\wedge}\lambda R\ \exists d({}^{\vee}\Phi(R))$ (R not free in Φ)
4. $[\Phi\,;\Psi] = {}^{\wedge}\lambda R\ {}^{\vee}\Phi({}^{\wedge}\lambda p\ ({}^{\vee}p \wedge {}^{\vee}\Psi(R)))$ (R not free in Φ or Ψ)

Like in *DFL*, an identity statement $\alpha \mathrel{\tilde{=}} \beta$ in *DFL(2)* is interpreted as the lift of the statement that α and β have the same static content. On its first definition negation is static. Like in *DFL*, the negation of Φ is defined as the lift of the negation of Φ's truth-conditional content. The difference with *DFL* resides solely in the interpretation of $\uparrow$ and $\downarrow$, so $\sim\Phi$ now equals ${}^{\wedge}\lambda R\ {}^{\vee}R({}^{\wedge}\neg{}^{\vee}\Phi({}^{\wedge}\lambda p\ {}^{\vee}p))$. The formula $\sim\Phi$ denotes the property of properties of propositions of being a property of the proposition that Φ does not have the property of being true.

Existential quantification in *DFL(2)* is also very similar to existential quantification in *DFL*. The formula $\mathcal{E}d\Phi$ denotes the property of properties of propositions to be a property of Φ in a state which at most differs from the state of evaluation with respect to the value of d. The difference with *DFL* resides solely in the type of objects abstracted over, i.e., properties of propositions in stead of propositions.

The *DFL(2)* notion of conjunction substantially differs from its *DFL* counterpart. In *DFL* the dynamic conjunction of Φ with Ψ consists, basically, in the application of Φ's extension to the intension of Ψ. This application still involves a form of conjunction of truth-conditional content because that has already been anticipated upon in the interpretation of the lift of type t to the type of context change potentials. In *DFL(2)* there is no notion of conjunction involved in the lift of a *DIL* formula, and, therefore, a form of conjunction of truth-conditional content shows up in the definition of dynamic conjunction. In the dynamic conjunction $[\Phi\,;\Psi]$ Φ is ascribed the property of being true in conjunction with Ψ, or, rather, the property of being true in conjunction with the proposition that Ψ has the property R, a property which is abstracted over. Since $[\Phi\,;\Psi]$ is defined to be true iff it has the property ${}^{\wedge}\lambda p\ {}^{\vee}p$ of propositions of being true, the conjunction of Φ with Ψ is true iff Φ has the property of being true in conjunction with the proposition that Ψ has the property of being true.

This notion of dynamic conjunction apparently reflects the classical notion of conjunction. In the classical case, one may say, what the conjunction of ϕ with ψ says about the proposition that ϕ is that it has the property of being true in conjunction with ψ, i.e., that the extension of $^\wedge\lambda p\ (^\vee p \wedge \psi)$ applies to the intension of ϕ. (Clearly, $^\vee(^\wedge\lambda p\ (^\vee p \wedge \psi))(^\wedge\phi) \Leftrightarrow (\phi \wedge \psi)$.) With the present definition of dynamic conjunction the function/argument structure is simply reversed. In the dynamic conjunction of Φ with Ψ, the property of propositions of being true in conjunction with Ψ is ascribed the property expressed by Φ, this, as is already indicated, under abstraction over further updates of Ψ.

It is quite easy to see that *DFL(2)* retains associativity, and, consequently, that it licenses the donkey equivalences. Furthermore *DFL(2)* displays the same monotonicity properties as *DFL* does. All *DFL(2)* formulas denote upward monotonic quantifiers, and this property guarantees that conjunction involves genuine update.[13] *DFL(2)* also licenses the reduction equivalences in fact 2.11 from chapter 1 which enable us to compute the contents of dynamically interpreted *DFL* expressions. As we will see shortly, *DFL* and *DFL(2)* assign sentences the same truth-conditions.

DMG and DMG(2)

As was indicated at the start of this section, except for the types and interpretation of the intermediary language *DFL*, the dynamic Montague grammar *DMG(2)* is the same as *DMG*. The categories and the syncategorematic constructions of the fragment remain the same. The function $f_{D(2)}$ from *DMG(2)* categories to *DFL(2)* types differs only from the corresponding *DMG* function since the *DFL(2)* type shift $^\uparrow$ is used, instead of the *DFL* type shift. The translations of basic expressions remain (typographically) the same, the difference residing solely in the associated types and the interpretation of the lift $\uparrow$, identity, negation, existential quantification and conjunction. The same holds for the translations of syncategorematic constructions.

I will not give a demonstration of how *DMG(2)* works since such a demonstration would be completely analogous to the demonstration in chapter 1 of how *DMG* works. As they are defined presently, *DMG(2)* and *DMG* are equivalent systems since *DFL* and *DFL(2)* assign corresponding sentences corresponding truth-conditions.

Let Φ' be the *DFL(2)* formula obtained from the *DFL* formula Φ by substituting every n-th variable T of the *DFL* type $^\uparrow a$ in Φ by the n-th variable T' of the corresponding *DFL(2)* type $^\uparrow a$. Moreover, let the superscript † denote the *DFL* interpretation of expressions and operators, and ‡ their *DFL(2)* interpretation. Then:

13. In the *DFL(2)* type of formulas, if Φ is upward monotonic then if $[\![R]\!] \subseteq [\![R']\!]$ (i.e., if $[\![^\vee R]\!]([\![p]\!]) = 1$ then $[\![^\vee R']\!]([\![p]\!]) = 1$) and $[\![^\vee\Phi]\!]([\![R]\!]) = 1$ then $[\![^\vee\Phi]\!]([\![R']\!]) = 1$. Now assume that Φ is upward monotonic and $[\Phi\,;\Psi]$ is true, i.e., $[\![\downarrow[\Phi\,;\Psi]]\!] = [\![^\vee\Phi]\!]([\![^\wedge\lambda p\ (^\vee p \wedge \downarrow\Psi)]\!]) = 1$. Clearly, if $[\![^\vee(^\wedge\lambda p\ (^\vee p \wedge \downarrow\Psi))]\!]([\![p]\!]) = 1$ then $[\![^\vee(^\wedge\lambda p\ ^\vee p)]\!]([\![p]\!]) = 1$. So, by the monotonicity of Φ, $[\![^\vee\Phi]\!]([\![^\wedge\lambda p\ ^\vee p]\!]) = [\![\downarrow\Phi]\!] = 1$, and, hence, Φ is true.

Fact 2.2 (Static equivalence of DFL and DFL(2))
For every *DFL* sentence Φ:
- $\downarrow^{\ddagger}\Phi'^{\ddagger} \Leftrightarrow \downarrow^{\dagger}\Phi^{\dagger}$

This fact implies that the *DMG* translation of a sentence in *DFL* has the same truth-conditions as the sentence's *DMG(2)* translation in *DFL(2)*. (The proof of this fact can be found in the appendix to this chapter.)

We can even prove something stronger. There is a function $\Downarrow$ that turns the *DFL(2)* interpretation of a formula Φ' (of type ${}^{\uparrow\ddagger}a$) into the *DFL* interpretation of Φ (of type ${}^{\uparrow\dagger}a$). This function is given by the following simultaneous recursive definition of $\Uparrow$, the lift from *DFL* objects of type ${}^{\uparrow}a$ to *DFL(2)* objects of type ${}^{\uparrow}a$, and $\Downarrow$, the shift back from *DFL(2)* objects to *DFL* objects (R and p are as above, q is a variable of type $\langle s, t\rangle$):

Definition 2.5 (Partial closure)
1. $\Uparrow\phi_{{}^{\uparrow\dagger}e} = \phi$
   $\Downarrow\Phi_{{}^{\uparrow\ddagger}e} = \Phi$
2. $\Uparrow\phi_{{}^{\uparrow\dagger}t} = {}^\wedge\lambda R\ {}^\vee\phi({}^\wedge\ {}^\vee R({}^\wedge\mathtt{true}))$ (R not free in ϕ)
   $\Downarrow\Phi_{{}^{\uparrow\ddagger}t} = {}^\wedge\lambda q\ {}^\vee\Phi({}^\wedge\lambda p\ ({}^\vee p \wedge {}^\vee q))$ (q not free in Φ)
3. $\Uparrow\phi_{{}^{\uparrow\dagger}\langle a,b\rangle} = \lambda x_{{}^{\uparrow\ddagger}a} \Uparrow(\phi(\Downarrow x))$ (x not free in ϕ)
   $\Downarrow\Phi_{{}^{\uparrow\ddagger}\langle a,b\rangle} = \lambda x_{{}^{\uparrow\dagger}a} \Downarrow(\Phi(\Uparrow x))$ (x not free in Φ)

Again we have arrow elimination:

Fact 2.3 ($\Downarrow\Uparrow$-elimination)
- $\Downarrow\Uparrow\Phi \Leftrightarrow \Phi$

Fact 2.4 shows that the partial closure $\Downarrow\Phi'$ of a *DFL(2)* expression Φ' in fact gives Φ's *DFL* interpretation:

Fact 2.4 (Dynamic equivalence of DFL and DFL(2))
For every *DFL* sentence Φ:
- $\Downarrow\Phi'^{\ddagger} \Leftrightarrow \Phi^{\dagger}$

The proof of this fact can also be found in the appendix to this chapter.

So, the two systems *DMG* and *DMG(2)* work out in precisely the same way in practice. The mere difference is that *DMG(2)* has invested in higher types of translation. In the following section (2.2) we will see that this investment pays off, since in this higher type an appropriate definition of dynamic negation can be given.

2.2 Dynamic negation in DMG(2)

In *DFL(2)* (and *DFL*) the context change potential of a formula Φ is realized, basically, by means of the conjunction of Φ with another formula Ψ. We have seen that the conjunction of Φ with Ψ, in *DFL(2)*, amounts to ascribing Φ the property of propositions of being true in conjunction with the proposition that Ψ has some property R which is abstracted over. Furthermore, we have seen that this notion of dynamic conjunction can be conceived of as a classical form of conjunction with a reversed function/argument structure. This analogy between *DFL(2)* conjunction and classical conjunction can be used as a heuristic device to determine the purported effect of the *DFL(2)* conjunction of the dynamic negation of Φ with Ψ and, consequently, to motivate the *DFL(2)* notion of dynamic negation, which is given below.

When we turn to the issue of dynamic negation in *DFL(2)*, it is expedient to focus on the question what the conjunction of the dynamic negation $\sim\Phi$ of Φ with Ψ should amount to, since context change potentials get realized in conjunctions, basically. So, if negation is *dynamic*, i.e., if in the conjunction of $\sim\Phi$ with Ψ the latter is subject to the context change potential of Φ, then the conjunction should involve the application of Φ, in some or other way, to the property of propositions of being true in conjunction with Ψ. Furthermore, if $\sim\Phi$ involves a genuine form of *negation*, it should involve the negation of Φ's truth conditional content, of course. Remind that these are two of the three requirements imposed on negation in section 1.3.

So, it appears, the conjunction of $\sim\Phi$ with Ψ must involve the negation of the result of applying Φ's context change potential, in some or other way, to the property of propositions of being true in conjunction with Ψ. Put differently, the conjunction of the negation of Φ with Ψ should consist in denying Φ to have some property of propositions which, in some or other way, is derived from the property of propositions of being true in conjunction with Ψ.

Thus conceived, if we want to know what the dynamic negation of Φ should be, we have to know, first, exactly what property of propositions Φ is denied to have if its negation is conjoined with Ψ and, second, how this property relates to the property of being true in conjunction with Ψ. Employing the analogy between dynamic and classical conjunction there is a simple answer to these questions.

What property of propositions is the proposition that ϕ denied to have in the classical conjunction $(\neg\phi \wedge \psi)$ of the negation of ϕ with ψ? As the following equivalence shows, this is the property of propositions of being true in disjunction with the negation of ψ, that is, the property $^\wedge\lambda p\ (^\vee p \vee \neg\psi)$:

$$\neg^\vee(^\wedge\lambda p\ (^\vee p \vee \neg\psi))(^\wedge\phi) \Leftrightarrow \neg(\phi \vee \neg\psi) \Leftrightarrow (\neg\phi \wedge \psi)$$

Next, how is the property $^\wedge\lambda p\ (^\vee p \wedge \psi)$, which the proposition that ϕ is said to have when it is conjoined with ψ, related to the property $^\wedge\lambda p\ (^\vee p \vee \neg\psi)$, which the proposition that ϕ is denied to have when its negation is conjoined with ψ? These properties are each other's dual.

The dual R^* of a property of propositions R is defined as follows:

Definition 2.6 (Dual)

- $R^* = {}^\wedge\lambda p\ \neg{}^\vee R({}^\wedge\ \neg{}^\vee p)$ (p not free in R)

The following equivalences show that the two properties of propositions ${}^\wedge\lambda p\ ({}^\vee p \wedge \psi)$ and ${}^\wedge\lambda p\ ({}^\vee p \vee \neg\psi)$ are indeed each other's dual:

$$({}^\wedge\lambda p\ ({}^\vee p \wedge \psi))^* \Leftrightarrow {}^\wedge\lambda p\ \neg(\neg{}^\vee p \wedge \psi) \Leftrightarrow {}^\wedge\lambda p\ ({}^\vee p \vee \neg\psi)$$
$$({}^\wedge\lambda p\ ({}^\vee p \vee \neg\psi))^* \Leftrightarrow {}^\wedge\lambda p\ \neg(\neg{}^\vee p \vee \neg\psi) \Leftrightarrow {}^\wedge\lambda p\ ({}^\vee p \wedge \psi)$$

Notice that, also more in general, the double dual of a property of propositions R is R again:

Fact 2.5 (Double dual)

- $R^{**} \Leftrightarrow R$[14]

To conclude this excurs, the (classical) conjunction of the (classical) negation of ϕ with ψ can be conceived of as denying the proposition that ϕ to have the property which is the dual of the property of being true in conjunction with ψ:

$$\neg{}^\vee({}^\wedge\lambda p\ ({}^\vee p \wedge \psi))^*({}^\wedge\phi) \Leftrightarrow \neg{}^\vee({}^\wedge\lambda p\ \neg(\neg{}^\vee p \wedge \psi))({}^\wedge\phi) \Leftrightarrow \neg\neg(\neg\phi \wedge \psi) \Leftrightarrow \neg\phi \wedge \psi$$

Returning now to our question what property of propositions Φ is denied to have when the negation of Φ is conjoined with Ψ, the analogy between *DFL(2)* conjunction and classical conjunction suggests that it must be the dual of the property of propositions of being true in conjunction with Ψ. Consequently, it appears, the dynamic negation of Φ itself must be defined to denote the property of properties of propositions of having a dual which Φ does not have. Here is its definition:

Definition 2.7 (Dynamic negation(2))

- $\sim\Phi = {}^\wedge\lambda R\ \neg{}^\vee\Phi(R^*)$ (R not free in Φ)

I will now show that the present notion of dynamic negation is the one we were looking for. It is relatively easily observed that the present notion of dynamic negation satisfies our initial demands (cf., section 1.3).

Dynamic negation was required, first, to involve negation of truth-conditional content, and on the present definition it does indeed:

Fact 2.6 (Truth-conditional effects of $\sim$)

- $\downarrow\sim\Phi \Leftrightarrow \neg\downarrow\Phi$[15]

This fact shows that the dynamic negation of Φ is true iff Φ is false. (Recall that this also holds for the static and the dynamic negation of *DFL*.) Observe, furthermore,

---

14. $R^{**} = {}^\wedge\lambda p\ \neg{}^\vee({}^\wedge\lambda p\ \neg{}^\vee R({}^\wedge\neg{}^\vee p))({}^\wedge\neg{}^\vee p) \Leftrightarrow {}^\wedge\lambda p\ \neg\neg{}^\vee R({}^\wedge\neg\neg{}^\vee p) \Leftrightarrow {}^\wedge\lambda p\ {}^\vee R({}^\wedge{}^\vee p) \Leftrightarrow R$.

15. Since $({}^\wedge\lambda p\ {}^\vee p)^* = {}^\wedge\lambda p\ \neg{}^\vee({}^\wedge\lambda p\ {}^\vee p)({}^\wedge\neg{}^\vee p) \Leftrightarrow {}^\wedge\lambda p\ \neg\neg{}^\vee p \Leftrightarrow {}^\wedge\lambda p\ {}^\vee p$, we find that $\downarrow\sim\Phi = {}^\vee({}^\wedge\lambda R\ \neg{}^\vee\Phi(R^*))({}^\wedge\lambda p\ {}^\vee p) \Leftrightarrow \neg{}^\vee\Phi({}^\wedge\lambda p\ {}^\vee p)^* \Leftrightarrow \neg{}^\vee\Phi({}^\wedge\lambda p\ {}^\vee p) = \neg\downarrow\Phi$.

that on the present definition the dynamic negation of the lift of a *DIL* formula ϕ equals the lift of the negation of ϕ:

Fact 2.7

- $\sim\uparrow\phi \Leftrightarrow \uparrow\neg\phi$[16]

The facts 2.6 and 2.7 show how intimately dynamic negation is related to static negation. Notice that the last fact 2.7 also holds for *DFL*'s static negation, but not for the dynamic negation of *DFL*.

In the second place, dynamic negation was required to be dynamic:

Fact 2.8 (Double negation)

- $\sim\sim\Phi \Leftrightarrow \Phi$[17]

As was observed in section 1, if a notion of negation obeys the law of double negation within a dynamic framework, then the notion of negation is truly dynamic. Fact 2.8 also implies that *DMG(2)*, with dynamic negation, accounts for the equivalence of the examples 1 and 2. The property of licensing the law of double negation distinguishes the dynamic negation of *DFL(2)* (and that of *DFL*) from the static negation of *DFL* (and that of *DFL(2)*).

In the third place, dynamic negation was required to preserve monotonicity:

Fact 2.9 (Monotonicity of $\sim$)

- If Φ is upward monotonic, then $\sim\Phi$ is upward monotonic[18]

Since the dynamic negation of *DFL(2)* preserves monotonicity, it follows that the conjunction of Φ with Ψ always involves information update. This fact shows the crucial difference between the dynamic negation of *DFL(2)* and that of *DFL* which reverses monotonicity.

We have found the dynamic negation that satisfies our initial requirements. How does it behave besides that? The following list of facts shows that it behaves as desired. Sticking to the usual definitions of universal quantification, implication and disjunction in terms of existential quantification, negation and conjunction, the associativity facts of extended *DMG* are retained:

Fact 2.10 (Extended associativity(2))

- $[\mathcal{E}d\Phi\,;\Psi] \quad \Leftrightarrow \mathcal{E}d[\Phi\,;\Psi]$

---

16. ${}^{\vee}R^{*}({}^{\wedge}\phi) = {}^{\vee}({}^{\wedge}\lambda p\ \neg{}^{\vee}R({}^{\wedge}\neg{}^{\vee}p))({}^{\wedge}\phi) \Leftrightarrow \neg{}^{\vee}R({}^{\wedge}\neg\phi)$, so $\sim\uparrow\phi = {}^{\wedge}\lambda R\ \neg{}^{\vee}({}^{\wedge}\lambda R\ {}^{\vee}R({}^{\wedge}\phi))(R^{*}) \Leftrightarrow {}^{\wedge}\lambda R\ \neg{}^{\vee}R^{*}({}^{\wedge}\phi) \Leftrightarrow {}^{\wedge}\lambda R\ \neg\neg{}^{\vee}R({}^{\wedge}\neg\phi) \Leftrightarrow {}^{\wedge}\lambda R\ {}^{\vee}R({}^{\wedge}\neg\phi) = \uparrow\neg\phi$.

17. $\sim\sim\Phi = {}^{\wedge}\lambda R\ \neg{}^{\vee}({}^{\wedge}\lambda R\ \neg{}^{\vee}\Phi(R^{*}))(R^{*}) \Leftrightarrow {}^{\wedge}\lambda R\ \neg\neg{}^{\vee}\Phi(R^{**}) \Leftrightarrow {}^{\wedge}\lambda R\ {}^{\vee}\Phi(R) \Leftrightarrow \Phi$ (Φ is intensionally closed, and R does not occur free in Φ).

18. Notice that if $[\![R]\!] \subseteq [\![R']\!]$, then $[\![R'^{*}]\!] \subseteq [\![R^{*}]\!]$. Now suppose that Φ is upward monotonic, i.e., if $[\![R]\!] \subseteq [\![R']\!]$ and $[\![{}^{\vee}\Phi]\!]([\![R]\!]) = 1$ then $[\![{}^{\vee}\Phi]\!]([\![R']\!]) = 1$. Next, suppose that $[\![{}^{\vee}\sim\Phi]\!]([\![R]\!]) = [\![{}^{\vee}({}^{\wedge}\lambda R\ \neg{}^{\vee}\Phi(R^{*}))(R)]\!] = 1$, i.e., that $[\![{}^{\vee}\Phi(R^{*})]\!] = 0$. Since $[\![R]\!] \subseteq [\![R']\!]$ and, hence, $[\![R'^{*}]\!] \subseteq [\![R^{*}]\!]$, by the monotonicity of Φ, $[\![{}^{\vee}\Phi(R'^{*})]\!] = 0$. So, $[\![{}^{\vee}({}^{\wedge}\lambda R\ \neg{}^{\vee}\Phi(R^{*}))(R')]\!] = [\![{}^{\vee}\sim\Phi]\!]([\![R']\!]) = 1$.

- $[\mathcal{A}d\Phi\,;\Psi] \Leftrightarrow \mathcal{A}d[\Phi\,;\Psi]$
- $[[\Phi\,;\Psi]\,;\Upsilon] \Leftrightarrow [\Phi\,;[\Psi\,;\Upsilon]]$
- $[[\Phi \Rightarrow \Psi]\,;\Upsilon] \Leftrightarrow [\Phi \Rightarrow [\Psi\,;\Upsilon]]$
- $[[\Phi \textbf{ or } \Psi]\,;\Upsilon] \Leftrightarrow [\Phi \textbf{ or } [\Psi\,;\Upsilon]]$

Fact 2.10 implies that *DMG(2)*, like extended *DMG*, gives a proper analysis of the donkey sentences from chapter 1, as well as of the examples 3, 4 and 6.

The following fact shows that dynamic negation, like *DFL*'s dynamic negation, interacts with the quantifiers and connectives in a classical way:

Fact 2.11 (Properties of $\sim$)

- $\sim\mathcal{E}d\Phi \Leftrightarrow \mathcal{A}d\sim\Phi$
- $\sim\mathcal{A}d\Phi \Leftrightarrow \mathcal{E}d\sim\Phi$
- $\sim[\Phi\,;\Psi] \Leftrightarrow [\Phi \Rightarrow \sim\Psi]$
- $\sim[\Phi \Rightarrow \Psi] \Leftrightarrow [\Phi\,;\sim\Psi]$
- $\sim[\Phi \textbf{ or } \Psi] \Leftrightarrow [\sim\Phi\,;\sim\Psi]$

The facts 2.7, 2.6, 2.8 and 2.11 show in fact that the present notion of dynamic negation behaves like its classical counterpart.

The next fact follows from the facts 2.8 and 2.11:

Fact 2.12 (Dual associativity)

- $[\sim\Phi\,;\Psi] \Leftrightarrow \sim[\Phi \textbf{ or } \sim\Psi]$ $(\not\Leftrightarrow \sim[\Phi\,;\Psi])$
- $[\sim\Phi \Rightarrow \Psi] \Leftrightarrow [\Phi \textbf{ or } \Psi]$
- $[\sim\Phi \textbf{ or } \Psi] \Leftrightarrow [\Phi \Rightarrow \Psi]$

These equivalences are classical as well. More important than the equivalence between $[\sim\Phi\,;\Psi]$ and $\sim[\Phi \textbf{ or } \sim\Psi]$ is the non-equivalence of $[\sim\Phi\,;\Psi]$ and $\sim[\Phi\,;\Psi]$. Unlike *DFL*'s dynamic negation (cf., fact 1.4), *DFL(2)*'s dynamic negation does not associate with conjunction. Notice that in *DFL*, too, $[\sim\Phi\,;\Psi]$ and $\sim[\Phi \textbf{ or } \sim\Psi]$ are equivalent, and, hence, that $[\Phi\,;\Psi] \Leftrightarrow [\Phi \textbf{ or } \sim\Psi]$ (which is not the case in *DFL(2)*).

Using the facts 2.10 and 2.12 it can be seen that *DMG(2)* also gives a satisfactory analysis of example 5 since this example is rendered equivalent with example 6. Finally, I simply note here that in *DFL(2)* the arrow elimination facts from *DFL* remain valid, cf., chapter 1, fact 2.11.

Summarizing the findings, the *DFL(2)* notion of dynamic negation satisfies the original requirements set out in section 1.3, it underlies the extended dynamic interpretation which is required for an analysis of the examples 1–6, and, apart from that, it behaves like classical negation. As we will see now, this same notion of negation also underlies a proper analysis of the examples 7–10. I show, in some more detail, the *DMG(2)* treatment of (simple variants of) these examples. I start with Groenendijk and Stokhof's example 7.

No player leaves. He stays

Application and reduction yields the following translation of 7:

$$\sim\mathcal{E}d_i[\uparrow\texttt{player}(d_i)\,;\uparrow\texttt{leave}(d_i)]\,;\uparrow\texttt{stay}(d_i)$$

The truth-conditions of this formula are determined as follows:

$$\begin{array}{l}
\downarrow[\sim\mathcal{E}d_i[\uparrow\texttt{player}(d_i)\,;\uparrow\texttt{leave}(d_i)]\,;\uparrow\texttt{stay}(d_i)] \Leftrightarrow \\
\downarrow[\mathcal{A}d_i\sim[\uparrow\texttt{player}(d_i)\,;\uparrow\texttt{leave}(d_i)]\,;\uparrow\texttt{stay}(d_i)] \Leftrightarrow \\
\downarrow[\mathcal{A}d_i[\uparrow\texttt{player}(d_i) \Rightarrow \sim\uparrow\texttt{leave}(d_i)]\,;\uparrow\texttt{stay}(d_i)] \Leftrightarrow \\
\downarrow\mathcal{A}d_i[\uparrow\texttt{player}(d_i) \Rightarrow [\sim\uparrow\texttt{leave}(d_i)\,;\uparrow\texttt{stay}(d_i)]] \Leftrightarrow \\
\downarrow\mathcal{A}d_i[\uparrow\texttt{player}(d_i) \Rightarrow [\uparrow\neg\texttt{leave}(d_i)\,;\uparrow\texttt{stay}(d_i)]] \Leftrightarrow \\
\forall d_i(\texttt{player}(d_i) \rightarrow (\neg\texttt{leave}(d_i) \wedge \texttt{stay}(d_i))) \Leftrightarrow \\
\forall x(\texttt{player}(x) \rightarrow (\neg\texttt{leave}(x) \wedge \texttt{stay}(x)))
\end{array}$$

(By means of the facts 2.11 (twice), 2.10, 2.6, and the facts 2.11 and 2.4 from chapter 1.) The example is true (or: satisfied) iff every player does not leave, but stays. Example 8 is analysed in an essentially similar way.

Either there is a bathroom downstairs, or it is upstairs.

(Notice that this example 9 is equivalent to example 10, *If it is not the case that there is a bathroom downstairs, then it is upstairs.*) In normal form, the translation of example 9 reads as follows:

$$\mathcal{E}d_i[\uparrow\texttt{bathroom}(d_i)\,;\uparrow\texttt{downstairs}(d_i)]\ \mathbf{or}\ \uparrow\texttt{upstairs}(d_i)$$

This formula has the following truth-conditions:

$$\begin{array}{l}
\downarrow[\mathcal{E}d_i[\uparrow\texttt{bathroom}(d_i)\,;\uparrow\texttt{downstairs}(d_i)]\ \mathbf{or}\ \uparrow\texttt{upstairs}(d_i)] \Leftrightarrow \\
\downarrow\sim[\sim\mathcal{E}d_i[\uparrow\texttt{bathroom}(d_i)\,;\uparrow\texttt{downstairs}(d_i)]\,;\sim\uparrow\texttt{upstairs}(d_i)] \Leftrightarrow \\
\downarrow\sim[\mathcal{A}d_i[\uparrow\texttt{bathroom}(d_i) \Rightarrow \sim\uparrow\texttt{downstairs}(d_i)]\,;\sim\uparrow\texttt{upstairs}(d_i)] \Leftrightarrow \\
\downarrow\sim\mathcal{A}d_i[\uparrow\texttt{bathroom}(d_i) \Rightarrow [\sim\uparrow\texttt{downstairs}(d_i)\,;\sim\uparrow\texttt{upstairs}(d_i)]] \Leftrightarrow \\
\downarrow\mathcal{E}d_i[\uparrow\texttt{bathroom}(d_i)\,;\sim[\sim\uparrow\texttt{downstairs}(d_i)\,;\sim\uparrow\texttt{upstairs}(d_i)]] \Leftrightarrow \\
\downarrow\mathcal{E}d_i[\uparrow\texttt{bathroom}(d_i)\,;[\uparrow\texttt{downstairs}(d_i)\ \mathbf{or}\ \uparrow\texttt{upstairs}(d_i)]] \Leftrightarrow \\
\exists d_i(\texttt{bathroom}(d_i) \wedge (\texttt{downstairs}(d_i) \vee \texttt{upstairs}(d_i))) \Leftrightarrow \\
\exists x(\texttt{bathroom}(x) \wedge (\texttt{downstairs}(x) \vee \texttt{upstairs}(x)))
\end{array}$$

(By means of the definition of **or**, the facts 2.11, 2.10 and 2.11 again, the definition of **or**, and the facts 2.11 and 2.4 from chapter 1.) The example is true iff there is a bathroom which is either downstairs or upstairs.

If a chessbox doesn't contain a spare pawn, it is taped on top of it.

The last example that I discuss has the following translation:

$$\mathcal{E}d_i[\uparrow\texttt{c\_box}(d_i)\,;\sim\mathcal{E}d_j[\uparrow\texttt{s\_pawn}(d_j)\,;\uparrow\texttt{in}(d_i)(d_j)]] \Rightarrow \uparrow\texttt{on}(d_i)(d_j)$$

The truth-conditions of this formula are determined as follows:

$$\downarrow[\mathcal{E}d_i[\uparrow\texttt{c\_box}(d_i)\,;\sim\!\mathcal{E}d_j[\uparrow\texttt{s\_pawn}(d_j)\,;\uparrow\texttt{in}(d_i)(d_j)]] \Rightarrow \uparrow\texttt{on}(d_i)(d_j)] \Leftrightarrow$$
$$\downarrow[\mathcal{E}d_i[\uparrow\texttt{c\_box}(d_i)\,;\mathcal{A}d_j[\uparrow\texttt{s\_pawn}(d_j) \Rightarrow \sim\!\uparrow\texttt{in}(d_i)(d_j)]] \Rightarrow \uparrow\texttt{on}(d_i)(d_j)] \Leftrightarrow$$
$$\downarrow\sim\![\mathcal{E}d_i[\uparrow\texttt{c\_box}(d_i)\,;\mathcal{A}d_j[\uparrow\texttt{s\_pawn}(d_j) \Rightarrow \sim\!\uparrow\texttt{in}(d_i)(d_j)]]\,;\sim\!\uparrow\texttt{on}(d_i)(d_j)] \Leftrightarrow$$
$$\downarrow\sim\!\mathcal{E}d_i[\uparrow\texttt{c\_box}(d_i)\,;\mathcal{A}d_j[\uparrow\texttt{s\_pawn}(d_j) \Rightarrow [\sim\!\uparrow\texttt{in}(d_i)(d_j)\,;\sim\!\uparrow\texttt{on}(d_i)(d_j)]]] \Leftrightarrow$$
$$\downarrow\mathcal{A}d_i[\uparrow\texttt{c\_box}(d_i) \Rightarrow \sim\!\mathcal{A}d_j[\uparrow\texttt{s\_pawn}(d_j) \Rightarrow [\sim\!\uparrow\texttt{in}(d_i)(d_j)\,;\sim\!\uparrow\texttt{on}(d_i)(d_j)]]] \Leftrightarrow$$
$$\downarrow\mathcal{A}d_i[\uparrow\texttt{c\_box}(d_i) \Rightarrow \mathcal{E}d_j[\uparrow\texttt{s\_pawn}(d_j)\,;\sim\![\sim\!\uparrow\texttt{in}(d_i)(d_j)\,;\sim\!\uparrow\texttt{on}(d_i)(d_j)]]] \Leftrightarrow$$
$$\downarrow\mathcal{A}d_i[\uparrow\texttt{c\_box}(d_i) \Rightarrow \mathcal{E}d_j[\uparrow\texttt{s\_pawn}(d_j)\,;[\uparrow\texttt{in}(d_i)(d_j) \textbf{ or } \uparrow\texttt{on}(d_i)(d_j)]]] \Leftrightarrow$$
$$\forall d_i(\texttt{c\_box}(d_i) \rightarrow \exists d_j(\texttt{s\_pawn}(d_j) \wedge (\texttt{in}(d_i)(d_j) \vee \texttt{on}(d_i)(d_j)))) \Leftrightarrow$$
$$\forall x(\texttt{c\_box}(x) \rightarrow \exists y(\texttt{s\_pawn}(y) \wedge (\texttt{in}(x)(y) \vee \texttt{on}(x)(y))))$$

(By means of the fact 2.11, the definition of $\Rightarrow$, fact 2.10, fact 2.11 again (twice), the definition of **or**, and the facts 2.11 and 2.4 from chapter 1.) The example is true iff every chessbox comes with a spare pawn which is either inside of the box, or taped to the top of it.

To end this section, we may conclude that *DMG(2)* gives a uniform and compositional treatment of the anaphoric relationships in the examples 1–10 solely in terms of the *DFL(2)* lift of *DIL* constants, application, abstraction and the dynamic notions of negation, existential quantification and conjunction. Like in extended *DMG*, the dynamic notion of negation brings along dynamic notions of universal quantification, implication and disjunction in terms of which these examples are satisfactorily analyzed. Unlike the extended *DMG* notion of dynamic negation, the *DMG(2)* notion of negation preserves the upward monotonicity of all *DFL* formulas, and, hence, ensures that the interpretation of successive sentences involves information update. Consequently, the examples at issue are analyzed without a need for the constraints which extended *DMG* imposes in order to guarantee sound interpretation, and without making further adjustments (by introducing ambiguities) in order to be able always to comply with these constraints.

2.3 DMG, extended DMG and DMG(2)

Let us take stock. In chapter 1, section 3.2 I have sketched Groenendijk and Stokhof's *DMG* and in section 2.1 of the present chapter I have proposed an alternative but equivalent system *DMG(2)*. Furthermore, I have considered Groenendijk and Stokhof's extended dynamic version of *DMG* (section 1.1 of this chapter), and, finally, I have proposed my own extended dynamic version of *DMG(2)* (section 2.2 of this chapter). So, three different systems have passed in revue. In order to conclude this chapter, I show the characteristic properties of the three systems concerned (i.e., *DFL*, extended *DFL*, and *DFL(2)*) in terms of the (different) ways in which closed expressions in the three systems can be reduced to ordinary *DIL* expressions.

I will define three reduction algorithms $^\diamondsuit$, $^\heartsuit$ and $^\spadesuit$ for *DFL*, extended *DFL*, and *DFL(2)*, respectively. For any *DFL* or *DFL(2)* expression Φ without free variables, these algorithms generate a *DIL* formula that specifies the static contents of Φ. I represent the *DFL* (and *DFL(2)*) language here as having two primitive forms of quantification ($\mathcal{E}d$ and $\mathcal{A}d$) and two sentential connectives (; and $\Rightarrow$). This gives some redundancy in the definition of the algorithms, but it facilitates their practical application.

I start with the reduction algorithm $^{\Diamond}$ for *DFL*.

Definition 2.8 (DFL reduction)

1. $(\uparrow\phi)^{\Diamond} \quad = \phi$
2. $(\beta(\alpha))^{\Diamond} \quad = ((\beta(\alpha))^{\circ})^{\Diamond}$
3. $(\lambda T_{\uparrow a}\beta)^{\Diamond} \quad = \lambda T'_a\ ([\uparrow T'/T]\beta)^{\Diamond}$ (T' not free in Φ)
4. $(\alpha \tilde{=} \beta)^{\Diamond} \quad = (\alpha^{\Diamond} = \beta^{\Diamond})$
5. $(\sim\Phi)^{\Diamond} \quad = \neg(\Phi)^{\Diamond}$
6. $(\mathcal{E}d\Phi)^{\Diamond} \quad = \exists d(\Phi)^{\Diamond}$
   $(\mathcal{A}d\Phi)^{\Diamond} \quad = \forall d(\Phi)^{\Diamond}$
7. $[\uparrow\phi\,;\Psi]^{\Diamond} \quad = ((\uparrow\phi)^{\Diamond} \wedge \Psi^{\Diamond})$
   $[\uparrow\phi \Rightarrow \Psi]^{\Diamond} \quad = ((\uparrow\phi)^{\Diamond} \rightarrow \Psi^{\Diamond})$
8. $[\beta(\alpha)\,;\Psi]^{\Diamond} \quad = [(\beta(\alpha))^{\circ}\,;\Psi]^{\Diamond}$
   $[\beta(\alpha) \Rightarrow \Psi]^{\Diamond} \quad = [(\beta(\alpha))^{\circ} \Rightarrow \Psi]^{\Diamond}$
9. $[(\alpha \tilde{=} \beta)\,;\Psi]^{\Diamond} \quad = ((\alpha \tilde{=} \beta)^{\Diamond} \wedge \Psi^{\Diamond})$
   $[(\alpha \tilde{=} \beta) \Rightarrow \Psi]^{\Diamond} \quad = ((\alpha \tilde{=} \beta)^{\Diamond} \rightarrow \Psi^{\Diamond})$
10. $[\sim\Phi\,;\Psi]^{\Diamond} \quad = ((\sim\Phi)^{\Diamond} \wedge \Psi^{\Diamond})$
    $[\sim\Phi \Rightarrow \Psi]^{\Diamond} \quad = ((\sim\Phi)^{\Diamond} \rightarrow \Psi^{\Diamond})$
11. $[\mathcal{E}d\Phi\,;\Psi]^{\Diamond} \quad = (\mathcal{E}d[\Phi\,;\Psi])^{\Diamond}$
    $[\mathcal{E}d\Phi \Rightarrow \Psi]^{\Diamond} \quad = (\mathcal{A}d[\Phi \Rightarrow \Psi])^{\Diamond}$
12. $[\mathcal{A}d\Phi\,;\Psi]^{\Diamond} \quad = ((\mathcal{A}d\Phi)^{\Diamond} \wedge \Psi^{\Diamond})$
    $[\mathcal{A}d\Phi \Rightarrow \Psi]^{\Diamond} \quad = ((\mathcal{A}d\Phi)^{\Diamond} \rightarrow \Psi^{\Diamond})$
13. $[[\Phi\,;\Psi]\,;\Upsilon]^{\Diamond} \quad = [\Phi\,;[\Psi\,;\Upsilon]]^{\Diamond}$
    $[[\Phi\,;\Psi] \Rightarrow \Upsilon]^{\Diamond} \quad = [\Phi \Rightarrow [\Psi \Rightarrow \Upsilon]]^{\Diamond}$
14. $[[\Phi \Rightarrow \Psi]\,;\Upsilon]^{\Diamond} \quad = ([\Phi \Rightarrow \Psi]^{\Diamond} \wedge \Upsilon^{\Diamond})$
    $[[\Phi \Rightarrow \Psi] \Rightarrow \Upsilon]^{\Diamond} \quad = ([\Phi \Rightarrow \Psi]^{\Diamond} \rightarrow \Upsilon^{\Diamond})$

where $((\uparrow\beta)(\alpha))^{\circ} \quad = \uparrow(\beta(\alpha^{\Diamond}))$
$((\lambda T\ \beta)(\alpha))^{\circ} = [\alpha/T]\beta$ if all free variables in α are free for T in β
$(\beta(\alpha_1)(\alpha_2))^{\circ} = ((\beta(\alpha_1))^{\circ}(\alpha_2))^{\circ}$

The (simultaneously defined) auxiliary algorithm $^{\circ}$ normalizes application structures of the form $\beta(\alpha)$. In such application structures β is either a raised term (a raised constant, variable or application structure), or a λ-term, or an application structure itself. (β can not be a plain variable since we only consider the reduction of *DFL* expressions without free variables. However, it can be a raised free variable $\uparrow T$ which is introduced in the reduction of λ-terms.)

In the first case the application of the lift of β to α is replaced by the lift of the application of β to the reduction of α. So, the lifting operator $\uparrow$ is exported, and can be subject to subsequent $^{\diamondsuit}$ reduction. In the second case λ-conversion takes place. (If some free variables in α are not free for T in β, we may use an alphabetical variant of β.) In the third case, the application structure $\beta(\alpha_1)$ is normalized first, and the application of this normalized structure to α_2 is normalized next. Notice that in any case the application of $^{\circ}$ produces a normalized expression which is subject to further $^{\diamondsuit}$ reduction.

Inspecting the clauses of definition 2.8, we see that in cases of genuine reduction either the lifting operator disappears (clause 1), or abstraction over objects of a raised type $\upharpoonright a$ is replaced by abstraction over objects of type a (clause 3), or dynamic operators are replaced by their static counterparts (clauses 4–7, 9, 10, 12 and 14). In the remaining clauses we either find a normalization of application structures by means of the auxiliary algorithm $^{\circ}$ (clauses 2 and 8), or applications of the *DFL* associativity facts, which are used to put a formula in its so-called 'normal binding form' (clauses 11 and 13).

The following facts tells us that the algorithm $^{\diamondsuit}$ effectively determines the static contents of the *DFL* expression which it is applied to:

Fact 2.13 (DFL closure)
For every *DFL* expression Φ without free variables:

- $\downarrow\Phi \Leftrightarrow \Phi^{\diamondsuit}$

I now turn to the reduction algorithm $^{\heartsuit}$ for extended *DFL*. Apart from three clauses its definition equals that of the reduction algorithm for *DFL*.

Definition 2.9 (Extended DFL reduction)
$^{\heartsuit}$ is like $^{\diamondsuit}$ except for the clauses 10, 12 and 14:

10. $[\sim\Phi\,;\Psi]^{\heartsuit} = (\sim[\Phi\,;\Psi])^{\heartsuit}$
    $[\sim\Phi \Rightarrow \Psi]^{\heartsuit} = (\sim[\Phi \Rightarrow \Psi])^{\heartsuit}$
12. $[\mathcal{A}d\Phi\,;\Psi]^{\heartsuit} = (\mathcal{A}d[\Phi\,;\Psi])^{\heartsuit}$
    $[\mathcal{A}d\Phi \Rightarrow \Psi]^{\heartsuit} = (\mathcal{E}d[\Phi \Rightarrow \Psi])^{\heartsuit}$
14. $[[\Phi \Rightarrow \Psi]\,;\Upsilon]^{\heartsuit} = [\Phi \Rightarrow [\Psi\,;\Upsilon]]^{\heartsuit}$
    $[[\Phi \Rightarrow \Psi] \Rightarrow \Upsilon]^{\heartsuit} = [\Phi\,;[\Psi \Rightarrow \Upsilon]]^{\heartsuit}$

Fact 2.14 (Extended DFL closure)
For every extended *DFL* expression Φ without free variables:

- $\downarrow\Phi \Leftrightarrow \Phi^{\heartsuit}$

So, the characteristic difference between *DMG* and extended *DMG* lies in the associativity of negation, universal quantification and implication with conjunction and implication. As was argued in section 1, the associativity of the last two operators ($\mathcal{A}d$ and $\Rightarrow$) is well-motivated, but that of the first ($\sim$) is problematic.

I now turn to the reduction algorithm $^{\spadesuit}$ for extended *DFL*. Apart from the clause for the reduction of the conjunction of the negation of a formula Φ with a formula Ψ, its definition equals that of the reduction algorithm for extended *DFL*. The most perspicuous reduction is obtained by employing an $\sim$-elimination scheme $^{\bullet}$:

Definition 2.10 (DFL(2) reduction)
$^{\spadesuit}$ is like $^{\heartsuit}$ except for the following clause:

10. $[\sim\Phi\,;\Psi]^{\spadesuit} = ([(\sim\Phi)^{\bullet}\,;\Psi])^{\spadesuit}$
$[\sim\Phi \Rightarrow \Psi]^{\spadesuit} = ([(\sim\Phi)^{\bullet} \Rightarrow \Psi])^{\spadesuit}$

where

- $(\sim\uparrow\phi)^{\bullet} = \uparrow\neg\phi$
- $(\sim\beta(\alpha))^{\bullet} = (\sim(\beta(\alpha))^{\star})^{\bullet}$
- $(\sim(\alpha \,\tilde{=}\, \beta))^{\bullet} = \uparrow\neg((\alpha \,\tilde{=}\, \beta)^{\spadesuit})$
- $(\sim\sim\Phi)^{\bullet} = \Phi$
- $(\sim\mathcal{E}d\Phi)^{\bullet} = \mathcal{A}d(\sim\Phi)^{\bullet}$
  $(\sim\mathcal{A}d\Phi)^{\bullet} = \mathcal{E}d(\sim\Phi)^{\bullet}$
- $(\sim[\Phi\,;\Psi])^{\bullet} = [\Phi \Rightarrow (\sim\Psi)^{\bullet}]$
  $(\sim[\Phi \Rightarrow \Psi])^{\bullet} = [\Phi\,;(\sim\Psi)^{\bullet}]$
  where * is like $^{\circ}$ except that $((\uparrow\beta)(\alpha))^{\star} = \uparrow(\beta(\alpha^{\spadesuit}))$, of course.

Fact 2.15 (DFL(2) closure)
For every *DFL(2)* expression Φ without free variables:

- $\downarrow\Phi \Leftrightarrow \Phi^{\spadesuit}$

Observe that the negation elimination scheme displays a classical pattern. So, characteristic of *DMG(2)* are just the associativity facts (cf., clauses 11 in definition 2.8, preserved in definition 2.10) and the extended associativity facts (cf., clauses 12 and 14 in definition 2.9, also preserved in definition 2.10). The last, especially, are validated without allowing negation to associate with conjunction.

***

We have seen that *DMG* gives a fully compositional reformulation of basic aspects of *DRT*. For as far as donkey anaphora or, more in general, intersentential relationships between indefinite noun phrases and pronouns are concerned, *DMG* shows there is no need to resort to an intermediary level of representation. Furthermore, in *DMG* the topic of extended dynamic interpretation quite naturally arises, but, as is argued in the section 1.2 and 1.3, the framework does not allow a fully satisfactory treatment of it. The phenomena of extended dynamic interpretation at issue seem to require a notion of dynamic negation, but in *DMG* such a notion appears to generate unwanted monotonicity properties, and, as a consequence, it yields information downdate in cases where information update would be appropriate. For that reason, the resulting system has to be kept under control by constraints.

What I have shown is that, basically, a reformulation of *DMG* in a higher type gives us just the amount of elbow room required for an adequate definition of dynamic negation with the right monotonicity properties. The proposed notion of negation has been studied to some extent and it has been shown that it generates the purported dynamic interpretation of the other (derived) sentential operators. Apart from that it appears to behave classically.

Of course we are not done yet. As already was indicated at the start of section 1.2, the extended dynamic interpretation of the derived operators is not their most usual one, and it remains an issue what factors in fact trigger their extended dynamic interpretation. I also indicated there that for our present concerns we may content ourselves with unraveling the logic of the extended dynamics, and, furthermore, with the fact that the classical, static interpretation of the logical operators at issue remains definable in terms of their dynamic interpretation and the static closure operation. Still, the triggering issue can not be neglected in a full-blown extended dynamic semantics. Without going into much detail, I will conclude with some examples where the issue is most pressing.

In *DMG(2)* we have recovered the classical interdefinability of the sentential operators and quantifiers, and this, it seems, is well motivated by the interdefinability phenomena in natural language which appear to be preserved also when extended dynamic interpretation is concerned. For instance, this interdefinability correctly predicts the equivalence between the bathroom disjunction 5 and the bathroom implication 6 and it also serves to give a single explanation of the dynamics of the natural language determiners *no* and *every* as displayed in, for instance, the examples 4 and 7 respectively. However, there appears to be a limit to this interdefinability in natural language.

Consider the following examples:

(11) If there is no bathroom downstairs, then it is upstairs.

(12) If every bathroom is not downstairs, then it is upstairs.

In *DMG(2)*, and in extended *DMG*, the two sentences are equivalent. Still, intuitively they do not seem (fully) equivalent. In example 11, the pronoun *it* appears to trigger an extended dynamic interpretation of the antecedent of the conditional and the example therefore entails the existence of a bathroom which, if it is not downstairs, is upstairs. However, example 12 does not seem to entail anything about the existence of bathrooms. It even seems impossible to establish a sensible anaphoric relationship in example 12.[19] In other words, in examples such as 11 and 12, the

19. This has already been observed in Egli [1979, pp. 275-276]. Examples like these are also discussed

classical interdefinability of the quantifiers seems not to be reflected by linguistic facts.

Something similar goes for the following pair of examples:

(13) There is a boy who did not pass the exam. He failed on TEX-tronics.
(14) Not every boy passed the exam. He failed on TEX-tronics.

The first example of this pair displays an ordinary anaphoric relationship between the pronoun *he* and the indefinite noun phrase *a boy*. However, in the second example, which is predicted to be equivalent, the anaphoric relationship seems impossible.

It appears that the factors that trigger extended dynamic interpretation distinguish between quantified structures which are traditionally conceived of as equivalent, or, alternatively, that allegedly equivalent quantified structures display different extended dynamic behaviour. The factors which trigger or suppress (extended) dynamic interpretation enable an anaphoric reading of the examples 11 and 13, whereas they prohibit an equivalent reading of the examples 12 and 14, respectively. So, in view of examples such as 11–14, the rigid interdefinability of the quantifiers and operators may have to be relaxed, in some or other way. However, since, as is already remarked, a treatment of the triggering issue falls beyond the scope of this chapter, a further investigation of this topic must be left for another occasion.

***

As for a final conclusion, I think we may say that the extended dynamic enterprise did pay its dividends. At least a principled and systematic semantic account has been given of a range of phenomena that at first glance might seem considerably intractable. Furthermore, if it were not inspired by the compositional treatment of anaphoric relationships in Groenendijk and Stokhof's *DMG*, I think we would have had a hard time to run up against the systematic behaviour that the extended dynamics does display.

Appendix

In the proofs of some facts stated in this chapter, I use the superscript † to indicate the interpretation of an expression or operator in *DFL* and ‡ to refer to its interpretation in *DFL(2)*. In equivalences *df* means 'by definition', *rd* means 'by reduction' and *ih* means 'by induction hypothesis'.

I start with a proof of the arrow elimination facts.

Facts 1.2.5, 2.1, 2.3

- $\downarrow\uparrow\phi \Leftrightarrow \phi$ where $\downarrow\uparrow$ is either $\downarrow^\dagger\uparrow^\dagger$, $\downarrow^\ddagger\uparrow^\ddagger$ or $\Downarrow\Uparrow$

---

in Heim [1982, p. 58ff] and Roberts [1989, p. 703].

These facts are proven by induction on the type of ϕ. Basic cases:

$$\downarrow^\dagger\uparrow^\dagger\alpha_e =_{df} {}^{\vee\wedge}\alpha =_{df} \downarrow^\ddagger\uparrow^\ddagger\alpha_e =_{df} \Downarrow\Uparrow\alpha_e$$

$$\downarrow^\dagger\uparrow^\dagger\phi_t =_{df} {}^\vee({}^\wedge\lambda p(\phi \wedge {}^\vee p))({}^\wedge\mathtt{true}) \Leftrightarrow_{rd} \phi \wedge \mathtt{true} \Leftrightarrow_{rd} \phi$$
$$\downarrow^\ddagger\uparrow^\ddagger\phi_t =_{df} {}^\vee({}^\wedge\lambda R\ {}^\vee R({}^\wedge\phi))({}^\wedge\lambda p\ {}^\vee p) \Leftrightarrow_{rd} (\lambda p\ {}^\vee p)({}^\wedge\phi) \Leftrightarrow_{rd} \phi$$
$$\Downarrow\Uparrow\phi_{\uparrow^\dagger t} =_{df} {}^\wedge\lambda q^\vee({}^\wedge\lambda R^\vee\phi({}^{\wedge\vee}R({}^\wedge\mathtt{true})))({}^\wedge\lambda p({}^\vee p \wedge {}^\vee q)) \Leftrightarrow_{rd} {}^\wedge\lambda q^\vee\phi({}^{\wedge\vee}q) \Leftrightarrow^* \phi$$

(* ϕ is intensionally closed; q is not free in ϕ.)

The induction steps, the same in all three cases, look as follows:

$$\downarrow\uparrow\phi_{\langle a,b\rangle} =_{df} \lambda x_a \downarrow\uparrow(\phi(\downarrow\uparrow x)) \Leftrightarrow_{ih} \phi$$

The following fact is used in the proof of the equivalence of *DFL* and *DFL(2)*:

Fact 2.16

- $\Downarrow\uparrow^\ddagger\phi \Leftrightarrow \uparrow^\dagger\phi$
  $\downarrow^\ddagger\Uparrow\Phi \Leftrightarrow \downarrow^\dagger\Phi$

This fact is proved by simultaneous induction on the types a and ${}^{\uparrow\dagger}a$ of ϕ and Φ, respectively. Basic case for the types t and ${}^{\uparrow\dagger}t$ (for the types e and ${}^{\uparrow\dagger}e$ ($= \langle s, e\rangle$) the proof is straightforward):

$$\Downarrow\uparrow^\ddagger\phi_t \quad =_{df} {}^\wedge\lambda q\ {}^{\vee\wedge}(\lambda R\ {}^\vee R({}^\wedge\phi))({}^\wedge\lambda p\ ({}^\vee p \wedge {}^\vee q)) \Leftrightarrow_{rd} {}^\wedge\lambda q\ (\phi \wedge {}^\vee q) =_{df} \uparrow^\dagger\phi$$
$$\downarrow^\ddagger\Uparrow\Phi_{\uparrow^\dagger t} =_{df} {}^\vee({}^\wedge\lambda R\ {}^\vee\Phi({}^\wedge\ {}^\vee R({}^\wedge\mathtt{true})))({}^\wedge\lambda p\ {}^\vee p) \quad \Leftrightarrow_{rd} {}^\vee\Phi({}^\wedge\mathtt{true}) \quad =_{df} \downarrow^\dagger\Phi$$

Induction:

$$\Downarrow\uparrow^\ddagger\phi_{\langle a,b\rangle} \quad =_{df} \lambda x \Downarrow\uparrow^\ddagger(\phi(\downarrow^\ddagger\Uparrow x)) \Leftrightarrow_{ih} \lambda x \uparrow^\dagger(\phi(\downarrow^\dagger x)) =_{df} \uparrow^\dagger\phi$$
$$\downarrow^\ddagger\Uparrow\Phi_{\uparrow^\dagger\langle a,b\rangle} =_{df} \lambda x \downarrow^\ddagger\Uparrow(\Phi(\Downarrow\uparrow^\ddagger x)) \Leftrightarrow_{ih} \lambda x \downarrow^\dagger(\Phi(\uparrow^\dagger x)) =_{df} \downarrow^\dagger\Phi$$

Now I turn to the proof of the equivalence facts 2.2 and 2.4:

Facts 2.2, 2.4

For every *DFL* sentence Φ:

- $\downarrow^\ddagger\Phi'^\ddagger \Leftrightarrow \downarrow^\dagger\Phi^\dagger$
- $\Downarrow\Phi'^\ddagger \Leftrightarrow \Phi^\dagger$

where Φ' is obtained from Φ by substituting every n-th variable T of type ${}^{\uparrow\dagger}a$ in Φ by the n-th variable T' of type ${}^{\uparrow\ddagger}a$

Although the facts are stated for *sentences*, they will be proved for arbitrary expressions without free variables. Furthermore, since free variables do appear in the induction used in the proof, something even stronger has to be proved. What in fact needs to be proved are the following equivalences:

- $\downarrow^\ddagger\Phi^{\text{`}\prime\ddagger} \Leftrightarrow \downarrow^\dagger\Phi^{\text{`}\dagger}$
- $\Downarrow\Phi^{\text{`}\prime\ddagger} \Leftrightarrow \Phi^{\text{`}\dagger}$

  where Φ‘ is obtained from Φ by

– substituting every n-th *free* variable V of type ${}^{\uparrow\dagger}a$ in Φ by $\uparrow U$, where U is the n-th variable of type a if n is odd

and where $\Phi^{`\prime}$ is obtained from $\Phi^{`}$ by

– substituting every n-th *free* variable T of type ${}^{\uparrow\dagger}a$ in $\Phi^{`}$ by $\Uparrow T$ if n is even

– substituting every other n-th variable T of type ${}^{\uparrow\dagger}a$ in $\Phi^{`}$ by the n-th variable T' of type ${}^{\uparrow\ddagger}a$

Typically, the inductive proof that $\downarrow^{\ddagger}(\lambda V\Phi)^{`\prime\ddagger} \Leftrightarrow \downarrow^{\dagger}(\lambda V\Phi)^{`\dagger}$ uses the induction hypothesis that $\downarrow^{\ddagger}\Phi^{`\prime\ddagger} \Leftrightarrow \downarrow^{\dagger}\Phi^{`\dagger}$, where free occurrences of the variable V in Φ are replaced by $\uparrow U$ in $\Phi^{`\dagger}$ and $\Phi^{`\prime\ddagger}$. So, in that case V (and U) are assumed to be odd numbered variables. In case they are even numbered, the proof proceeds by replacing them by odd numbered variables first.

Similarly, the inductive proof that $\Downarrow(\lambda T\Phi)^{`\prime\ddagger} \Leftrightarrow (\lambda T\Phi)^{`\dagger}$ uses the induction hypothesis that $\Downarrow\Phi^{`\prime\ddagger} \Leftrightarrow \Phi^{`\dagger}$, where free occurrences of the variable T in Φ (and $\Phi^{`\dagger}$) are replaced by $\Uparrow T$ in $\Phi^{`\prime\ddagger}$. So, in that case T is assumed to be an even numbered variable. In case it is odd numbered, the proof proceeds by replacing it by an even numbered variable first.

The proof proceeds by (simultaneous) induction on the complexity of Φ. Basic cases (the variable T is assumed to be even numbered; in case it is numbered odd, then $T^{`}$ is $\uparrow T$, and the proof is as in the case of a constant `con`):

$$\downarrow^{\ddagger}(\uparrow\mathtt{con})^{`\prime\ddagger} =_{df} \downarrow^{\ddagger}\uparrow^{\ddagger}\mathtt{con} \Leftrightarrow_{1.2.5,2.1} \downarrow^{\dagger}\uparrow^{\dagger}\mathtt{con} =_{df} \downarrow^{\dagger}(\uparrow\mathtt{con})^{`\dagger}$$
$$\Downarrow(\uparrow\mathtt{con})^{`\prime\ddagger} =_{df} \Downarrow\uparrow^{\ddagger}\mathtt{con} \Leftrightarrow_{2.16} \uparrow^{\dagger}\mathtt{con} =_{df} (\uparrow\mathtt{con})^{`\dagger}$$

$$\downarrow^{\ddagger}(T)^{`\prime\ddagger} =_{df} \downarrow^{\ddagger}\Uparrow T \Leftrightarrow_{2.16} \downarrow^{\dagger}T =_{df} \downarrow^{\dagger}(T)^{`\dagger}$$
$$\Downarrow(T)^{`\prime\ddagger} =_{df} \Downarrow\Uparrow T \Leftrightarrow_{2.3} T =_{df} (T)^{`\dagger}$$

As concerns application structures, I assume that Φ is in normal form, i.e., that all λ-conversions have been executed (possibly after renaming of variables). Therefore, the head functor in an application structure is either a raised constant or a (raised) variable. I restrict attention to singular application structures (having the form $\beta(\alpha)$) since the generalization to multiple application structures (having the form $\beta(\alpha_1)\ldots(\alpha_n)$) is straightforward. The variable T is assumed to be even numbered. In case T is odd, $(T(\alpha))^{`} = \uparrow T(\alpha')$. Induction for application structures:

$$\begin{aligned}
\downarrow^{\ddagger}(\uparrow\mathtt{con}(\alpha))^{`\prime\ddagger} &=_{df} \downarrow^{\ddagger}\uparrow^{\ddagger}\mathtt{con}(\downarrow^{\ddagger}\alpha^{`\prime\ddagger}) =_{ih} \downarrow^{\ddagger}\uparrow^{\ddagger}\mathtt{con}(\downarrow^{\dagger}\alpha^{`\dagger}) \Leftrightarrow_{1.2.5,2.1}\\
&\phantom{=_{df}\ } \downarrow^{\dagger}\uparrow^{\dagger}\mathtt{con}(\downarrow^{\dagger}\alpha^{`\dagger}) =_{df} \downarrow^{\dagger}(\uparrow\mathtt{con}(\alpha))^{`\dagger}\\
\Downarrow(\uparrow\mathtt{con}(\alpha))^{`\prime\ddagger} &=_{df} \Downarrow\uparrow^{\ddagger}(\mathtt{con}(\downarrow^{\ddagger}\alpha^{`\prime\ddagger})) \Leftrightarrow_{ih} \Downarrow\uparrow^{\ddagger}(\mathtt{con}(\downarrow^{\dagger}\alpha^{`\dagger})) \Leftrightarrow_{2.16}\\
&\phantom{=_{df}\ } \uparrow^{\dagger}(\mathtt{con}(\downarrow^{\dagger}\alpha^{`\dagger})) =_{df} (\uparrow\mathtt{con}(\alpha))^{`\dagger}
\end{aligned}$$

$$\begin{aligned}
\downarrow^{\ddagger}(T(\alpha))^{`\prime\ddagger} &=_{df} \downarrow^{\ddagger}(\Uparrow T(\alpha^{`\prime\ddagger})) =_{df} \downarrow^{\ddagger}\Uparrow(T(\Downarrow\alpha^{`\prime\ddagger})) \Leftrightarrow_{ih}\\
&\phantom{=_{df}\ } \downarrow^{\ddagger}\Uparrow(T(\alpha^{`\dagger})) \Leftrightarrow_{1.2.5,2.1} \downarrow^{\dagger}T(\alpha^{`\dagger}) =_{df} \downarrow^{\dagger}(T(\alpha))^{`\dagger}\\
\Downarrow(T(\alpha))^{`\prime\ddagger} &=_{df} \Downarrow(\Uparrow T(\alpha^{`\prime\ddagger})) =_{df} \Downarrow\Uparrow(T(\Downarrow\alpha^{`\prime\ddagger})) \Leftrightarrow_{ih}\\
&\phantom{=_{df}\ } \Downarrow\Uparrow(T(\alpha^{`\dagger})) \Leftrightarrow_{2.3} T(\alpha^{`\dagger}) =_{df} (T(\alpha))^{`\dagger}
\end{aligned}$$

In the induction for λ-terms, T is assumed to be numbered even and V odd. If V is the n-th variable of type ${}^{\uparrow\dagger}a$, then U is the n-th variable of type a. Induction:

$$\downarrow^{\ddagger}(\lambda V\ \Phi)^{\text{`}\prime\ddagger} \Leftrightarrow_1 \lambda U \downarrow^{\ddagger}(\Phi)^{\text{`}\prime\ddagger} \Leftrightarrow_{ih} \lambda U \downarrow^{\dagger}(\Phi)^{\text{`}\dagger} \Leftrightarrow_2 \downarrow^{\dagger}(\lambda V\ \Phi)^{\text{`}\dagger}$$
$$\Downarrow(\lambda T\ \Phi)^{\text{`}\prime\ddagger} \Leftrightarrow_3 \lambda T \Downarrow\Phi^{\text{`}\prime\ddagger} \Leftrightarrow_{ih} \lambda T\ \Phi^{\text{`}\dagger} \Leftrightarrow_4 (\lambda T\ \Phi)^{\text{`}\dagger}$$

(1) If V' is the n-th variable of type ${}^{\uparrow\ddagger}a$, $\downarrow^{\ddagger}(\lambda V\ \Phi)^{\text{`}\prime\ddagger}$ equals $\downarrow^{\ddagger}(\lambda V'\ ([V'/V]\Phi)^{\text{`}\prime\ddagger})$ (by the definition of $'$), which equals $\lambda U \downarrow^{\ddagger}([\uparrow U/V'][V'/V]\Phi)^{\text{`}\prime\ddagger})$ (by the definition of $\downarrow$ and $\lambda$-conversion) and this equals $\lambda U \downarrow^{\ddagger}\Phi^{\text{`}\prime\ddagger}$ (by the definition of ‘; V is assumed to be numbered odd). (2) Similarly, $\downarrow^{\dagger}(\lambda V\ \Phi)^{\text{`}\dagger}$ equals $\lambda U\downarrow^{\dagger}([\uparrow U/V]\Phi)^{\text{`}\dagger}$ (definition of $\downarrow$ and λ-conversion) which equals $\lambda U \downarrow^{\dagger}(\Phi)^{\text{`}\dagger}$ (by the definition of ‘; $V$ and $U$ are assumed to be numbered odd). (3) Furthermore, if $T$ and $T'$ are the $n$-th variables of type ${}^{\uparrow\dagger}a$ and ${}^{\uparrow\ddagger}a$, respectively, $\Downarrow(\lambda T\ \Phi)^{\text{`}\prime\ddagger}$ equals $\Downarrow\lambda T'\ ([T'/T]\Phi)^{\text{`}\prime\ddagger}$ (definition of $'$) which equals $\lambda T \Downarrow([\Uparrow T/T'][T'/T]\Phi)^{\text{`}\prime\ddagger}$ (definition of $\Downarrow$ and λ-conversion) which equals $\lambda T \Downarrow\Phi^{\text{`}\prime\ddagger}$ (definition of ‘; $T$ is assumed to be numbered even). (4) Finally, $\lambda T\ \Phi^{\text{`}\dagger}$ equals $(\lambda T\ \Phi)^{\text{`}\dagger}$. ($T$ is assumed to be numbered even.)

Induction for the sentential operators:

$$\begin{array}{ll}
\downarrow^{\ddagger}(\alpha \doteqdot \beta)^{\text{`}\prime\ddagger} \Leftrightarrow_{df,rd} & (\downarrow^{\ddagger}(\alpha)^{\text{`}\prime\ddagger} = \downarrow^{\ddagger}(\alpha)^{\text{`}\prime\ddagger}) \Leftrightarrow_{ih} \\
& (\downarrow^{\dagger}(\alpha)^{\text{`}\dagger} = \downarrow^{\dagger}(\alpha)^{\text{`}\dagger}) \Leftrightarrow_{df,rd} \downarrow^{\dagger}(\alpha \doteqdot \beta)^{\text{`}\dagger} \\
\Downarrow(\alpha \doteqdot \beta)^{\text{`}\prime\ddagger} \Leftrightarrow_{df,rd} & \Downarrow\uparrow^{\ddagger}((\downarrow\alpha)^{\text{`}\prime\ddagger} = (\downarrow\beta))^{\text{`}\prime\ddagger} \Leftrightarrow_{2.16} \\
& \uparrow^{\dagger}((\downarrow\alpha)^{\text{`}\prime\ddagger} = (\downarrow\beta))^{\text{`}\prime\ddagger} \Leftrightarrow_{ih} \\
& \uparrow^{\dagger}((\downarrow\alpha)^{\text{`}\dagger} = (\downarrow\beta))^{\text{`}\dagger} \Leftrightarrow_{df,rd} (\alpha \doteqdot \beta)^{\text{`}\dagger}
\end{array}$$

$$\begin{array}{ll}
\downarrow^{\ddagger}(\sim\Phi)^{\text{`}\prime\ddagger} \Leftrightarrow_{df,rd} & \neg\downarrow^{\ddagger}(\Phi)^{\text{`}\prime\ddagger} \Leftrightarrow_{ih} \neg\downarrow^{\dagger}(\Phi)^{\text{`}\dagger} \Leftrightarrow_{df,rd} \downarrow^{\dagger}(\sim\Phi)^{\text{`}\dagger} \\
\Downarrow(\sim\Phi)^{\text{`}\prime\ddagger} \Leftrightarrow_{df,rd} & {}^{\wedge}\lambda q\ (\neg{}^{\vee}\Phi^{\text{`}\prime\ddagger}({}^{\wedge}\lambda p\ {}^{\vee}p) \wedge {}^{\vee}q) \Leftrightarrow_{df,rd} \\
& {}^{\wedge}\lambda q\ (\neg{}^{\vee}\Downarrow\Phi^{\text{`}\prime\ddagger}({}^{\wedge}\mathtt{true}) \wedge {}^{\vee}q) \Leftrightarrow_{ih} \\
& {}^{\wedge}\lambda q\ (\neg{}^{\vee}\Phi^{\text{`}\dagger}({}^{\wedge}\mathtt{true}) \wedge {}^{\vee}q) \Leftrightarrow_{df,rd} (\sim\Phi)^{\text{`}\dagger}
\end{array}$$

$$\begin{array}{ll}
\downarrow^{\ddagger}(\mathcal{E}d\Phi)^{\text{`}\prime\ddagger} \Leftrightarrow_{df,rd} & \exists d\downarrow^{\ddagger}(\Phi)^{\text{`}\prime\ddagger} \Leftrightarrow_{ih} \exists d\downarrow^{\dagger}(\Phi)^{\text{`}\dagger} \Leftrightarrow_{df,rd} \downarrow^{\dagger}(\mathcal{E}d\Phi)^{\text{`}\dagger} \\
\Downarrow(\mathcal{E}d\Phi)^{\text{`}\prime\ddagger} \Leftrightarrow_{df,rd} & {}^{\wedge}\lambda q\ \exists d({}^{\vee}\Phi^{\text{`}\prime\ddagger}({}^{\wedge}\lambda p\ {}^{\vee}p \wedge {}^{\vee}q)) \Leftrightarrow_{df,rd} \\
& {}^{\wedge}\lambda q\ \exists d({}^{\vee}\Downarrow\Phi^{\text{`}\prime\ddagger}(q)) \Leftrightarrow_{ih} {}^{\wedge}\lambda q\ \exists d({}^{\vee}\Phi^{\text{`}\dagger}(q)) \Leftrightarrow_{df,rd} (\mathcal{E}d\Phi)^{\text{`}\dagger}
\end{array}$$

$$\begin{array}{ll}
\downarrow^{\ddagger}(\Phi\ ;\Psi)^{\text{`}\prime\ddagger} \Leftrightarrow_{df,rd} & {}^{\vee}\Phi^{\text{`}\prime\ddagger}({}^{\wedge}\lambda p\ {}^{\vee}p \wedge {}^{\vee}\Psi^{\text{`}\prime\ddagger}({}^{\wedge}\lambda p\ {}^{\vee}p)) \Leftrightarrow_{df,rd} \\
& {}^{\vee}\Downarrow\Phi^{\text{`}\prime\ddagger}({}^{\wedge}\ {}^{\vee}\Downarrow\Psi^{\text{`}\prime\ddagger}({}^{\wedge}\mathtt{true})) \Leftrightarrow_{ih} \\
& {}^{\vee}\Phi^{\text{`}\dagger}({}^{\wedge}\ {}^{\vee}\Psi^{\text{`}\dagger}({}^{\wedge}\mathtt{true})) \Leftrightarrow_{df,rd} \downarrow^{\dagger}(\Phi\ ;\Psi)^{\text{`}\dagger} \\
\Downarrow(\Phi\ ;\Psi)^{\text{`}\prime\ddagger} \Leftrightarrow_{df,rd} & {}^{\wedge}\lambda q\ {}^{\vee}\Phi^{\text{`}\prime\ddagger}({}^{\wedge}\lambda p\ {}^{\vee}p \wedge {}^{\vee}\Psi^{\text{`}\prime\ddagger}({}^{\wedge}\lambda p\ {}^{\vee}p \wedge {}^{\vee}q)) \Leftrightarrow_{df,rd} \\
& {}^{\wedge}\lambda q\ {}^{\vee}\Downarrow\Phi^{\text{`}\prime\ddagger}({}^{\wedge}\ {}^{\vee}\Downarrow\Psi^{\text{`}\prime\ddagger}(q)) \Leftrightarrow_{ih} \\
& {}^{\wedge}\lambda q\ {}^{\vee}\Phi^{\text{`}\dagger}({}^{\wedge}\ {}^{\vee}\Psi^{\text{`}\dagger}(q)) \Leftrightarrow_{df,rd} (\Phi\ ;\Psi)^{\text{`}\dagger}
\end{array}$$

Fact 2.2 and 2.4 are proved as follows now. Since there are no free variables in Φ, $\downarrow^{\ddagger}\Phi'^{\ddagger}$ is $\downarrow^{\ddagger}\Phi^{\text{`}\prime\ddagger}$, which by the above proof equals $\downarrow^{\dagger}\Phi^{\text{`}\dagger}$ which is $\downarrow^{\dagger}\Phi^{\dagger}$; for the same reason $\Downarrow\Phi'^{\ddagger}$ is $\Downarrow\Phi^{\text{`}\prime\ddagger}$, which by the above proof equals $\Phi^{\text{`}\dagger}$ which is $\Phi^{\dagger}$.

Chapter 3

Flexible and dynamic interpretation

In the preceding two chapters we have seen some sample systems which give a compositional treatment of (some of) the dynamics of natural language interpretation. The dynamics is accounted for by translating the sentences of natural language into sentences of a formal language (that of *DPL*, *DFL*, or *DFL(2)*) which has a genuinely dynamic interpretation. The crucial notions in these systems are those of dynamic conjunction, dynamic existential quantification, and, in *DFL(2)*, dynamic negation.

An interesting question that comes to mind now is whether it is really necessary to employ a dynamic formalism. Wouldn't it be possible to obtain the same results statically? This chapter aims to show that this is indeed possible within a static classical system if we add a certain flexibility in interpretation.[1] As we will see in this chapter, the adoption of a version of Hendriks' [1988, 1992] system of type change on top of a relatively simple basic fragment of natural language, enables us to account for the phenomena dealt with in the preceding two chapters, and for much more.

A flexible approach to the phenomena involved seems a natural choice for two reasons. In the first place, the *DMG(2)* type of sentence meanings turns out to be a *derived* type of sentence meanings in several kinds of flexible categorial grammars, and the *DMG(2)* lift $\uparrow$ of formulas corresponds to a rule of lifting or value raising which is generally available in such grammars. So, *DMG(2)* and flexible systems of interpretation already have something in common. In the second place, the dynamics of *DMG* and *DMG(2)* has to do, apparently, with the (extended) scope of quantifiers and sentential operators. Since we are dealing with scope and shifting scope configurations, it seems natural to try to apply a flexible account of quantifier scope to the extended dynamic phenomena discussed in the preceding chapter.

1. Cf., Zeevat [1989, 1991] for an alternative static and compositional reformulation of *DRT*.

In this chapter a flexible approach is adopted in order to account for the dynamics of interpretation we have considered sofar. It may be expedient to indicate two 'limitations' of this enterprise first. Structures in discourse will be analyzed in terms of the scope of quantifiers and (sentential) operators. There is no intention to treat other quantifiers than the classical four, or to account for different ways to organize discourse than by means of the classical connectives.

Furthermore, it will become clear presently that a flexible approach like the one adopted here greatly overgenerates. Already in Hendriks' original system, which deals with scope phenomena at the sentential level, we find a vast landscape of possible readings, not all of which are realized in natural languages, and the numbers of readings greatly increase when the system is applied to scope phenomena at the level of discourse. I will not have much to say about (the controlling of) the numbers of readings. The main aim is simply to see to what extent dynamic aspects of meaning can be accounted for statically.

We will proceed as follows. Section 1 presents a (restricted) version of Hendriks' flexible Montague grammar (*FMG*) which gives a treatment of quantifier scope phenomena within sentences. In section 2, the class of scope bearing elements is extended with sentential connectives and the system of type change is applied at the level of discourse. Some basic *DMG(2)* notions turn out to be derivable notions and a first set of examples dealt with in *DMG(2)* (and *DMG*) is accounted for flexibly. Section 3 addresses the issue of the extended scope of downward monotonic expressions. Here I introduce a generalized notion of the dual of chapter 2 in the system of type shifts. The incorporation of this dual in the type changing system allows us to derive the other *DFL(2)* notions and to treat the remaining examples dealt with in *DMG(2)*. Section 4 finally discusses some remaining issues.

1 Flexible Montague grammar

Hendriks [1988, 1992] presents a flexible Montague grammar, *FMG*, in which the lexical expressions of a fragment of natural language are associated with the simplest types their meaning allows for.[2] The complications brought along by phenomena of quantifier scope, which led Montague to the association of natural language expressions with interpretations in more involved types, are handled in Hendriks' system by means of the adoption of a flexible category-to-type assignment together with a system of type shifting rules. The type changing system is presented as an alternative to and an improvement of other approaches to quantifier scope phenomena (such as Cooper's storage mechanism). Here, I will not discuss the relative merits of

2. Hendriks presents the first systematic elaboration of the adoption of semantic type flexibility which has been argued for in, among others, Partee and Rooth [1983], van Benthem [1984] and Groenendijk and Stokhof [1984, Ch. VI].

different approaches to quantifier scope phenomena, but I present Hendriks' system right away (see Hendriks [1988, 1992] for extensive discussion). I first discuss (a part of) the basic fragment and then show how the system of type shifts fits in.

The basic FMG fragment

In *FMG*, expressions of the fragment of natural language belong to the basic categories S (the category of sentences), CN (the category of common nouns), NP (the category of noun phrases), or to derived categories of the form A/B and $B\backslash A$. Derived categories which are employed in the fragment are those of intransitive verb phrases IV ($= NP\backslash S$), of determiners DET ($= NP/CN$), and of transitive verb phrases TV ($= IV/NP$). Each category is assigned a unique basic type by the function f which is defined as follows:

Definition 1.1 (Basic category to type assignment)

- $f(S) = t$
  $f(NP) = e$
  $f(CN) = \langle\langle s,e\rangle,t\rangle$
- $f(A/B) = f(B\backslash A) = \langle\langle s,f(B)\rangle,f(A)\rangle$

As appears from the definition of the basic category to type assignment, the basic types which are assigned to syntactic categories are essentially simpler than in traditional Montague grammar. The basic type of NP's is that of individuals, instead of (generalized) quantifiers over individual concepts, and the basic type of TV's is that of relations between individual concepts instead of relations between individual concepts and quantifiers over individual concepts.

The lexical expressions in the fragment are associated with basic translations. A basic translation of an expression of a certain category is of the basic type assigned to that category, or of a type derivable from the basic type by the type shifts defined below. For reasons of readibility, I will write $\hat{a}$ for the type $\langle s,a\rangle$. In the following definition x and y are variables of type $\hat{e}$, and P and Q of type $\hat{\langle}\hat{e},t\rangle$, `john` and `mary` are constants of type e, `man` and `walk` are constants of type $\langle e,t\rangle$, and `love` is a constant of type $\langle e,\langle e,t\rangle\rangle$[3]:

Definition 1.2 (Basic translations)

- man_{CN} $\rightsquigarrow \lambda x\ \mathtt{man}(^{\vee}x)$
- $walk_{IV}$ $\rightsquigarrow \lambda x\ \mathtt{walk}(^{\vee}x)$
- $love_{TV}$ $\rightsquigarrow \lambda y\lambda x\ \mathtt{love}(^{\vee}y)(^{\vee}x)$
- a_{DET} $\rightsquigarrow \lambda P\lambda Q\ \exists x(^{\vee}P(^{\wedge}x) \wedge {}^{\vee}Q(^{\wedge}x))$

3. In Hendriks' translation of common nouns like *man* also the PTQ meaning postulate is encoded that such common nouns only hold of constant individual concepts. However, as in *DMG(2)*, intensional types and expressions have nothing to do with ordinary intensionality, and, hence, matters concerning individual concepts and the like can be disregarded.

- $every_{DET} \rightsquigarrow \lambda P \lambda Q\ \forall x({}^{\vee}P({}^{\wedge}x) \rightarrow {}^{\vee}Q({}^{\wedge}x))$
- $no_{DET} \rightsquigarrow \lambda P \lambda Q\ \neg\exists x({}^{\vee}P({}^{\wedge}x) \wedge {}^{\vee}Q({}^{\wedge}x))$
- $Mary_{NP} \rightsquigarrow \texttt{mary}$
- $John_{NP} \rightsquigarrow \texttt{john}$

Natural language expressions are assigned basic translations which are as simple as can be. So, on its basic translation, the noun phrase *Mary* denotes an individual, viz., the denotation of `mary`, and the transitive verb *love* denotes a relation which holds of two individual concepts iff their extensions stand in the relation denoted by the constant `love`.

In some simple cases, the types of the basic translations of a functor expression and an argument expression 'fit'. For instance, the transitive verb *loves* has a basic translation which can be applied to the intensions of the basic translations of the noun phrases *Mary* and *John*. The resulting sentence *John loves Mary* has a basic translation which can be reduced as follows:

$$(\lambda y \lambda x\ \texttt{love}({}^{\vee}y)({}^{\vee}x))({}^{\wedge}\texttt{mary})({}^{\wedge}\texttt{john}) \Leftrightarrow \texttt{love}(\texttt{mary})(\texttt{john})$$

However, the determiners *a*, *every* and *no* have a translation which is not of the basic type associated with the category *DET*. Recall that this category is defined as *NP/CN* and that the basic type of *NP*'s is the type e.[4] However, the combination of these determiners with a common noun does not have a translation of the type e of individuals, but of the type $\langle\hat{}\langle\hat{}e, t\rangle, t\rangle$ of quantifiers over individual concepts. For this reason, a noun phrase which consists of such a determiner and a common noun cannot immediately combine with intransitive verb phrases like the proper names *Mary* and *John* do. Here Hendriks' system of type flexibility comes in. The translations of functor expressions can be made applicable to quantifying argument expressions by means of a system of type shifts (which, at the same time, determine the scope of the quantifying expressions).

Type sets

In addition to their basic translations, expressions of the *FMG* fragment are assigned translations derived from the basic translations by general type shifting rules. Hendriks uses three such rules, the rules of *value raising*, *argument raising* and *argument lowering*. The rule of value raising lifts expressions of a type a into expressions of the type of quantifiers over the intensions of objects of type a. The rule of argument raising lifts functional expressions which can be applied to expressions of a type

4. As a matter of fact, Hendriks associates the category *DET* with a basic type which *is* the type of the translations of these determiners. In other words, in Hendriks' system the basic type of the category of determiners is *not* the type of functions from the intensions of the basic type of common nouns to objects of the basic type of noun phrases. However, in both Hendriks' and in the above presentation of *FMG*, the combination of a determiner with a common noun makes up a noun phrase which has a basic translation of a type which is not the basic type of *NP*'s.

a into expressions which can be applied to expressions of the type of quantifiers over objects of type a. For the purposes of this chapter we may disregard argument lowering.

Hendriks' rules allow us to assign expressions of the fragment derived translations of different types. For this reason, the expressions of a certain category are associated not just with a unique basic type, but with a set of derivable types. Before we define the type *sets* associated with the categories of the fragment, it is useful to introduce some notation conventions:

Notation convention 1

- If $\vec{a}$ is a sequence of types $a_1 \dots a_n$, then $\langle \vec{a}, b \rangle = \langle a_1, \dots \langle a_n, b \rangle \dots \rangle$
- If $\vec{a}$ is a sequence of types $a_1 \dots a_n$, $\vec{x}$ a sequence of variables $x_1 \dots x_n$ with types $a_1 \dots a_n$, respectively, and ϕ an expression of type b, then $\lambda \vec{x} \phi = \lambda x_1 \dots \lambda x_n \phi$ (of type $\langle \vec{a}, b \rangle$)
- If $\vec{a}$ is a sequence of types $a_1 \dots a_n$, $\vec{x}$ a sequence of variables $x_1 \dots x_n$ with types $a_1 \dots a_n$, respectively, and ϕ an expression of type $\langle \vec{a}, b \rangle$, then $\phi(\vec{x}) = \phi(x_1) \dots (x_n)$ (of type b)

Each syntactic category C is associated with a set of types $T(C)$, the type set of C. This set contains the basic type assigned to C, and types derived from the basic type as in the following definition:

Definition 1.3 (Type set)
The type set $T(C)$ of category C is the smallest set such that:

1. $f(C) \in T(C)$
2. if $\langle \vec{a}, b \rangle \in T(C)$, then $\langle \vec{a}, \langle \hat{}\langle \hat{}b, t \rangle, t \rangle \rangle \in T(C)$
3. if $\langle \vec{a}, \langle b, \langle \vec{c}, t \rangle \rangle \rangle \in T(C)$, then $\langle \vec{a}, \langle \hat{}\langle \hat{}\langle b, t \rangle, t \rangle, \langle \vec{c}, t \rangle \rangle \rangle \in T(C)$

where $\vec{a}$ and $\vec{c}$ are arbitrary sequences of types, and b is an arbitrary type

According to the first clause the basic type associated with a category is in its type set. The second clause in this definition is related to the rule of value raising which will be defined presently. By this clause, the type set of the category of noun phrases also contains the type of quantifiers over individual concepts. The third clause in the definition corresponds to the rule of argument raising. By this clause, the category of transitive verb phrases is associated with a type set which also includes the type of expressions which can be successively applied to two expressions which denote quantifiers over individual concepts.

The FMG type shifts

Besides their basic translations, the lexical expressions of the *FMG* fragment are assigned translations derived from the basic translations by means of the rules of value raising and argument raising. Value raising is defined as follows (in this definition $n \geq 1$, derivability is indicated by the arrow $\Rightarrow$):

Definition 1.4 (n-th Value raising)
If $\vec{a}$ is a sequence of types $a_1, \ldots, a_{n-1}$ and ϕ is an expression of type $\langle \vec{a}, b \rangle$, then

- $\phi \Rightarrow \lambda \vec{x} \lambda Y\ {}^{\vee}Y({}^{\wedge}\phi(\vec{x}))$ ($\vec{x}$ and Y not free in ϕ)

where $\vec{x}$ is a sequence of variables of types $a_1, \ldots, a_{n-1}$ and Y is a variable of type $\hat{\langle} \hat{b}, t \rangle$, all variables distinct[5]

If we raise the n-th value of ϕ, an expression results the application of which to $x_1, \ldots, x_{n-1}$ denotes the set of properties of the object denoted by the intension of the application of ϕ to $x_1, \ldots, x_{n-1}$. For instance, if we raise the first value of the translation of the noun phrase *Mary*, we get the translation $\lambda P\ {}^{\vee}P({}^{\wedge}\texttt{mary})$, which denotes the set of properties of the individual concept of Mary. Notice that this is the translation of the proper name *Mary* in *MG*. In what follows I will write $[n\text{VR}](\phi)$ to indicate the expression that results from raising the n-th value of ϕ.

Argument raising is defined as follows:

Definition 1.5 (n-th Argument raising)
If b is a type, $\vec{a}$ and $\vec{c}$ are sequences of types $a_1, \ldots, a_{n-1}$ and $c_1, \ldots, c_m$ and ϕ is an expression of type $\langle \vec{a}, \langle b, \langle \vec{c}, t \rangle \rangle \rangle$, then

- $\phi \Rightarrow \lambda \vec{x} \lambda Y \lambda \vec{z}\ {}^{\vee}Y({}^{\wedge}\lambda y\ \phi(\vec{x})(y)(\vec{z}))$ ($\vec{x}$, Y, $\vec{z}$ and y not free in ϕ)

where $\vec{x}$ and $\vec{z}$ are sequences of variables of types $a_1, \ldots, a_{n-1}$ and $c_1, \ldots, c_m$, and y and Y are variables of type b and $\hat{\langle} \hat{\langle} b, t \rangle, t \rangle$, all variables distinct

The result of raising the n-th argument of an expression ϕ will be indicated by $[n\text{AR}](\phi)$. If the n-th argument of a function ϕ is of type b, then the n-th argument of the function $[n\text{AR}](\phi)$ is of the type of the intensions of quantifiers over objects of the type b. So, if the first argument of the translation $\lambda y \lambda x\ \texttt{love}({}^{\vee}y)({}^{\vee}x)$ of the transitive verb *love* is raised, the resulting translation denotes the relation between intensions of quantifiers over individual concepts and individual concepts $\lambda T \lambda x\ {}^{\vee}T({}^{\wedge}\lambda y\ \texttt{love}({}^{\vee}y)({}^{\vee}x))$. Notice that this is the translation of the transitive verb *love* in *MG*.

The FMG translation sets

Now that we have defined the basic translations of lexical expressions and the rules to construct derived translations, we can define the *sets* of translations which are assigned to *FMG* expressions. The translation set of a lexical expression consists of its basic translation and translations derived from the basic translation by means of value raising and argument raising:

---

5. Like Hendriks [1992], we might be more specific by requiring $\vec{x}$ to be the sequence of variables $x_1, \ldots, x_{n-1}$ where x_i is the i-th variable of type a_i. In what follows, I will pass over this subtlety.

Definition 1.6 (Translation set)
The translation set α'' of a lexical expression α is the smallest set such that:

- if $\alpha \rightsquigarrow \beta$, then $\beta \in \alpha''$
- if $\beta \in \alpha''$ and $\beta \Rightarrow \gamma$, then $\gamma \in \alpha''$

FMG finally has an adjusted rule of functional application. (For now I disregard Hendriks' rules for constructing conjunctions and disjunctions.) Since *FMG* has a bidirectional grammar with two kinds of functor categories $A\backslash B$ and B/A, it has two rules of application, right application and left application. Furthermore, since expressions are assigned *sets* of translations, the application of an expression of a functor category to an expression of the appropriate argument category also has a set of translations, each element of which consists of the application of a translation of the functor expression to the intension of a translation of the argument expression. The translation sets of such constructions are defined as follows:

Definition 1.7 (Functional application)
The translation set $(\beta_{B/A}\ \alpha_A)''_B$ of a compound expression $(\beta_{B/A}\ \alpha_A)_B$ is the smallest set such that:

- if $\alpha' \in \alpha''$, $\beta' \in \beta''$, α' has type a, and β' type $\langle \hat{a}, b\rangle$ then $\beta'(^\wedge\alpha') \in (\beta_{B/A}\ \alpha_A)''_B$

(Similarly for $(\alpha_A\ \beta_{A\backslash B})_B$.)

An example

As was already said, *FMG*'s type shifting rules can be used to make mismatching basic translations 'fit', and they serve to account for the different scope configurations that may obtain between the quantifying noun phrases in a sentence. Consider the following example:

(1) Every man loves a woman

This sentence has two readings, according to the respective scope of the two noun phrases *every man* and *a woman*. On one of these readings *every man* has widest scope, and on this reading the sentence is true iff for every man there is a woman which he loves, possibly a different woman for each man. On the other reading *a woman* has widest scope, and on that reading the sentence is true iff there is a woman such that every man loves her. In *FMG* both readings are accounted for.

The noun phrases *every man* and *a woman* have basic translations which result from applying the basic translations of *every* and *a* to the intensions of the basic translations of *man* and *woman*, respectively. By λ-conversion and $^{\vee\wedge}$-elimination these translations can be reduced to the following expressions:

$$\lambda Q\ \forall x(\mathtt{man}(x) \rightarrow {}^\vee Q(^\wedge x))$$
$$\lambda Q\ \exists y(\mathtt{woman}(y) \wedge {}^\vee Q(^\wedge y))$$

Both expressions are of type $\langle\hat{}\langle\hat{}e, t\rangle, t\rangle$, which is not the type of the arguments of the basic translation of the functor expression *love*, which is type $\hat{}e$. By means of [1AR] and [2AR], the translation of the transitive verb *love* can be made applicable to the translations of these two noun phrases. The two instances of [AR] can be applied in two orders, and the order in which they are applied determines which of the two noun phrases gets widest scope. If the first argument of the translation of *love* is raised first, and next the second argument, then the second argument, viz, the subject noun phrase *every man*, gets widest scope. If first the second argument is raised, and next the first argument, then the first argument, i.e., the object noun phrase *a woman*, gets widest scope.

So, the following two translations are in the translation set of *love*:

(i) $[2\text{AR}]([1\text{AR}](\lambda y \lambda x\ \texttt{love}({}^{\vee}y)({}^{\vee}x))) \Leftrightarrow$
$\lambda V \lambda U\ {}^{\vee}U({}^{\wedge}\lambda x\ {}^{\vee}V({}^{\wedge}\lambda y\ \texttt{love}({}^{\vee}y)({}^{\vee}x)))$

(ii) $[1\text{AR}]([2\text{AR}](\lambda y \lambda x\ \texttt{love}({}^{\vee}y)({}^{\vee}x))) \Leftrightarrow$
$\lambda V \lambda U\ {}^{\vee}V({}^{\wedge}\lambda y\ {}^{\vee}U({}^{\wedge}\lambda x\ \texttt{love}^{\vee}(y)({}^{\vee}x)))$

If translation (i) is applied to the intension of the translation of *a woman* and to the intension of the translation of *every man*, then we get the wide scope *every man* translation of example 1, which can be reduced to the following formula:

$$\forall x(\texttt{man}(x) \rightarrow \exists y(\texttt{woman}(y) \wedge \texttt{love}(y)(x)))$$

If we use translation (ii) of the verb *love*, then we get the wide scope *a woman* translation, which can be reduced to the following formula:

$$\exists y(\texttt{woman}(y) \wedge \forall x(\texttt{man}(x) \rightarrow \texttt{love}(y)(x)))$$

On deriving readings

Hendriks' system of type change is very powerful. It allows one to generate all the scope configurations between the quantifiers in a sentence that respect its application structure. I conclude this section with an informal characterization of the way in which specific scope configurations can be derived on the basis of the (syntactic) application structure of an expression and the basic translations of its lexical constituents. (The appendix to this chapter gives a more formal characterization of the 'lifting' of quantifiers through application structures. See Hendriks [1992] for the semantic properties of the type changing system.)

We may think of an application structure as a structure that consists of a lexical expression of a functor category together with n argument expressions ($n \geq 0$). The arguments can be lexical expressions, too, but they may also be application structures themselves (for example, embedded sentences, or compound relative noun phrases). In the fragment of flexible Montague grammar, a functor may be applied to an argument expression which has a basic translation of a type that does not match with the type of the relevant argument place of the translation of the functor. For instance, the i-th argument place of the translation of a functor may be of type $\hat{a}$, whereas the i-th argument has a translation of the type of quantifiers over objects of type $\hat{a}$. In that case, I will say that the functor applies to a quantifying argument in the i-th place.

In case a functor expression applies to a quantifying argument in the i-th place, the type mismatch is resolved after applying [iAR] to the translation of the functor. In that case the functor lands in the scope of the quantifying argument, together with all of its arguments that do fit. If the functor has other quantifying arguments, other instances of [AR] have to apply to the translation of the functor. A subsequent application of [jAR] makes that the scope of a quantifying j-th argument comprises the functor, of which the i-th argument is already raised, together with its 'fitting' arguments (among which is the quantifying i-th argument).

All basic type mismatches which are generated by the fragment above can be resolved by means of argument raising only, and as long as the quantifying expressions involved are arguments of the same functor, [AR] takes care of their scope configurations, too. In order to extend the scope of quantifying arguments beyond the application structure of which these arguments are immediate constituents, value raising must be used. By first raising the value of a functor and after that of one or more of its arguments, the quantifying arguments are lifted to a higher level in an application structure. If the value of the functor is raised, then the functor with its arguments constitutes a quantifying argument of a functor one level higher, and [AR] has to apply to that functor in order to make it fit again. By applying the same procedure to the functor of ... the functor of a quantifying argument, the scope of the argument can be made to comprise larger and larger bits of the application structure in which it occurs.

Generally, in order to account for any conceivable scope configuration between the quantifying expressions in a given application structure, we need zero or more applications of [nVR] to the translations of functors which are applied to precisely $n-1$ arguments, and applications of [AR] to resolve basic type mismatches or to resolve type mismatches which originate from applications of [VR].

2 Flexibility in discourse

Three adjustments must be made in order to turn *FMG* into a rudimentary flexible discourse grammar that covers the results of *DMG* and *DMG(2)*. In the first place, in order to be able to account for the phenomenon of dynamic binding, natural language expressions should be translated into *DIL*, instead of *IL* (cf., chapter 1).[6] In what follows this adjustment will simply be taken for granted, and the quantifiers $\exists$ and

6. As we have seen in chapter 1, *DIL* only provides for the prerequisites for a dynamic interpretation of discourse markers. The system itself is not dynamic in an intuitive sense.

$\forall$ which occur in the translations of natural language expressions will be taken to quantify over the values of discourse markers from now on. Pronouns are translated as discourse markers and some suitable system of indexing is assumed.

The second adjustment consists in an extension of the class of (flexible) scope bearing expressions. Not only quantifying noun phrases, but also sentential operators turn out to be expressions with varying scope. Formally, this will be handled by introducing these operators categorematically, as expressions of a syntactic functor category, and by extending the type changing system with a rule that serves to establish the varying scope of these operators.[7] This adjustment will be carried out in this section.

The third adjustment has to do with the lifting of downward monotonic expressions. This is the subject of section 3. In that section it will be argued that an extension of the scope of such expressions requires an alternative interpretation of the type shifts involved in the lifting. The required interpretation is based on a generalization of the notion of a dual from chapter 2.

Extension of the basic FMG fragment

In order to extend a flexible sentence grammar to a flexible discourse grammar, sentential connectives and modifiers are introduced categorematically. The operators are associated with the following basic translations (p and q are variables of type $\hat{}t$, P and Q are variables of type $\hat{}\langle\hat{}e, t\rangle$, and x is a variable of type $\hat{}e$):

Definition 2.1 (Basic translation of connectives and modifiers)

- $not_{S/S}$ $\rightsquigarrow \lambda p\ \neg{}^{\vee}p$
- $\cdot_{S\backslash(S/S)}$ $\rightsquigarrow \lambda p \lambda q\ ({}^{\vee}p \wedge {}^{\vee}q)$
- $if_{(S/S)/S}$ $\rightsquigarrow \lambda p \lambda q\ ({}^{\vee}p \rightarrow {}^{\vee}q)$
- $or_{S\backslash(S/S)}$ $\rightsquigarrow \lambda p \lambda q\ ({}^{\vee}p \vee {}^{\vee}q)$
- $who_{CN\backslash(CN/IV)}$ $\rightsquigarrow \lambda P \lambda Q \lambda x\ ({}^{\vee}P(x) \wedge {}^{\vee}Q(x))$

The expression *not* is a sentence modifier which involves the negation of the sentences to which it applies. The operator . is associated with an operation of sentence sequencing. Semantically it involves the conjunction of any two sentences to which it applies. With the lexical expression *if* conditional sentences *if S, S′* can be constructed, with *or* disjunctions, and the relative pronoun *who* can be used to construct adnominal relative clauses, like *man who loves Mary*.[8]

---

7. It must be pointed out here that Hendriks [1988, 1992] also deals with coordinations and with their interaction with the scope of quantifiers, but in a syncategorematic way.
8. It is a deliberate limitation, not a principled one, that we don't have relative clauses like *man whom Mary loves*, cf., footnote 17 in chapter 1.

As was already indicated, all these expressions will be treated as scope bearing operators.[9] More in particular, they will be considered to be operators which, like quantifying noun phrases, can be assigned a variety of scopes. For instance, the scope of the operator *if* will be subject to type changes which allow its nuclear scope to extend to sentences which follow a conditional sentence in a discourse. Thus, we will be able to account for sequences such as *If a customer comes in he is offered coffee. When he has finished his coffee, he is sent up to me*, in which, apparently, the second sentence is intended to stand in the scope of the *if*-clause.

For this reason the set of scope bearing expressions will be taken to consist of all expressions of a type which ends in the type $\langle \hat{}\langle \vec{b}, \langle \vec{d}, t\rangle\rangle \langle \vec{d}, t\rangle\rangle$, where $\vec{b}$ and $\vec{d}$ are arbitrary sequences of types. This set includes expressions of the type $\langle \hat{}\langle \hat{b}, t\rangle, t\rangle$, that is, expressions of the type of generalized quantifiers over objects of type $\hat{b}$, but it also includes modifying expressions of the type $\langle \hat{}\langle \vec{d}, t\rangle, \langle \vec{d}, t\rangle\rangle$.

The rule of division

The rules of [AR] and [VR] do not enable us to lift sentential modifiers and, thus, extend the scope of these expressions. Therefore, some adaptation of the type changing system is called for. I will introduce a rule, that of division, which will enable us to account for the varying scopes of both quantifying noun phrases and sentential modifiers and operators. The rule is a generalization of a type shifting rule which originates from Geach [1972]. The type shift, together with its interpretation, is defined as follows:

Definition 2.2 (Type set (2))
The type set $T(C)$ of category C is the smallest set such that:
1&2. as above
3. if $\langle \vec{a}, \langle \hat{}\langle \vec{b}, \langle \vec{d}, t\rangle\rangle, \langle \vec{d}, t\rangle\rangle\rangle \in T(C)$, then $\langle \vec{a}, \langle \hat{}\langle \vec{b}, \langle c, \langle \vec{d}, t\rangle\rangle\rangle, \langle c, \langle \vec{d}, t\rangle\rangle\rangle\rangle \in T(C)$
where $\vec{a}$, $\vec{b}$ and $\vec{d}$ are arbitrary sequences of types, c an arbitrary type

Definition 2.3 (Division of the n-th argument)
If $\vec{a}$, $\vec{b}$ and $\vec{d}$ are sequences of types $a_1, \ldots, a_{n-1}$, $b_1, \ldots, b_m$ and $d_1, \ldots, d_k$, c is a type, and ϕ is an expression of type $\langle \vec{a}, \langle \hat{}\langle \vec{b}, \langle \vec{d}, t\rangle\rangle, \langle \vec{d}, t\rangle\rangle\rangle$, then

- $\phi \Rightarrow \lambda\vec{x}\lambda Y\lambda z\ \phi(\vec{x})(^\wedge\lambda\vec{y}\ {}^\vee Y(\vec{y})(z))$ ($\vec{x}$, Y and z not free in ϕ)

where $\vec{x}$ and $\vec{y}$ are sequences of variables of types $a_1, \ldots, a_{n-1}$ and $b_1, \ldots, b_m$, and z and Y are variables of type c and $\hat{}\langle \vec{b}, \langle c, \langle \vec{d}, t\rangle\rangle\rangle$, all variables distinct

Let ϕ be a scope bearing expression of type $\langle \hat{}\langle \vec{b}, \langle \vec{d}, t\rangle\rangle, \langle \vec{d}, t\rangle\rangle$, which can be applied to a function of type $\hat{}\langle \vec{b}, \langle \vec{d}, t\rangle\rangle$. By means of division, ϕ can be made applicable to a function of type $\hat{}\langle \vec{b}, \langle c, \langle \vec{d}, t\rangle\rangle\rangle$ which has an additional argument slot of type

9. Such a conception of these operators fits in with the general scheme of quantification presented in Partee [1991], cf., also Roberts [1992].

c. The divided function 'inherits' this argument slot. So, by means of division, the scope of an expression can be extended with additional arguments. Semantically, the application of the divided expression to a function with such an additional argument slot involves the (intensional) composition of both functions. The division of the n-th argument of an expression ϕ into the type c will be indicated as $[n\mathrm{GD}_c](\phi)$.

Let us consider two examples. Let ϕ be the translation $\lambda Q\ \forall d_i(\mathtt{man}(d_i) \rightarrow {}^{\vee}Q({}^{\wedge}d_i))$ of the quantifying noun phrase *every man*. This translation can be applied to the intension of the basic translation $\lambda x\ \mathtt{walk}({}^{\vee}x)$ of the intransitive verb phrase *walks*. However, ϕ can be made to apply to the intension of the basic translation $\lambda y \lambda x\ \mathtt{hate}({}^{\vee}y)({}^{\vee}x)$ of the transitive verb phrase *hates* as well. Since the latter has an additional argument of type $\hat{}e$, ϕ has to be divided into the type $\hat{}e$, the result of which can be reduced to the expression $\lambda S \lambda x\ \forall d_i(\mathtt{man}(d_i) \rightarrow {}^{\vee}S({}^{\wedge}d_i)(x))$, where S is a variable of type $\hat{}\langle\hat{}e, \langle\hat{}e, t\rangle\rangle$. The application of this expression to the intension of the basic translation of the transitive verb *hate* reduces to the expression $\lambda x\ \forall d_i(\mathtt{man}(d_i) \rightarrow \mathtt{hate}(d_i)({}^{\vee}x))$, which denotes the set of individual concepts the extensions of which hate every man. Notice that the same result can be obtained by raising the first argument of the translation of the transitive verb, and applying that translation to ϕ. (In fact, as we will see below, the adoption of the rule of division allows us to do without [AR].)

The second example illustrates a central application of division. The same expression ϕ can be made applicable to the intension of the raised translation $\lambda x \lambda R\ {}^{\vee}R({}^{\wedge}\mathtt{walk}({}^{\vee}x))$ of *walks*, where R has the type $\hat{}\langle\hat{}t, t\rangle$. To this end ϕ has to be divided into the type $\hat{}\langle\hat{}t, t\rangle$ and the result of this can be reduced to the expression $\lambda P \lambda R\, \forall d_i(\mathtt{man}(d_i) \rightarrow {}^{\vee}P({}^{\wedge}d_i)(R))$, with P a variable of type $\hat{}\langle\hat{}e, \langle\hat{}\langle\hat{}t, t\rangle, t\rangle\rangle$. The application of this expression to the intension of the raised translation of *walk* can be reduced to the expression $\lambda R\ \forall d_i(\mathtt{man}(d_i) \rightarrow {}^{\vee}R({}^{\wedge}\mathtt{walk}(d_i)))$. Notice that this expression equals the *DFL(2)* translation of the sentence *Every man walks*.

Before we turn to the flexible counterparts of some *DMG* and *DMG(2)* results, it is useful to introduce two more notation conventions and to comment upon the relation between [AR] and [GD].

The expression that results from dividing a scope bearing expression is always a scope bearing expression. So, if the n-th argument of a function ϕ can be divided, it can be divided several times. Such iterated division is abbreviated as follows:

Notation convention 2
If ϕ is an expression of type $\langle\vec{a}, \langle\hat{}\langle\vec{b}, \langle\vec{d}, t\rangle\rangle, \langle\vec{d}, t\rangle\rangle\rangle$, $\vec{c}$ a sequence of types $c_1, \ldots, c_m$:

- $[n\mathrm{GD}_{\vec{c}}](\phi) = [n\mathrm{GD}_{c_1}](\ldots [n\mathrm{GD}_{c_m}](\phi) \ldots)$

The division of ϕ into $\vec{c}$ yields an expression of the type $\langle\vec{a}, \langle\hat{}\langle\vec{b}, \langle\vec{c}, \langle\vec{d}, t\rangle\rangle\rangle, \langle\vec{c}, \langle\vec{d}, t\rangle\rangle\rangle\rangle$ which can be reduced to the expression $\lambda\vec{x}\lambda Y\lambda\vec{z}\ \phi(\vec{x})({}^{\wedge}\lambda\vec{y}\ {}^{\vee}Y(\vec{y})(\vec{z}))$, where $\vec{z}$ is a sequence of variables of types $c_1, \ldots, c_m$ and Y is a variable of type $\hat{}\langle\vec{b}, \langle\vec{c}, \langle\vec{d}, t\rangle\rangle\rangle$.

Also, if the value of some expression is raised, the result is a scope bearing expression. The iterated division of an expression of which the value is raised is abbreviated as follows:

Notation convention 3
If ϕ is an expression of type $\langle\vec{a},b\rangle$ and $\vec{c}$ a sequence of types $c_1,\ldots,c_m$ then:

- $[n\mathrm{VR}_{\langle\vec{c},t\rangle}](\phi) = [n\mathrm{GD}_{\vec{c}}]([n\mathrm{VR}](\phi))$

If the n-th value of such an expression ϕ is raised, and the value is subsequently divided into the sequence of types $\vec{c}$, the resulting expression $[n\mathrm{VR}_{\langle\vec{c},t\rangle}](\phi)$ is of type $\langle\vec{a},\langle\hat{}\langle\hat{}b,\langle\vec{c},t\rangle\rangle,\langle\vec{c},t\rangle\rangle\rangle$, and can be reduced to the expression $\lambda\vec{x}\lambda Y\lambda\vec{z}\ {}^{\vee}Y({}^{\wedge}\phi(\vec{x}))(\vec{z})$.

As was already noticed above, the rule of generalized division effectively overlaps with the rule of argument raising. In fact, with just division and value raising, i.e., without argument raising, we can derive all scope configurations which may obtain between the quantifiers in a piece of text. The following fact shows how applications of [AR] can be replaced by applications of [VR] and [GD]:

Fact 2.1 ([nAR]-elimination)
If b is a type, $\vec{a}$ and $\vec{c}$ are sequences of types $a_1,\ldots,a_{n-1}$ and $c_1,\ldots,c_m$ and ϕ is an expression of type $\langle\vec{a},\langle b,\langle\vec{c},t\rangle\rangle\rangle$, then

- $([n\mathrm{AR}](\phi))(\vec{x})(T) \Leftrightarrow ([n\mathrm{VR}_{\langle\vec{c},t\rangle}](\phi))(\vec{x})({}^{\wedge}[1\mathrm{GD}_{\vec{c}}]({}^{\vee}T))$

where $\vec{x}$ is a sequence of variables of types $a_1,\ldots,a_{n-1}$ and T is a variable of type $\hat{}\langle\hat{}\langle b,t\rangle,t\rangle$, all variables distinct[10]

This fact shows that if the n-th argument of a functor expression is raised in order to accomodate a quantifying argument expression, the same result can be obtained by raising the functor to a divided value and by dividing the argument expression. The possibility of replacing applications of [AR] by applications of [VR] and [GD] makes [AR] in fact superfluous. (See the appendix for a more general statement of the possibility of [AR]-elimination.) However, for the sake of simplicity we will keep on using the rule of [AR].

A flexible account of DMG(2) results (1)

Given the categorematic treatment of connectives and modifiers, and the extension of the set of flexible scope bearing expressions with expressions of these categories, we can obtain some of the crucial *DMG(2)* results. In the first place the *DMG(2)* lift $\uparrow$ corresponds to an instance of value raising (cf., chapter 2). If ϕ is of type t, then:

10. Proof: If $\vec{z}$ is a sequence of variables with types $\vec{c}$, and y is a variable of type b, then:

$$([n\mathrm{VR}_{\langle\vec{c},t\rangle}](\phi))(\vec{x})({}^{\wedge}[1\mathrm{GD}_{\vec{c}}]({}^{\vee}T)) \Leftrightarrow \lambda\vec{z}\ ([1\mathrm{GD}_{\vec{c}}]({}^{\vee}T))({}^{\wedge}\phi(\vec{x}))(\vec{z}) \Leftrightarrow$$
$$\lambda\vec{z}\ {}^{\vee}T({}^{\wedge}\lambda y\ \phi(\vec{x})(y)(\vec{z})) \Leftrightarrow ([n\mathrm{AR}](\phi))(\vec{x})(T)$$

Fact 2.2

- $\uparrow\phi \Leftrightarrow {}^\wedge\lambda R\ {}^\vee R({}^\wedge\phi) \Leftrightarrow {}^\wedge[1\mathrm{VR}](\phi)$ (R of type ${}^\frown\langle{}^\frown t, t\rangle$

Also the *DMG(2)* notions of dynamic conjunction and dynamic disjunction can be derived from the (static) basic translations of these connectives by means of argument raising and division (p is a variable of type ${}^\frown t$, R a variable of type ${}^\frown\langle{}^\frown t, t\rangle$ and Φ and Ψ are expressions of type ${}^\frown\langle{}^\frown\langle{}^\frown t, t\rangle, t\rangle$; A' indicates the basic translation of A):

Fact 2.3

- $[\Phi\ ;\Psi] \Leftrightarrow {}^\wedge\lambda R\ {}^\vee\Phi({}^\wedge\lambda p\ ({}^\vee p \wedge {}^\vee\Psi(R))) \Leftrightarrow {}^\wedge[1\mathrm{AR}]([2\mathrm{GD}_{{}^\frown\langle{}^\frown t,t\rangle}](.'))(\Phi)(\Psi)$
- $[\Phi$ **or** $\Psi] \Leftrightarrow {}^\wedge\lambda R\ {}^\vee\Phi({}^\wedge\lambda p\ ({}^\vee p \vee {}^\vee\Psi(R))) \Leftrightarrow {}^\wedge[1\mathrm{AR}]([2\mathrm{GD}_{{}^\frown\langle{}^\frown t,t\rangle}](or'))(\Phi)(\Psi)$

The applications of [GD] divide the connectives into the type of functions expressing continuation with subsequent discourse, cf., the second example given under definition 2.3. Thus, the second argument of these connectives may come to range over further discourse. The applications of [AR] enable the connectives to apply to a first argument of a raised type. Thus, quantifiers in such a first conjunct or disjunct may come to range over the second one, and, since these second arguments are divided, also over further discourse. Notice that in the fragment as it has been presented sofar the two sentential connectives . and *or* are the only fully upward monotonic sentential operators, that is, upward monotonic in all of their arguments. Type changes of downward monotonic expressions are the subject of the next section.[11]

The extended dynamic interpretations of some examples discussed in chapter 2 make up one of the readings of these examples in the flexible system. Consider, for instance, example 2 (example 4 in chapter 2):

(2) Every$_i$ player chooses a$_j$ pawn. He$_i$ puts it$_j$ on square one.

The transitive verb *chooses* in this example is flanked by two quantifying noun phrases. Therefore the first and second argument of its basic translation have to be raised. The scope of the noun phrases is extended by raising the value of the verb first. Since the verb has two arguments it is the third value that is raised. The following translation is used:

$$[2\mathrm{AR}]([1\mathrm{AR}]([3\mathrm{VR}](\lambda y\lambda x\ \mathtt{choose}({}^\vee y)({}^\vee x)))) \Leftrightarrow$$
$$\lambda U\lambda V\lambda R\ {}^\vee V({}^\wedge\lambda x\ {}^\vee U({}^\wedge\lambda y\ {}^\vee R({}^\wedge\mathtt{choose}({}^\vee y)({}^\vee x))))$$

---

11. The *DMG(2)* notions of dynamic existential quantification and of dynamic universal quantification can be derived if we introduce expressions like *there is an individual such that* and *for every individual it holds that*, of the category S/S, with type $\langle{}^\frown t, t\rangle$, and with basic translations $\lambda p\ \exists d({}^\vee p)$ and $\lambda p\ \forall d({}^\vee p)$, respectively. On the basis of these expressions the *DMG(2)* notions of quantification can be derived in the following way:

- $\mathcal{E}d\Phi \Leftrightarrow {}^\wedge\lambda R\ \exists d{}^\vee\Phi(R) \Leftrightarrow {}^\wedge[1\mathrm{GD}_{{}^\frown\langle{}^\frown t,t\rangle}](\lambda p\ \exists d({}^\vee p))(\Phi)$
- $\mathcal{A}d\Phi \Leftrightarrow {}^\wedge\lambda R\ \forall d{}^\vee\Phi(R) \Leftrightarrow {}^\wedge[1\mathrm{GD}_{{}^\frown\langle{}^\frown t,t\rangle}](\lambda p\ \forall d({}^\vee p))(\Phi)$

If we apply this translation of the verb to the intensions of the basic translations of a_j *pawn*, $\lambda Q\ \exists d_j(\texttt{pawn}(d_j) \wedge {}^\vee Q({}^\wedge d_j))$, and *every$_i$ player*, $\lambda Q\ \forall d_i(\texttt{player}(d_i) \rightarrow {}^\vee Q({}^\wedge d_i))$, we get a translation of the first sentence of example 2 which can be reduced to the following expression:

$$\lambda R\ \forall d_i(\texttt{player}(d_i) \rightarrow \exists d_j(\texttt{pawn}(d_j) \wedge {}^\vee R({}^\wedge\texttt{choose}(d_j)(d_i))))$$

The intension of this expression is equivalent with the *DMG(2)* translation of the first sentence of example 2.

The basic translation of the second sentence is $\texttt{put\_on}(1)(d_j)(d_i)$. The two sentences, with these translations, can be combined by the sequencing operator if the first argument of its basic translation is raised (P is a variable of type $\check{}\langle\check{}\langle\check{}t, t\rangle, t\rangle$):

$$[1\text{AR}](\lambda p\lambda q\ ({}^\vee p \wedge {}^\vee q))) \Leftrightarrow \lambda P\lambda q\ {}^\vee P({}^\wedge\lambda p\ ({}^\vee p \wedge {}^\vee q))$$

The application of this translation to the intensions of the translations of the first and second sentence of example 2 can be reduced to the following formula:

$$\forall d_i(\texttt{player}(d_i) \rightarrow \exists d_j(\texttt{pawn}(d_j) \wedge \texttt{choose}(d_j)(d_i) \wedge \texttt{put\_on}(1)(d_j)(d_i)))$$

Under this translation example 2 has the same *truth conditions* as in extended *DMG* and in *DMG(2)*. The example also has a translation which has the same *dynamic potential* as in *DMG(2)*. This translation is obtained if we use a raised translation of the second sentence of example 2, i.e., $\lambda R\ {}^\vee R({}^\wedge\texttt{put\_on}(1)(d_j)(d_i))$, and the *DMG(2)* translation of the sequencing operator which is obtained by means of [1AR] and a division of the second argument into the type $\check{}\langle\check{}t, t\rangle$.

In a completely similar fashion we can derive the dynamic interpretations of simple donkey examples like 3 (example 1 in chapter 1) and internally dynamic disjunctions like 4 (example 9 in chapter 2):

(3) A$_i$ man walks in the park. He$_i$ whistles.
(4) Either there is a$_i$ bathroom downstairs or it$_i$ is upstairs.

The *DMG(2)* interpretations of these examples are obtained by means of the translations: [1AR]([2VR](*walk_in_the_park′*)), [2VR](*whistle′*) and [1AR]([2GD$_{\check{}\langle\check{}t,t\rangle}$](*.′*)), and [1AR]([2VR](*is_downstairs′*)), [2VR](*is_upstairs′*) and [1AR]([2GD$_{\check{}\langle\check{}t,t\rangle}$](*or′*)). Employing the basic translations of the other lexical expressions in the examples 3 and 4, and by means of (intensional) functional application, we get the following (reduced) translations of the examples:

$$\lambda R\ \exists d_i(\texttt{man}(d_i) \wedge \texttt{walk}(d_i) \wedge {}^\vee R({}^\wedge\texttt{whistle}(dd_i)))$$
$$\lambda R\ \exists d_i(\texttt{bathroom}(d_i) \wedge (\texttt{down}(d_i) \vee {}^\vee R({}^\wedge\texttt{up}(d_i))))$$

These expressions correspond to the *DMG(2)* interpretations of the examples.

Extended coverage

We see that some basic *DMG* and *DMG(2)* results can be obtained on the basis of a relatively simple, static fragment of natural language which is extended with a system of type change. As is to be expected, the completely general nature of the type shifting rules allows us to derive much more, and this extended coverage proves to be useful for an account of certain complicated pieces of discourse.

Consider the following example:

(5) Every$_i$ customer is offered coffee, that is, if he$_i$ looks wealthy. He$_i$ is sent up to me as quickly as possible. If he$_i$ doesn't look wealthy, he$_i$ can wait.

Example 5 displays complicated semantic dependencies. On a natural reading of this example the noun phrase *every$_i$ customer* binds pronouns in all three sentences. Furthermore, the verb phrase *is offered coffee* and the second sentence *He$_i$ is sent up* ... should be read, apparently, as being qualified by the phrase *if he$_i$ looks wealthy*. Finally, the third sentence *If he$_i$ doesn't* ... should not be read as being so qualified, but it does stay within the scope of the quantifying noun phrase *every$_i$ customer*. The semantic structure of example 5 can be indicated in the following way:

Every customer
    (if he looks wealthy
        (is offered coffee and
        he is sent up) and
    if he doesn't look wealthy he can wait)

However, the syntactically driven application structure is completely different:

((if he looks wealthy$_{S\backslash S}$
    is offered coffee$_{NP\backslash S}$
        every customer$_{NP}$) $\cdot_{S\backslash(S/S)}$
he is sent up$_S$) $\cdot_{S\backslash(S/S)}$
if he doesn't look wealthy he can wait$_S$

Nevertheless, by means of the type changing system the semantic structure of example 5 can be accounted for. The expressions *customer*, *is offered coffee*, *look wealthy*, *is sent up to me as quickly as possible*, and *can wait* can be associated with (simplified) basic translations $\lambda x\ \mathtt{cu}({}^\vee x)$, $\lambda x\ \mathtt{co}({}^\vee x)$, $\lambda x\ \mathtt{we}({}^\vee x)$, $\lambda x\ \mathtt{su}({}^\vee x)$, $\lambda x\ \mathtt{wa}({}^\vee x)$, all of type $\langle\hat{e}, t\rangle$. Furthermore, we can use the following derived translations:

(a) *is offered coffee*: [1AR]([2VR]([2VR]($\lambda x\ \mathtt{co}({}^\vee x)$)))
(b) *if he$_i$ looks wealthy*: [1AR]([2VR]([1GD$_{\hat{}\langle\hat{t},t\rangle}$] ($\lambda q\ (\mathtt{we}(d_i) \to {}^\vee q)$)))
(c) $\cdot_1$: [1AR]([3VR]([1AR]($\lambda p \lambda q\ ({}^\vee p \wedge {}^\vee q)$)))
(d) $\cdot_2$: [1AR]($\lambda p \lambda q\ ({}^\vee p \wedge {}^\vee q)$)

In (c) I have specified the translation of the sequencing operation which is used to combine the first sentence of example 5 with the second sentence, and in (d) the translation which is used to combine the result of that with the third sentence.

The application of [1AR] in (a) makes the verb phrase *is offered coffee* applicable to the quantifying noun phrase *every$_i$ customer*. Since this application of [1AR] is preceded by an application of [2VR], the scope of this quantifier is raised to a higher level in the application structure. To this (second) application of [2VR] in (a) corresponds an application of [1AR] in (b) which, in its turn, is preceded by an application of [2VR]. Thus, the scope of the quantifying noun phrase is raised higher up again. To the application of [2VR] in (b) corresponds the last (outermost) application of [1AR] in (c), which is also preceded by an application of [VR]. Hence, the scope of the quantifiying noun phrase turns out to range over the whole of example 5, and it is finally fixed by the application of [1AR] in (d), which is not preceded by an application of [VR].

By means of the three remaining type changes, the first (innermost) application of [2VR] in (a), the application of [1GD] in (b) and the first application of [1AR] in (c), the scope of the relativizing phrase *if he*$_i$ *looks wealthy* is made to range over the conjunction with the second sentence. The first two type shifts lift this operator, and its scope is fixed by the application of [1AR] to the sequencing operation which combines the first two sentences of example 5.

Using the indicated translations, we arrive at the following, reduced, translation of example 5, which gives its intended interpretation (see the appendix for a detailed derivation of this example):

$$\forall d_i(\mathtt{cu}(d_i) \rightarrow ((\mathtt{we}(d_i) \rightarrow (\mathtt{co}(d_i) \wedge \mathtt{su}(d_i))) \wedge \\ (\neg\mathtt{we}(d_i) \rightarrow \mathtt{wa}(d_i))))$$

On overgeneration

The last example may have given an idea of the strenghth of the flexible system. Clearly, this strength has its drawbacks. It has already been indicated that Hendriks' flexible Montague grammar derives all theoretically possible scope configurations that may hold between the quantifiers in a sentence. In the present system the set of scope bearing expressions is extended, and, furthermore, since the grammar is extended to a (rudimentary) discourse grammar, all scope configurations can be derived which hold between the scope bearing expressions in a complete text. Clearly, we have a source of overgeneration here which has to be restricted.

For instance, if a scope bearing expression occurs in a sentence in the middle of a piece of text, it is easily shown that the expression can be made to range over the entire text. This is not as it should be. Although it may be the case that scope bearing expressions in a sentence S may range over certain sentences which *follow* S in a discourse, it seems unreasonable to extend their scope over sentences which *precede* S. So, a first restriction that comes up quite naturally would be to exclude such backwards binding.[12]

---

12. Of course, there exist cases of backwards binding or kataphora, cf., for instance, van Deemter [1991, Ch. II] and the examples 12 and 13 in section 4. However, as van Deemter argues, there do not seem to be examples of kataphoric relationships in which expressions in a sentence S bind pronouns in a sentence which precedes S in a discourse. Kataphora appears to be an intrasentential phenomenon.

Overgeneralizations like the ones indicated can be eliminated by imposing restrictions upon the system of type change. For instance, we can restrict the translation set associated with the sequencing operation by means of which the sentences in a discourse are combined. If we exclude the use of [1GD], [2AR] and [2VR] in obtaining derived translations of this operation, then the first argument (or value) of the translation associated with this operation can only be raised, and the second argument can only be divided. As a consequence, scope bearing expressions in a sentence S which constitutes the first argument of this operator may come to range over expressions in a sentence which follows S and which constitutes the second argument of the operator, but not the other way around.[13] So, by adopting such a restriction we have a notion of sequencing which is flexible *and* directional, i.e., asymmetric, since it allows expressions in a sentence to range over subsequent sentences, but not over preceding ones.

3 Flexibility and monotonicity

In chapter 2 we have come across several examples in which downward monotonic expressions appear to bind pronouns across sentence boundaries. However, it would not do to account for these phenomena by simply lifting these expressions with the system of type change at hand. If the type shifting rules operate as they stand, we again face the problems we encountered with the notion of dynamic negation as complementation in *DMG* and *DMG(2)*. As the discussion in section 1 of chapter 2 may have shown it would give plainly wrong results if the scope of a downward monotonic expression like *no man* is extended to other sentences in a discourse.

In this section I will show that the lifting of downward monotonic expressions can be associated with an appropriate interpretation if we employ a suitable generalization of the notion of the dual defined in chapter 2. We will see that, with such a generalized notion of the dual, not only the examples discussed in chapter 2 can be dealt with, but also some other examples which are even more puzzling.

---

13. Proof: Suppose $and(^\wedge P)(^\wedge Q)$ is a translation of a sequence $S.S'$, where and is a translation associated with the sequencing operation, and P and Q are translations of S and S', respectively. I will show that under the restrictions formulated above no subexpression of Q ranges over any subexpression of P. This proposition, which is abbreviated as (A), is proved by induction on the number of type shifts used to derive and from the basic translation of the sequencing operation. In the basic case, the basic translation of . is used. Then $and(^\wedge P)(^\wedge Q)$ comes down to the conjunction of P and Q and clearly (A) holds. Next, I show that if (induction hypothesis) (A) holds of $and(^\wedge P)(^\wedge Q)$ for arbitrary P and Q of appropriate types, then (A) holds of $[n\text{TC}](and)(^\wedge R)(^\wedge S)$, for any allowed type change $[n\text{TC}]$ and for arbitrary R and S of appropriate types. Three cases can be distinguished. If $n \geq 3$, and if (A) holds of $and(^\wedge P)(^\wedge Q)$, then (A) holds of $[(n-2)\text{TC}](and(^\wedge P)(^\wedge Q)) \Leftrightarrow [n\text{TC}](and)(^\wedge P)(^\wedge Q)$. If $n = 2$, then $[n\text{TC}]$ must be [2GD], and, using the induction hypothesis, (A) holds of $\lambda z\ and(^\wedge P)(^\wedge \lambda \vec{y}\ Q(\vec{y})(z)) \Leftrightarrow [2\text{GD}](and)(^\wedge P)(^\wedge Q)$. Finally, if $n = 1$, then $[n\text{TC}]$ is not allowed to be [1GD], and it cannot be [1VR]. So, if $n = 1$, then $[n\text{TC}]$ is [1AR], and, using the induction hypothesis, (A) holds of $\lambda \vec{z}\ P(^\wedge \lambda p\ (and)(p)(^\wedge Q)(\vec{z})) \Leftrightarrow [n\text{TC}](and)(^\wedge P)(^\wedge Q)$.

Monotonicity and duality

Before we turn to the adapted interpretation of the lifting of downward monotonic expressions, let me first give a general definition of monotonicity:

Definition 3.1 (Monotonicity)
If ϕ is an expression of type $\langle\vec{a}, \langle\langle\vec{b}, t\rangle, \langle\vec{c}, t\rangle\rangle\rangle$, and $\vec{a}$, $\vec{b}$ and $\vec{c}$ are sequences of types $a_1, \ldots, a_{n-1}$, $b_1, \ldots b_k$ and $c_1, \ldots c_m$, respectively, then

- ϕ is upward monotonic in its n-th argument iff
  $\forall M, s, g\ \forall\vec{x}, p, q, \vec{z}\ (p \subseteq q \rightarrow ([\![\phi]\!]_{M,s,g}(\vec{x})(p)(\vec{z}) \rightarrow [\![\phi]\!]_{M,s,g}(\vec{x})(q)(\vec{z})))$
- ϕ is downward monotonic in its n-th argument iff
  $\forall M, s, g\ \forall\vec{x}, p, q, \vec{z}\ (p \supseteq q \rightarrow ([\![\phi]\!]_{M,s,g}(\vec{x})(p)(\vec{z}) \rightarrow [\![\phi]\!]_{M,s,g}(\vec{x})(q)(\vec{z})))$

where $\vec{x}$ and $\vec{z}$ are sequences of meta-variables which range over objects of types $a_1, \ldots, a_{n-1}$ and $c_1, \ldots c_m$, respectively, and p and q are meta-variables which range over objects of type $\langle\vec{b}, t\rangle$

Inspecting the fragment from the sections 1 and 2 we find four expressions with downward monotonic basic translations. These expressions are *every*, which is downward monotonic in its first argument; *no*, which is downward monotonic in its first and in its second argument; and *not* and *if*, which are both downward monotonic in their first argument.

When downward monotonic expressions are involved in type shifts the monotonicity properties of these operators and of other expressions may be disturbed. For instance, if a downward monotonic argument of some expression is raised it becomes upward monotonic in that argument. Also, if a functor is upward monotonic in its second argument, and if its first argument is raised and applied to a downward monotonic argument expression, then the result is downward monotonic in the argument in which the original expression was upward monotonic in the first place.

Following the strategy adopted in chapter 2, the present goal is to find adapted interpretations of type shifts which preserve monotonicity properties. To this end I introduce a generalization of the notion of a dual which is employed in *DMG(2)*. This notion of a dual will be used in the interpretation of type shifts by means of which downward monotonic expressions are lifted.[14]

14. Some remarks are in order here. If an expression is subject to type change, the choice of using an ordinary or a dual type shift depends upon its *semantic* properties, viz., monotonicity properties. This aspect of the proposed system of type change raises the question, which has not been properly addressed as yet, whether the system of type shifts has to be conceived of as belonging to syntax proper, or to semantics.

On the one hand, type shifts are operations upon translations, i.e., syntactic (*(D)IL*) operations. On the other hand, the translations of natural language expressions merely serve to represent the interpretations of these expressions. In *PTQ* the language of translation is used as a dispensible device to formulate these interpretations. For these reasons, if one wants to give a rigourously compositional map from the syntactic algebra of analyzed natural language expressions to the

The generalized dual, an operation that can be applied to expressions of all types, is given by the following recursive definition:

Definition 3.2 (Generalized dual)

- $\phi^*_e = \phi$
  $\phi^*_t = \neg\phi$
- $\phi^*_{\langle a,b\rangle} = \lambda x_a(\phi(x)^*)^*$ (x not free in ϕ)
- $\phi^*_{\langle s,a\rangle} = {}^\wedge({}^\vee\phi)^*$

This notion of a dual is a proper generalization of the notion of the dual in *DMG(2)*, of the classical notion of negation and of the *DMG(2)* notion of dynamic negation. If ϕ is an expression of type t, $\phi^* = \neg\phi$ is the (classical) negation of ϕ. If an expression R is of type $\hat{}\langle\hat{t}, t\rangle$, then $R^* = {}^\wedge\lambda p\ ({}^\vee R(p^*))^* = {}^\wedge\lambda p\ \neg{}^\vee R({}^\wedge\neg{}^\vee p)$, which is the dual of R in *DMG(2)*. And if Φ is an expression of type $\hat{}\langle\hat{}\langle\hat{t}, t\rangle, t\rangle$, then $\Phi^* = {}^\wedge\lambda R\ ({}^\vee\Phi(R^*))^* = {}^\wedge\lambda R\ \neg{}^\vee\Phi(R^*)$, which is the *DMG(2)* dynamic negation of Φ.

The present notion of duality also comprises the classical duality of $\wedge$ and $\vee$ ($(\lambda p\lambda q\ (p \wedge q))^* \Leftrightarrow \lambda p\lambda q\ (p \vee q)$) and the notion of duality as it is used in the theory of generalized quantifiers (where the dual of a quantifier $\mathcal{Q}$ is defined as the set of sets of individuals the complements of which are not in $\mathcal{Q}$).

It is easily seen that the double generalized dual of any expression ϕ equals ϕ:

Fact 3.1 (Double dual)

- $\phi^{**} \Leftrightarrow \phi$

Dual type changes

If type changing rules are applied to downward monotonic expressions they are associated with the following interpretations (types and variables are as in the definitions of [nVR], [nAR], and [nGD], respectively):

---

algebra of their possible interpretations there are two ways to proceed. One might take the system of type shifts to belong to the syntax of natural language, and associate each application of a type changing rule with a syntactic operation. Alternatively, one might take it to belong to the semantics, and associate with each analyzed expression a set of interpretations.

In the present context we do not need to take a definite stand in this issue. It suffices to note three things. First, the first option is intuitively less satisfactory, since it complicates the syntax for what are semantic reasons intuitively speaking, but it probably turns out to be more tractable formally speaking. Second, if semantic properties of expressions play a role in determining what are the expression's derived interpretations, then the type shifts are preferably conceived of as semantic. But, third, also syntactic considerations may determine what are approppriate type shifts. For these reasons, the ultimate choice betweeen the classification of the system of type shifts as belonging either to syntax or to semantics will have to be guided by more far-reaching considerations than can be dealt with here.

Definition 3.3 (Dual type changes)
[n-th Value raising (dual)] Under the conditions in the definition of [nVR], if ϕ is downward monotonic in its n-th argument and y is a variable of type b, then

- $\phi \Rightarrow \lambda\vec{x}\lambda Y\ {}^{\vee}Y^*({}^{\wedge}\lambda y\ \phi(\vec{x})(y^*))$

[n-th Argument raising (dual)] Under the conditions in the definition of [nAR], if ϕ is downward monotonic in its n-th argument, then

- $\phi \Rightarrow \lambda\vec{x}\lambda Y\lambda\vec{z}\ {}^{\vee}Y^*({}^{\wedge}\lambda y\ \phi(\vec{x})(y^*)(\vec{z}))$

[Division of the n-th argument (dual)] Under the conditions in the definition of [nGD], if ϕ is downward monotonic in its n-th argument, then

- $\phi \Rightarrow \lambda\vec{x}\lambda Y\lambda z\ \phi(\vec{x})({}^{\wedge}\lambda\vec{y}\ {}^{\vee}Y(\vec{y})(z^*))$

The alternative type shifting rules, which will be indicated by [nDVR], [nDAR] and [nDGD], respectively, only differ from the original rules in the use of duals of arguments abstracted over. In the case of [nDVR] and [nDAR], both the dual of the original downward monotonic n-th argument of ϕ is taken and the dual of the n-th argument of the raised expression. In the case of [nDGD], the dual of the 'inherited' arguments is taken. An iterated dual division into a sequence of types $\vec{c}$ will be indicated by [$n\text{DGD}_{\vec{c}}$].

Employing the definition of * the results of applying [nDVR] or [nDAR] can be spelled out a little further. For instance, if ϕ is downward monotonic in its n-th argument, then it is of some type $\langle\vec{a},\langle\langle\vec{b},t\rangle,\langle\vec{c},t\rangle\rangle\rangle$. The result of raising ϕ's n-th value by means of [nDVR] then can be reduced in the following way (variables are typed as in the definition above, and if $\vec{z}$ is a sequence of variables $z_i,\ldots,z_j$, then $\vec{z^*}$ is a sequence $z_i^*,\ldots,z_j^*$):

$$\lambda\vec{x}\lambda Y\ {}^{\vee}Y^*({}^{\wedge}\lambda y\ \phi(\vec{x})(y^*)) \Leftrightarrow \lambda\vec{x}\lambda Y\ \neg{}^{\vee}Y({}^{\wedge}\lambda y\lambda\vec{z}\ \neg\phi(\vec{x})(y)(\vec{z^*}))$$

Furthermore, if ϕ is of type $\langle\vec{a},\langle b,\langle\vec{c},t\rangle\rangle\rangle$, then the result of raising ϕ's n-th argument by means of [nDAR] can be reduced in the following way:

$$\lambda\vec{x}\lambda Y\lambda\vec{z}\ {}^{\vee}Y^*({}^{\wedge}\lambda y\ \phi(\vec{x})(y^*)(\vec{z})) \Leftrightarrow \lambda\vec{x}\lambda Y\lambda\vec{z}\ \neg{}^{\vee}Y({}^{\wedge}\lambda y\ \neg\phi(\vec{x})(y)(\vec{z}))$$

Here we see that the expression that results from applying [nDVR] or [nDAR] to ϕ is also downward monotonic in its n-th argument. Of course, also a raised expression [nDVR](ϕ) can be divided into a sequence of types $\vec{c}$ by means of [$n\text{DGD}_{\vec{c}}$]. The expression that results, $\lambda\vec{x}\lambda Y\lambda\vec{z}\ \neg{}^{\vee}Y({}^{\wedge}\lambda y\lambda\vec{z}\ \neg\phi(\vec{x})(y)(\vec{z^*}))(\vec{z^*})$, will be indicated by [$n\text{DVR}_{\langle\vec{c},t\rangle}$]($\phi$).

A flexible account of DMG(2) results (2)

Employing the dual type changes, two more *DMG(2)* notions, that of dynamic negation and that of dynamic implication, turn out to be derivable notions. In the following fact p is a variable of type ${}^{\wedge}t$, R a variable of type ${}^{\wedge}\langle{}^{\wedge}t,t\rangle$, and Φ and Ψ are expressions of type ${}^{\wedge}\langle{}^{\wedge}\langle{}^{\wedge}t,t\rangle,t\rangle$:

Fact 3.2

- $\sim\Phi \Leftrightarrow {}^\wedge\lambda R\ \neg{}^\vee\Phi(R^*) \Leftrightarrow {}^\wedge[\text{1DGD}_{\hat{}\langle\hat{}t,t\rangle}](not')(\Phi)$[15]
- $[\Phi \Rightarrow \Psi] \Leftrightarrow {}^\wedge\lambda R\ \neg{}^\vee\Phi({}^\wedge\lambda p\ ({}^\vee p \wedge \neg{}^\vee\Psi(R))) \Leftrightarrow {}^\wedge[\text{1DAR}]([\text{2GD}_{\hat{}\langle\hat{}t,t\rangle}](if'))(\Phi)(\Psi)$

So, with the present system of type change, in which we use the dual interpretation of type shifts if they are applied to downward monotonic expressions, the *DMG(2)* interpretations of dynamic sentential operators all are derivable from the static basic translations of these operators in the fragment above.

I will now illustrate the dual type changing rules by showing how they can be used to obtain the *DMG(2)* truth conditions of examples which are discussed in chapter 2. I first discuss example 6, which is example 1 in chapter 2:

(6) It is not the case that John owns no$_i$ car. It$_i$ is standing in front of the house.

In order to account for the anaphoric relationship between the noun phrase *no*$_i$ *car* and the pronoun *it*$_i$ the noun phrase is assigned scope over both sentences in this example. This can be achieved by raising the third value of the translation of the verb *own* first, and next adapt it to its quantifying argument expression by raising its first value and dividing this raised value.[16] The basic translation of the quantifying noun phrase is also divided, and, since it is downward monotonic, the dual interpretation of the rule of division is used. The following derived translations result (again, R is a variable of type $\hat{}\langle\hat{}t,t\rangle$, which is abbreviated as $\hat{}\tau$, $\mathcal{T}$ is a variable of type $\hat{}\langle\hat{}\langle\hat{}e,\langle\hat{}e,\langle\hat{}\tau,t\rangle\rangle\rangle,\langle\hat{}e,\langle\hat{}\tau,t\rangle\rangle\rangle$, and $\mathcal{Q}$ is a variable of type $\hat{}\langle\hat{}e,\langle\hat{}e,\langle\hat{}\tau,t\rangle\rangle\rangle$):

$$[\text{1VR}_{\langle\hat{}e,\langle\hat{}\tau,t\rangle\rangle}]([\text{3VR}](\lambda y\lambda x\ \texttt{own}({}^\vee y)({}^\vee x))) \Leftrightarrow$$
$$\lambda\mathcal{T}\lambda x\lambda R\ {}^\vee\mathcal{T}({}^\wedge\lambda y\lambda x\lambda R\ {}^\vee R({}^\wedge\texttt{own}({}^\vee y)({}^\vee x)))(x)(R)$$

$$[\text{1DGD}_{\hat{}e,\hat{}\tau}](\lambda Q\ \neg\exists d_i(\texttt{car}(d_i) \wedge {}^\vee Q({}^\wedge d_i))) \Leftrightarrow$$
$$\lambda\mathcal{Q}\lambda x\lambda R\ \neg\exists d_i(\texttt{car}(d_i) \wedge {}^\vee\mathcal{Q}({}^\wedge d_i)(x^*)(R^*))$$

Application of the first translation to the intension of the second translation gives a translation of the phrase *owns no car*, which can be reduced to the following expression:

$$\lambda x\lambda R\ \neg\exists d_i(\texttt{car}(d_i) \wedge {}^\vee R^*({}^\wedge\texttt{own}(d_i)({}^\vee x^*))) \Leftrightarrow$$
$$\lambda x\lambda R\ \neg\exists d_i(\texttt{car}(d_i) \wedge {}^\vee R^*({}^\wedge\texttt{own}(d_i)({}^\vee x)))$$

Application of this expression to the intension of the basic translation of *John* gives the following translation of the sentence *John owns no*$_i$ *car*:

$$\lambda R\ \neg\exists d_i(\texttt{car}(d_i) \wedge {}^\vee R^*({}^\wedge\texttt{own}(d_i)(\texttt{john})))$$

Notice that this translation is equivalent with the translation of the sentence *John owns no*$_i$ *car* in *DMG(2)*.

The basic translation $\lambda p\ \neg{}^\vee p$ of *it is not the case that* is divided in order to be applicable to the intension of the above translation of *John owns no car*. The

---

15. Notice that $[\text{1DGD}_{\hat{}\langle\hat{}t,t\rangle}](not') \Leftrightarrow [\text{1DAR}][\text{2VR}](not')$.

16. It is important *not* to use [1AR] to this end, cf., the argument against argument raising below.

dual interpretation of this type shift results in the translation $\lambda P\lambda R\ \neg{}^\vee P(R^*)$ and after application we get the following (reduced) translation of the first sentence of example 6:

$$\lambda R\ \neg\neg\exists d_i(\texttt{car}(d_i) \wedge {}^\vee R^{**}({}^\wedge\texttt{own}(d_i)(\texttt{john}))) \Leftrightarrow$$
$$\lambda R\ \exists d_i(\texttt{car}(d_i) \wedge {}^\vee R({}^\wedge\texttt{own}(d_i)(\texttt{john})))$$

In fact, this translation corresponds to the *DMG(2)* interpretation of the sentence *John owns a car*. If we raise the first argument of the sequencing operator, apply the result to the intension of the above translation of the first sentence of example 6 and apply the result of that to the intension of the translation $\texttt{before\_the\_house}(d)$ of the second sentence, we get a translation of the example which can be reduced to the following expression:

$$\exists d(\texttt{car}(d) \wedge \texttt{own}(d)(\texttt{john}) \wedge \texttt{before\_the\_house}(d))$$

This expression has the truth-conditions assigned to example 6 in *DMG(2)*.

The second example is 7 (example 7 in chapter 2, R and τ are as above, $\mathcal{T}$ is a variable of type $\hat{}\langle\hat{}\langle\hat{}e, \langle\hat{}\tau, t\rangle\rangle, \langle\hat{}\tau, t\rangle\rangle$ this time, $\mathcal{Q}$ is a variable of type $\hat{}\langle\hat{}e, \langle\hat{}\tau, t\rangle\rangle$ now):

(7) No$_i$ player leaves the room. He$_i$ stays where he$_i$ is.

The noun phrase *no player$_i$* is made capable of binding the pronoun *he$_i$* if we use the following type shifts:

$$[1\text{VR}_{\langle\hat{}\tau,t\rangle}]([2\text{VR}](\lambda x\ \texttt{leave}({}^\vee x))) \Leftrightarrow$$
$$\lambda\mathcal{T}\lambda R\ {}^\vee\mathcal{T}({}^\wedge\lambda x\lambda R\ {}^\vee R({}^\wedge\texttt{leave}({}^\vee x)))(R)$$
$$[1\text{DGD}_{\hat{}\tau}](\lambda Q\ \neg\exists d_i(\texttt{player}(d_i) \wedge {}^\vee Q({}^\wedge d_i))) \Leftrightarrow$$
$$\lambda\mathcal{Q}\lambda R\ \neg\exists d_i(\texttt{player}(d_i) \wedge {}^\vee\mathcal{Q}({}^\wedge d_i)(R^*))$$

By means of functional application we get the following reduced translation of the first sentence of example 7:

$$\lambda R\ \neg\exists d_i(\texttt{player}(d_i) \wedge {}^\vee R^*({}^\wedge\texttt{leave}(d_i))) \Leftrightarrow$$
$$\lambda R\ \forall d_i(\texttt{player}(d_i) \rightarrow {}^\vee R({}^\wedge\neg\texttt{leave}(d_i)))$$

Using the same translation of the sequencing operator as in the preceding example and the basic translation $\texttt{stay}(d_i)$ of *He$_i$ stays where he$_i$ is* we get the following (reduced) translation of example 7:

$$\forall d_i(\texttt{player}(d_i) \rightarrow (\neg\texttt{leave}(d_i) \wedge \texttt{stay}(d_i)))$$

The truth conditions are, again, the same as in *DMG(2)*.

The last example that I discuss here is example 8 (example 10 in chapter 2):

(8) If it is not the case that there is a$_i$ bathroom downstairs, then it$_i$ is upstairs.

In the translation set of the sentence *there is a$_i$ bathroom downstairs* we find the following (reduced) expression (with R again a variable of type $\hat{}\tau = \hat{}\langle\hat{}t, t\rangle$):

$$\lambda R\ \exists d_i(\texttt{bathroom}(d_i) \wedge {}^{\vee}R({}^{\wedge}\texttt{downstairs}(d_i))$$

The antecedent of example 8 consists of the negation of this sentence. Applying the dually divided negation operator to the intension of the above translation we get the following translation of the antecedent:

$$\lambda R\ \neg\exists d_i(\texttt{bathroom}(d_i) \wedge {}^{\vee}R^{*}({}^{\wedge}\texttt{downstairs}(d_i))) \Leftrightarrow$$
$$\lambda R\ \neg\exists d_i(\texttt{bathroom}(d_i) \wedge \neg{}^{\vee}R({}^{\wedge}\neg\texttt{downstairs}(d_i)))$$

The first argument of the basic translation of *if* is raised. Since this translation is downward monotonic in that argument, [1DAR] is used (P is a variable of type $\hat{}\langle\hat{}\tau, t\rangle = \hat{}\langle\hat{}\langle\hat{}t, t\rangle, t\rangle$):

$$[\text{1DAR}](\lambda p\lambda q\ ({}^{\vee}p \to {}^{\vee}q)) \Leftrightarrow$$
$$\lambda P\lambda q\ \neg{}^{\vee}P({}^{\wedge}\lambda p\ \neg({}^{\vee}p \to {}^{\vee}q))$$

The application of this expression to the intension of the translation of the antecedent of example 8 and to that of the consequent ($\texttt{upstairs}(d_i)$) yields the following (reduced) translation of the example:

$$\neg\neg\exists d_i(\texttt{bathroom}(d_i) \wedge \neg\neg(\neg\texttt{downstairs}(d_i) \to \texttt{upstairs}(d_i))) \Leftrightarrow$$
$$\exists d_i(\texttt{bathroom}(d_i) \wedge (\neg\texttt{downstairs}(d_i) \to \texttt{upstairs}(d_i)))$$

The other examples which are discussed in chapters 1 and 2 have translations which assign them the same truth conditions as in *DMG(2)*. The required type changes are indicated in the appendix to this chapter.

An argument against argument raising

It may be noticed that the present system, with [AR] and [DAR], still allows also unmodified extensions of the scope of downward monotonic expressions. For instance, for a sequence *No player leaves the room. He stays where he is* we can derive the reading that no player both leaves the room and stays where he is, by applying [3VR] and [2AR] to the translation of the verb *leave* and raising the first argument of the sequencing operator. Since the two translations which are raised are not downward monotonic in the addressed arguments, no *dual* interpretation of the involved type shifts has to be used. In other words, the scope of downward monotonic arguments can be extended with argument raising, and without type shifts operating on the arguments themselves. As has already been indicated, such extensions of the scope of downward monotonic expressions do not generate appropriate readings.

This observation may in fact constitute an argument against using [AR] (or [DAR], for that matter) in the present system of type change. The reason is that in a system in which applications of [AR] are replaced by applications of [VR] and [GD], the scope of quantifying expressions is determined by type shifts which operate both on the functor expressions and on the (quantifying) argument expressions. The pay off of this way of deriving scope configurations is that it can be made sensitive to the specific semantic properties of all the expressions in an application structure, not just to those of the functor.

The conclusion of this may be that we have to dispose of all applications of [AR]. However, we need not be that forbidding. If the dual type changes are used appropriately, then there will be no downward monotonic *sentence* translations. And for this reason it does no harm to allow argument raising in the translation of expressions of type S/S and $S\backslash S$. More in specific, we may keep on using [1AR] to get derived translations of the sequencing operation which do not allow monotonicity properties to get mixed up. For this reason we can keep to the restrictions on the type shifts of the sequencing operation which exclude the extensions of the scope of quantifying expressions over preceding discourse, cf., the end of section 2.

Furthermore, the fact remains that in all cases in which [AR] accomodates an (upward monotonic) function to an upward monotonic quantifying argument, i.e., those applications of [AR] that do not disturb monotonicity properties, the application of [AR] can be replaced by applications of [VR] and [GD]. Furthermore, similar applications of [nDAR] can be eliminated by means of [nDVR] and [1GD], as the following fact shows:

Fact 3.3 ([nDAR]-elimination)
Under the conditions stated in [nAR]-elimination:

- $([n\mathrm{DAR}](\phi))(\vec{x})(T) \Leftrightarrow ([n\mathrm{DVR}_{\langle\vec{c},t\rangle}](\phi))(\vec{x})(^\wedge[1\mathrm{GD}_{\vec{c}}](^\vee T))$[17]

For this reason, and for ease of exposition, we can keep on using applications of [AR] and [DAR] in the sequel, that is, if they accomodate functor expressions to upward monotonic quantifying argument expressions and, hence, are eliminable.

More involved structures

It has been shown that we can obtain *DMG(2)* results by means of a system of type change with an adapted interpretation of type shifts of downward monotonic expressions. In the adapted system of type change, the *DMG(2)* notions of the lift of expressions of type t, of dynamic negation, conjunction, disjunction and implication all are derived notions. Furthermore, as the treatment of example 5 has already shown, the type changing system enables a treatment of more intricate scope phenomena than the rigid dynamic Montague grammars do. I will now discuss two more examples. The interpretation of these examples crucially relies upon the duals of the type changing rules.

The first example is example 9:

---

17. Proof: If $\vec{z}$ is a sequence of variables with types $\vec{c}$, and y is a variable of type b, then:
$([n\mathrm{DVR}_{\langle\vec{c},t\rangle}](\phi))(\vec{x})(^\wedge[1\mathrm{GD}_{\vec{c}}](^\vee T)) \Leftrightarrow \lambda\vec{z}\ \neg([1\mathrm{GD}_{\vec{c}}](^\vee T))(^\wedge\lambda y\lambda\vec{z}\ \neg\phi(\vec{x})(y)(\vec{z}^{*}))(\vec{z}^{*}) \Leftrightarrow$
$\lambda\vec{z}\ \neg^\vee T(^\wedge\lambda y\ \neg\phi(\vec{x})(y)(\vec{z}^{**})) \Leftrightarrow ([n\mathrm{DAR}](\phi))(\vec{x})(T)$

(9) If a_i farmer owns a_j donkey, he_i stones it_j. If he_i leases it_j, he_i pleases it_j.

In this example the second conjunct *if he leases it, he pleases it* should *not* be read as dependent upon the antecedent of the first conjunct *a farmer owns a donkey*, but, nevertheless, the pronouns in the second conjunct are bound by the noun phrases in the antecedent. Such a reading can be accounted for by simply 'raising' the two noun phrases *a farmer* and *a donkey* and assigning them scope over the whole conjunction. Since these noun phrases are raised out of a downward monotonic context (the antecedent of the implication), we must use a dual type change. The following translations are used:

$$[2AR]([1AR]([3VR](own'))); \; [1DAR]([3VR](if')); \; [1AR](.')$$

If we employ these translations, we get the following (reduced) translation of example 9 (cf., the appendix for a detailed derivation):

$$\forall d_i(\texttt{farmer}(d_i) \rightarrow \forall d_j(\texttt{donkey}(d_j) \rightarrow ((\texttt{own}(d_j)(d_i) \rightarrow \texttt{stone}(d_j)(d_i)) \wedge (\texttt{lease}(d_j)(d_i) \rightarrow \texttt{please}(d_j)(d_i)))))$$

This translation expresses a proper interpretation of example 9.

The second, really involved, example is the following:

(10) No_i farmer beats a donkey, if he_i isn't insane. He_i will not yell at it either. If he_i is insane, then he_i might beat a donkey.

This example must be compared with example 5 which exhibits the same syntactic structure. The main difference with example 5 is that the quantifying noun phrase that has widest scope is not the upward monotonic noun phrase *every customer*, but the downward monotonic noun phrase *no farmer*. This downward monotonic noun phrase should not be lifted as such. But, as we will see, if we extend the scope of *no farmer* by means of a dual type shift, we get an appropriate interpretation of the example in which *no farmer beats a donkey* is read as *every farmer does not beat a donkey* with *every farmer* having wide scope.

Crucial are the following type shifts ($\lambda x \; \texttt{bad}(^{\vee}x)$ abbreviates a basic translation of *beats a donkey*, the anaphoric relationship between *a donkey* and *it* is left outside of consideration here; τ abbreviates the type $\langle\hat{}\langle\hat{}\langle\hat{}t, t\rangle, t\rangle, t\rangle$ now):

(11) beats a donkey: $[1VR_{\langle\hat{}\tau,t\rangle}]([2VR]([2VR](\lambda x \; \texttt{bad}(^{\vee}x))))$
no$_i$ farmer: $[1DGD_{\hat{}\tau}](\lambda Q \; \neg\exists d_i(\texttt{cu}(d_i) \wedge {}^{\vee}Q(^{\wedge}d_i)))$

The difference with example 5 is that instead of the application of [1AR] on the raised verb phrase, we find an application of [1VR] here, with a corresponding division of the quantifying subject noun phrase. Now, if we shift the types of the other expressions as in the treatment of example 5 above, we get the following (reduced) translation of example 10 (cf., again the appendix):

$$\forall d(\texttt{farmer}(d) \rightarrow ((\texttt{sane}(d) \rightarrow (\neg\texttt{bad}(d) \wedge \neg\texttt{yell\_at\_it}(d))) \wedge (\neg\texttt{sane}(d) \rightarrow \texttt{maybe\_bad}(d))))$$

This translation gives the right reading of the example.

The last two examples show that, although it would give awkward results to lift downward monotonic expressions in discourse, a proper use of the notion of a generalized dual yields adequate results.

4 Remaining issues

We have seen that the dynamics of interpretation, to the extent that it is accounted for in *DMG* and *DMG(2)*, can also be handled by starting from a simple static framework, and adding to that a system of type change, employing dual interpretations of type shifts in order to cope with the lifting of downward monotonic expressions. This result raises two questions. In the first place we may wonder which (different) readings the system *with* dual type shifts derives for sentences which are also dealt with in Hendriks' system of type change, which is a system *without* such dual rules. In the second place, the possibility of this flexible account of dynamic phenomena asks for a reassesment of the concept of dynamic interpretation. These two issues are briefly addressed in what follows.

Quantifier scope revisited

The dual interpretations of type shifts of downward monotonic expressions are motivated by intuitions concerning the extra-sentential binding properties of downward monotonic quantifiers. However, since it is assumed sofar that the dual interpretations of type shifts *always* have to be used when they are applied to downward monotonic expressions, lifts of downward monotonic expressions *within* sentences are also executed by means of dual type shifts. So, with respect to the issue of wide scope downward monotonic expressions within sentences the present system generates readings which differ from the readings derived in Hendriks' system of type change. I will now try to find support for any one of the two analyses. As we will see, the results are rather inconclusive.

First consider the following pair of examples, both exhibiting a kataphoric relationship:

(12) If he$_i$ is in danger, every$_i$ man prays to God.
(13) If he$_i$ isn't insane, no$_i$ man beats a donkey.

Both in a system with, and in one without dual type changes, sentence 12 has a translation under which *every$_i$ man* has wide scope. The sentence has the following (reduced) translation[18]:

18. This translation results from applying [2VR] to the translation $\lambda x\ \mathtt{pray\_to\_God}({}^{\vee}x)$ of the intransitive verb phrase *prays to God*, by raising the first argument of the resulting expression, and by applying [2AR] to the basic translation of *if* .

$\forall d_i(\texttt{man}(d_i) \rightarrow (\texttt{in\_danger}(d_i) \rightarrow \texttt{pray\_to\_God}(d_i)))$

However, it would give non-intuitive results if in the structurally similar example 13 the downward monotonic quantifying noun phrase *no*$_i$ *man* is assigned wide scope in the same way. In that case a translation would result which can be reduced to the following expression:

$\neg\exists d_i(\texttt{man}(d_i) \wedge (\neg\texttt{insane}(d_i) \rightarrow \texttt{bad}(d_i)))$

This formula is true iff it is both true that no man is insane and that no man beats a donkey. Clearly, this formula does not express a genuine reading of example 13. On my proposal the noun phrase *no*$_i$ *man* has to be subjected to a dual type shift, since this noun phrase is downward monotonic. Employing such a dual type shift, example 13 gets the following translation[19]:

$\forall d_i(\texttt{man}(d_i) \rightarrow (\neg\texttt{ins}(d_i) \rightarrow \neg\texttt{bad}(d_i)))$

On this reading sentence 13 is true iff it holds of every man that if the man is not insane, then he does not beat a donkey. This is the proper reading of the example.

The second example concerns a 'de re' belief with a downward monotonic noun phrase:

(14) John believes that no theory is sound.

If the quantifying noun phrase *no theory* is assigned wide scope without using a dual interpretation of the type changes involved, the following translation results[20]:

$\neg\exists d(\texttt{theory}(d) \wedge \texttt{believe}(^\wedge\texttt{sound}(d))(\texttt{john}))$

This formula is true iff there is no theory such that John believes it to be sound. I am not really sure, but I am inclined to judge this not to be a genuine reading of the example. With the dual type changes the following translation would be obtained:

$\forall d(\texttt{theory}(d) \rightarrow \texttt{believe}(^\wedge\neg\texttt{sound}(d))(\texttt{john}))$

This does constitute a genuine reading of the example. If John thinks of every theory in some domain of discussion "This one is unsound", the situation can be adequately described by means of sentence 14. Notice that such a situation does not license the conclusion that John believes (de dicto) that all theories are unsound.

These two examples seem to favour the dual interpretations of type shifts by means of which downward monotonic quantifiers are assigned wide scope within sentences. Notice that in these examples the quantifiers are raised out of a sentential clause,

19. Using the translation $[1\text{VR}_{\langle\hat{}\langle\hat{}t,t\rangle,t\rangle}]([2\text{VR}](\lambda x\ \texttt{bad}(^\vee x)))$ of the verb phrase *beats a donkey*, where $\lambda x\ \texttt{bad}(^\vee x)$ abbreviates its basic translation, the translation $[1\text{DGD}_{\hat{}\langle\hat{}t,t\rangle}](\lambda Q\ \neg\exists d_i(\texttt{man}(d_i) \wedge {}^\vee Q(^\wedge d_i)))$ of the noun phrase *no*$_i$ *man*, and the translation $[2\text{AR}](\lambda p \lambda q(^\vee p \rightarrow {}^\vee q))$ of *if*.

20. It must be noted that an account of belief ascriptions requires us to re-introduce *IL*'s ordinary intensionality. Clearly, beliefs cannot be represented adequately by means of states which are, or behave like, discourse marker assignments. In order for the present discussion to proceed smoothly, such an adaptation is simply assumed.

viz., out of the consequent of a conditional in example 13, and out of a sentential complement in example 14. So it remains to be seen how the dual type shifts behave with respect to the most simple examples of wide scope readings, the cases in which a downward monotonic object argument of a transitive verb gains scope over the verb's subject argument. As we will see presently, such examples are by and large inconclusive with respect to the choice between the dual and the non-dual type shifts.

First consider an example with an existentially quantifying subject argument:

(15) A Mac adorns no desk.

On the wide scope object reading of this example which is obtained without dual type shifts this sentence states that there is no desk such that a Mac adorns it. This is an acceptable reading of the example. However, if the object is assigned wide scope using the dual type shifts, we get a translation stating that for every desk there is a Mac that does not adorn it. Such a reading is out.

The second example has a universally quantified subject[21]:

(16) Every change is no improvement.

This time, the wide scope object reading obtained without dual type shifts is out. For on that reading the sentence states that there is no improvement which is equal to every change, and the sentence would turn out true on this reading if, for instance, there are two changes. However, employing the dual type shifts, the wide scope object reading states that every improvement is no change, and this reading is equivalent with the (acceptable) narrow scope object reading.

Finally, consider the following two examples:

(17) No man lost no game.
(18) Not every man lost no game.

On its non-dual wide scope object reading example 17 states that every game is such that a man lost it, which is also the dual wide scope object reading of example 18. The dual wide scope object reading of example 17 states that every game is such that every man lost it, which is also the non-dual wide scope object reading of example 18. For both sentences both derived readings are out.

The conclusion seems to be that the dual type shifts appear to behave well in an account of complicated discourse phenomena, that they give acceptable interpretations of the raising of downward monotonic expressions out of sentential contexts, but that they fail to do any better then the non-dual rules when we are concerned with establishing the scope of quantifiers within one sentential clause. It appears that sentences with downward monotonic object arguments do not have genuine wide scope object readings, except example 15 which has an existentially quantified

21. This example is due to Herman Hendriks [p.c.].

subject argument. In fact, for this sentence the wide scope object reading obtained by Hendriks' non-dual type shifts gives the right result.

Dynamic interpretation revisited

The system of type change presented in the sections 2 and 3 enables us to deal with scope and binding phenomena in a way which significantly differs from the way in which these phenomena are dealt with in *DMG* and *DMG(2)*. *DMG* and *DMG(2)* embrace a dynamic, directional notion of interpretation. In these sytems the dynamics of interpretation is taken to consist in the way in which (expressions in) sentences may change the context of interpretation of subsequent sentences and such a notion of dynamic interpretation has a strong intuitive appeal. In this chapter I have given an account of the same phenomena which starts from ordinary, static interpretations. Scope extensions in this account might also be called dynamic, but then they are dynamic in some non-directional sense. In principle, expressions in the flexible system may change the (context of) interpretation of preceding discourse as much as that of subsequent discourse.

Of course we are not forced to favour any one of the two approaches at the cost of the other, since we also have the possibility of combining the two approaches. Instead of adding a system of type change to a simple, static fragment like the one above, one might add it to a dynamic fragment like *DMG* (or some fragment employing update semantics, cf., Veltman [1990]). Thus, we might separate the treatment of the basic, incremental, dynamics of natural language interpretation from the treatment of the extended dynamics and of 'deviating' scope configurations and discourse structures. Actually, such a division of labour can be argued for.

The basic dynamics of interpretation which is dealt with in *DPL* and *DMG* is a pervasive phenomenon and it is not in need of any constraints. Indefinites that do not occur in certain subordinate positions are always accessible as antecedents for subsequent anaphoric reference. On the other hand, the extended dynamics of *DMG(2)* and the discourse structures treated in this chapter exhibit rather exceptional features of interpretation. In general, the scope of an *if*-clause does *not* extend to other sentences, in previous or in subsequent discourse, and, normally, negation is *not* dynamic. If we were to separate the treatment of the two kinds of phenomena, viz., the (pervasive) dynamics of indefinite noun phrases and the (exceptional) scope extensions of other expressions, we might be in a better position to formulate restrictions on scope extensions, or, rather, formulate the conditions under which type shifts are invoked, without this affecting the (unconstrained) dynamics of interpretation.

Clearly, the system of type change cannot be added to a dynamic fragment like *DMG* just like that. The system (in particular the definition of the type shifts and that of the dual) will have to be adapted to the type of sentence meanings and the specific sentential operators in such a fragment. However, there does not appear to be any real obstacle to combining the kind of flexibility presented in this chapter with the systems of dynamic interpretation mentioned.

5 Appendix

In the first section of this appendix I give a proof of the fact that by means of the rules of value raising and argument raising any quantifier can be lifted over any functor which has the quantifier in its syntactic scope. In the second section I show that the same results can be obtained using just the rules of value raising and division, i.e., without using argument raising. In the last section I show in more detail the deriviation of certain readings of examples which are discussed above.

Scope configurations in the FMG fragment

In this section I want to show that all scope configuration can be derived in Hendriks' system of type change. I will prove that the scope of any quantifying argument can be lifted over functors, the type of which does not end in type e, by means of applications of [VR] and [AR]. Some terminology must be developed first.

Definition 5.1 (Application structures)
The application structure AS_{ss} of a syntactic structure ss is built up from the lexical expressions in ss and '(' and ')', such that:

1. if ss is a lexical expression b, then $AS_{ss} = b$
2. if ss is $(\beta_{B/A}\alpha_A)$ or $(\alpha_A\beta_{A\backslash B})_B$ and if $AS_\beta = b$ and $AS_\alpha = a$, then $AS_{ss} = b(a)$

I will omit reference to the syntactic structures from now on, and I will speak of application structures simpliciter.

The following definition enables us to talk about the constituents of application structures, which are called 'functors' and 'arguments'. Notice that, strictly speaking, we should talk about *occurrences* of functors (and arguments) in an application structure. I will avoid this lengthy terminology, however, simply by using the terms 'functors' and 'arguments' to refer to occurrences of functors and arguments.

Definition 5.2 (Functors, arguments and range)
If b is an application structure $f(a_1)\ldots(a_{n-1})$, where f is a lexical expression, then:

1. f is the *functor of* b
2. a_i is the *i-th argument of f in b* (for $0 < i < n$)
3. the functor of the i-th argument of f in b is the *i-th daughter of f in b*

An application structure a is an *argument in* b iff a is b or a is an argument in an argument of the functor of b

A lexical expression f is a *functor in* b iff f is the functor of b or f is a functor in an argument of b

A functor *has* $n-1$ *arguments in* b iff it is the functor of an argument $f(a_1)\dots(a_{n-1})$ in b

A functor f in b *ranges over a functor* f' *in* b, $f > f'$, iff:

1. for some i, f' is the i-th daughter of f in b or
2. there is a functor f'' such that $f > f''$ and $f'' > f'$

I now turn to a definition of the possible translations of application structures and of quantifying arguments. For the purposes of this section it is expedient not to stick to the basic translations given in the sections 1 and 2, but to generalize over certain possible translations. Quantifying arguments are equated with the arguments which constitute type mismatches relative to a possible translation. In the following definition I will say that an *IL* term *has an* n*-th value* b iff it has a type $\langle \vec{a}, b\rangle$, where $\vec{a}$ is a sequence of types $a_1, \dots, a_{n-1}$.

Definition 5.3 (Possible translations and quantifying arguments)
If b is an application structure, then a *possible translation for* b is a function t from functors to *IL* expressions such that for any functor f in b and any i-th daughter f_i of f in b:

1. if $t(f)$ has an $(i-1)$-th value $\langle \hat{}a, c\rangle$ and if f_i has $m-1$ arguments in b, then $t(f_i)$ has an m-th value which is either a or $\langle \hat{}\langle \hat{}a, t\rangle, t\rangle$

A functor f_i which is the i-th daughter of some functor f in b is a *quantifying functor* in b under possible translation t iff

1. $t(f)$ has an $(i-1)$-th value $\langle \hat{}a, c\rangle$
2. $t(f_i)$ has an m-th value $\langle \hat{}\langle \hat{}a, t\rangle, t\rangle$, if f_i has $m-1$ arguments in b

A *quantifying argument* in b is an argument which has a quantifying functor in b.

Notice that the *FMG* basic translation constitutes a possible translation for the application structure of any syntactic structure. Furthermore, under this basic translation an argument in b is a quantifying argument in b iff it is the application structure of a quantifying noun phrase.[22]

By means of the rules of value raising and argument raising translations can be derived which have 'fitting' types. Such translations are called normalizations:

Definition 5.4 (Normalization and translation)
If b is an application structure and t is a possible translation for b, then:

22. Under this 'contextual' definition of quantifying functors it depends upon the functor of which a quantifying noun phrase makes up an argument whether the noun phrase constitutes a quantifying argument. For instance, a quantifying noun phrase would not constitute a quantifying argument if it is the first argument of Hendriks' basic translation of the transitive verb *seek*. However, by means of Hendriks' rule of argument lowering the translation of such a functor can be 'lowered' to the effect that a quantifying argument noun phrase constitutes a quantifying argument after all. I just note that such an application of argument lowering is indeed a prerequisite in Hendriks' system in order for a quantifying noun phrase under those circumstances to take part in the play of varying scope configurations.

a possible translation t' of b is a *normalization of t for b* iff:

1. for every functor f in b, if f has $n-1$ arguments in b then $t'(f)$ is derived from $t(f)$ by means of applications of $[n\text{VR}]$ and $[i\text{AR}]$, where $0 < i < n$
2. there are no quantifying arguments in b under t'

the *translation of b under normalization t'* is the *IL*-term $[b]_{t'}$ such that:

$$[f(a_1)\ldots(a_{n-1})]_{t'} = t'(f)(^\wedge[a_1]_{t'})\ldots(^\wedge[a_{n-1}]_{t'})$$

Now I want to show that for any application structure b and translation t, every quantifying argument a in b can be raised to any functor f above a by a normalization of t, provided that the type of the translation of f does not end in type e.[23] In order to prove this fact, I first define a lifting procedure which can be used to raise a single quantifying argument a in b to the main functor f of b, and I show next that under the lift of a to f, the application structure b (without a, of course) stands in the scope of a.

Definition 5.5 (The lift of a_i to f_j in b)
Let b be an application structure with main functor f, t a possible translation for b and a_i the only quantifying argument in b under t with main functor f_i. If the type of $t(f)$ does not end in type e, *the lift of a_i to f in b for t* is the translation t' which at most differs from t in that:

1. $t'(f) = [i\text{AR}](t(f))$ if the i-th argument of f in b ranges over f_i
2. for all f' in b if $f > f' > f_i$, if f' has $n-1$ arguments in b and the i-th argument of f' in b ranges over f_i, then $t'(f') = [i\text{AR}][n\text{VR}](t(f'))$

The following fact shows that the lift of a_i to f indeed establishes the scope of a_i at f. I will use $t[a/y]$ to indicate the translation t' which at most differs from t in that $[a]_{t'} = y$. (In order to be more precise, if f is the functor of a and f has $n-1$ arguments in a, then t' at most differs from t in that $t'(f) = \lambda\vec{x}\ y$, where $\vec{x}$ is a sequence of $n-1$ variables of the types of the arguments of f in a under t.)

Fact 5.1
If b is an application structure with main functor f, t a possible translation for b, a_i the only quantifying argument in b under t, and if t' is the lift of a_i to f in b for t, then:

23. The scope of a quantifying noun phrase cannot be established at a functor with a translation the type of which ends in type e. Suppose, for instance, that the definite article were translated as some (partial?) function from properties of individual concepts to individuals. Then there still is no sensible interpretation of type e of the noun phrase *the mayor of every Italian city* with *every Italian city* read wide scope. Of course, there *is* a sensible wide scope *every Italian city* reading of the sentence *the mayor of every Italian city is an Italian*. However, on the wide scope *every Italian city* reading of that sentence the quantifying noun phrase is lifted, over the definite article, to the main functor of the sentence, *takes*, the basic translation of which does have a type which ends in type t.

- $[b]_{t'} \Leftrightarrow \lambda\vec{z}\ [a_i]_t(\wedge\lambda y\ ([b]_{t[a_i/\check{}y]}(\vec{z})))$

where y is a variable of type $\hat{}a$ if $[a_i]_t$ has a type $\langle\hat{}\langle\hat{}a,t\rangle,t\rangle$, and where $\vec{z}$ is a sequence of variables of types $\vec{c}$ if the type of $[b]_{t[a_i/\check{}y]}$ is $\langle\vec{c},t\rangle$

Fact 5.1 is proved by induction on the number l of functors f' such that $f > f' > f_i$.

Basic case: $l = 0$. Take b, f, t, a_i and t' as in fact 5.1. Since $l = 0$, a_i is the i-th argument of f for some number i: $0 < i < n$ where n is the number of arguments of f. In this case t' at most differs from t in that $t'(f) = [i\mathrm{AR}](t(f))$. The translation of b under t' is the following:

$$\begin{array}{l}[b]_{t'} = [f(\vec{a_1})(a_i)(\vec{a_2})]_{t'} \Leftrightarrow \\ ([i\mathrm{AR}]([f]_t))(\vec{\alpha_1})(\wedge[a_i]_t)(\vec{\alpha_2}) \Leftrightarrow \\ \lambda\vec{z}\ [a_i]_t(\wedge\lambda y\ t(f)(\vec{\alpha_1})(y)(\vec{\alpha_2})(\vec{z})) \Leftrightarrow \\ \lambda\vec{z}\ [a_i]_t(\wedge\lambda y\ ([b]_{t[a_i/\check{}y]}(\vec{z})))\end{array}$$

where $\vec{a_1}$ and $\vec{a_2}$ abbreviate $a_1,\ldots,a_{i-1}$ and $a_{i+1},\ldots,a_{n-1}$, respectively, and $\vec{\alpha_1}$ and $\vec{\alpha_2}$ abbreviate $\wedge[a_1]_t,\ldots,\wedge[a_{i-1}]_t$ and $\wedge[a_{i+1}]_t,\ldots,\wedge[a_{n-1}]_t$, respectively; y and $\vec{z}$ are as in fact 5.1.

Induction: $l > 0$. Take b, f, t, a_i and t' as in fact 5.1. Since $l > 0$, a_i is the i-th argument of some functor f': $f > f'$ for some number i: $0 < i < n$ where n is the number of arguments of f'. Now consider the translation t'' which at most differs from t in that $t''(f') = [i\mathrm{AR}][n\mathrm{VR}](t(f'))$. The application structure a' of which f' is the main functor has the following translation under t'':

$$\begin{array}{l}[a']_{t''} = [f'(\vec{a_1})(a_i)(\vec{a_2})]_{t''} \Leftrightarrow \\ ([i\mathrm{AR}]([n\mathrm{VR}]([f']_t)))(\vec{\alpha_1})(\wedge[a_i]_t)(\vec{\alpha_2}) \Leftrightarrow \\ \lambda R\ [a_i]_t(\wedge\lambda y\ {}^\vee R(\wedge t(f')(\vec{\alpha_1})(y)(\vec{\alpha_2}))) \Leftrightarrow \\ \lambda R\ [a_i]_t(\wedge\lambda y\ {}^\vee R(\wedge[a']_{t[a_i/\check{}y]}))\end{array}$$

($\vec{a_1}$ and $\vec{a_2}$, $\vec{\alpha_1}$ and $\vec{\alpha_2}$, and y as above.) Under this translation t'' not a_i, but a' is a quantifying argument in b. By the induction hypothesis, a' can be raised to the main functor f of b by the lift t''' of a' to f in b for t'', i.e.:

$$[b]_{t'''} \Leftrightarrow \lambda\vec{z}\ [a']_{t''}(\wedge\lambda u\ ([b]_{t''[a'/\check{}u]}(\vec{z})))$$

where u is a variable of type $\hat{}c$ if $[a']_{t''}$ has a type $\langle\hat{}\langle\hat{}c,t\rangle,t\rangle$, and where $\vec{z}$ is a sequence of variables of types $\vec{c}$ if the type of $[b]_{t''[a'/\check{}u]}$ is $\langle\vec{c},t\rangle$. Since the lift t''' of a' to f in b for t'' equals the lift t' of a_i to f in b for t, and since $[b]_{t''[a'/\check{}u]}$ equals $[b]_{t[a'/\check{}u]}$ we find that:

$$\begin{array}{l}[b]_{t'} \Leftrightarrow \lambda\vec{z}\ [a']_{t''}(\wedge\lambda u\ ([b]_{t[a'/\check{}u]}(\vec{z}))) \Leftrightarrow \\ \lambda\vec{z}\ (\lambda R\ [a_i]_t(\wedge\lambda y\ {}^\vee R(\wedge[a']_{t[a_i/\check{}y]})))(\wedge\lambda u\ ([b]_{t[a'/\check{}u]}(\vec{z}))) \Leftrightarrow \\ \lambda\vec{z}\ [a_i]_t(\wedge\lambda y\ ([b]_{t[a'/[a']_{t[a_i/\check{}y]}]}(\vec{z}))) \Leftrightarrow \\ \lambda\vec{z}\ [a_i]_t(\wedge\lambda y\ ([b]_{t[a_i/\check{}y]}(\vec{z})))\end{array}$$

That completes the proof of fact 5.1.

In principle, all conceivable scope configurations can be derived by lifting quantifying arguments one by one in the right order. (When any one quantifying argument a is lifted for t, one may 'neglect' the other quantifying arguments by employing, for the time being, some translation t' which at most differs from t in that under t' a is the only quantifying argument.) A proper order of lifts is required for the cases in which two or more quantifying arguments are lifted over the same functor. For instance, if a quantifying argument a_i is lifted over a functor f_i to a functor f, and if another quantifying argument a_j is next lifted to f_i, then a_j is automatically raised over a_i, and, hence, over f. Of course, this does not mean that the scope of a_j can not be fixed at f_i, only that it should be lifted before a_i is. Furthermore, if a_j is a quantifying argument within a quantifying argument a_i, then a_i should be lifted before a_j is. In such a case, a_j can only be lifted over functors which range over a_i if a_j is lifted over (the main functor of) a_i itself.

Division and argument raising

In this section I will show that any scope configuration that is derived by lifting quantifiers in the way defined in the preceding section can also be derived using [VR] and [GD] only. More in particular, I will show for any application structure b, possible translation t and normalization t' of t for b, that every application of [AR] in the derivation of the translation of the functor f of b can be replaced by an application of [VR] and a corresponding division of the translation of the functor of an argument of f, preserving the meaning.

We will be concerned with application structures of the following form:

$$b = f(a_1)\dots(a_{i-1})(a_i)(a_{i+1})\dots(a_{n-1}), \text{ where } 0 < n \text{ and for any } i\text{: } 0 < i < n\text{:}$$
$$a_i = f_i(a_{i,1})\dots(a_{i,(m-1)}) \text{ with } 0 \leq m$$

Let t be a possible translation for b and t' a normalization of t for b. Let us assume that the first application of [AR] in the derivation of $t(f)$ is an application of [iAR] to ϕ, where ϕ is derived from $t(f)$ without [AR], and that $t'(f)$ is obtained from [iAR](ϕ) by the successive application of the type shifts $[\mathrm{TC}_1], \dots, [\mathrm{TC}_k]$. The resulting translation $t'(f)$ then can be indicated as $[\mathrm{TC}_{k\dots1}][i\mathrm{AR}](\phi)$. Furthermore, let $t'(f_i)$ be obtained from $t(f_i)$ by means of the successive application of the type shifts $[\mathrm{TC}_{i,1}], \dots, [\mathrm{TC}_{i,l}]$ to ψ, where ψ is derived from $t(f_i)$. The translation $t'(f_i)$ can be indicated as $[\mathrm{TC}_{i,l\dots1}](\psi)$. Under these assumptions, the translation of b above boils down to the following formula:

$$([\mathrm{TC}_{k\dots1}][i\mathrm{AR}](\phi))(\vec{\alpha_1})(\hat{}([\mathrm{TC}_{i,l\dots1}](\psi))(\vec{\varepsilon}))(\vec{\alpha_2})$$

where $\vec{\alpha_1}$ and $\vec{\alpha_2}$ indicate the sequences of the intensions of the translations of $(a_1)\dots(a_{i-1})$ and $(a_{i+1})\dots(a_{n-1})$, respectively, and $\vec{\varepsilon}$ indicates the sequence of the intensions of the translations of the arguments of f_i.

Now it can be shown that, if t' is a normalization of t for b, then the application of [iAR] in the derivation of $t'(f)$ can be meaning preservingly replaced by an application of [VR] and a corresponding application of [GD] to the translation of the main functor of the i-th argument. However, in order to show that in fact all occurrences of [AR] in the derivation of t' can be replaced in this way, we will have to relax the assumption that t' is a normalization. Among the type shifts $[\mathrm{TC}_1], \ldots, [\mathrm{TC}_k]$, not only raisings [$i$AR] and [$n$VR] may occur, but also divisions $[n\mathrm{GD}_{\vec{e}}]$. So, I will prove the following fact:

Fact 5.2
If

1. for all f: $1 \leq f \leq k$, $[\mathrm{TC}_f]$ is [nVR] or $[n\mathrm{GD}_{\vec{e}}]$ or [gAR] $(0 < g < n)$
2. for all f: $1 \leq f \leq l$, $[\mathrm{TC}_{i,f}]$ is [mVR] or [gAR] $(0 < g < m)$
3. [iAR] occurs j times in $[\mathrm{TC}_1], \ldots, [\mathrm{TC}_k]$ and in $[\mathrm{TC}_{i,1}], \ldots, [\mathrm{TC}_{i,l}]$

then

$$([\mathrm{TC}_{k \ldots 1}][i\mathrm{AR}](\phi))(\vec{\alpha_1})({}^\wedge([\mathrm{TC}_{i,l \ldots 1}](\psi))(\vec{\varepsilon}))(\vec{\alpha_2}) \Leftrightarrow$$
$$([\mathrm{TC}_{k \ldots 2}][i\mathrm{VR}_{\langle\vec{c},t\rangle}](\phi))(\vec{\alpha_1})({}^\wedge([\mathrm{TC}_{i,l \ldots 1}][m\mathrm{GD}_{\vec{c}}](\psi))(\vec{\varepsilon}))(\vec{\alpha_2})$$

where if ϕ has a type $\langle\vec{a}, \langle\check{b}, \langle\vec{c}, t\rangle\rangle\rangle$, $\vec{a}$ is the sequence of the types of the sequence of translations $\vec{\alpha_1}$ and $\check{b}$ is the m-th value of ψ

By means of this fact all applications of [AR] in the derivation of a normalization t' of a possible translation t for an application structure b can be replaced in the following way. Starting with the first application of [AR] in the derivation of the translation of the functor of b, one successively replaces all applications of [AR] by [VR] in the derivation of the translation of this functor. When the applications of [AR] to the main functor f have been replaced, the applications of [AR] to the functors of the arguments of f can be replaced in the same way. Since we start from a normalization of a possible translation for b, the three conditions upon the replacement of applications of [AR] in the derivation of t'' will always be satisfied when the replacements are executed in the order indicated.

In the proof of fact 5.2 we need a device to refer to certain arbitrarily deeply embedded applications of [GD]:

Embedded division
If b is a type, then

- $b^0 = b$
- $b^{n+1} = \langle\check{\langle}\check{b}^n, t\rangle, t\rangle$ $(0 \leq n)$
- if ϕ is of type b^1, and $\vec{c}$ a sequence of types, then
$[1\mathrm{GD}^1_{\vec{c}}](\phi) = [1\mathrm{GD}_{\vec{c}}](\phi)$
- if ϕ is of type b^{n+1} $(0 < n)$ and $\vec{c}$ a sequence of types, then
$[1\mathrm{GD}^{n+1}_{\vec{c}}](\phi) = \lambda R\ \phi({}^\wedge\lambda p\ {}^\vee R({}^\wedge[1\mathrm{GD}^n_{\vec{c}}]({}^\vee p)))$
where p and R are variables of types $\check{b}^n$ and $\check{\langle}\check{\langle}\check{\langle}\check{b}, \langle\vec{c}, t\rangle\rangle, \langle\vec{c}, t\rangle\rangle^{n-1}, t\rangle$
- if ϕ is of type $\langle\vec{a}, b^n\rangle$ $(0 < n)$ and $\vec{a}$ is a sequence of types $a_1, \ldots, a_{m-1}$, then

$[m\mathrm{GD}^n_{\vec{c}}](\phi) = \lambda\vec{x}\ [1\mathrm{GD}^n_{\vec{c}}](\phi(\vec{x}))$
where $\vec{x}$ is a sequence of variables of types $a_1, \ldots, a_{m-1}$

So, if ϕ is of type b^n $(0 < n)$, then $[1\mathrm{GD}^n_{\vec{c}}](\phi)$ is of type $\langle\hat{}\langle\vec{b}, \langle\vec{c}, t\rangle\rangle, \langle\vec{c}, t\rangle\rangle^{n-1}$.

Employing the notation device for embedded divisions we can prove the following two propositions, which together entail fact 5.2:

Proposition 1
If
1. for all f: $1 \le f \le k$, $[\mathrm{TC}_f]$ is $[n\mathrm{VR}]$ or $[n\mathrm{GD}_{\vec{e}}]$ or $[g\mathrm{AR}]$ $(0 < g < n)$
2. $[i\mathrm{AR}]$ occurs $j-1$ times in $[\mathrm{TC}_1], \ldots, \mathrm{TC}_k]$ $(0 < j)$

then

$$([\mathrm{TC}_{k\ldots1}][i\mathrm{AR}](\phi))(\vec{x})(T) \Leftrightarrow ([\mathrm{TC}_{k\ldots1}][i\mathrm{VR}_{\langle\vec{c},t\rangle}](\phi))(\vec{x})(^\wedge[1\mathrm{GD}^j_{\vec{c}}](^\vee T))$$

where $\vec{x}$ is a sequence of variables and T a variable, all of appropriate types

Proposition 2
If
1. for all f: $1 \le f \le l$, $[\mathrm{TC}_{i,f}]$ is $[m\mathrm{VR}]$ or $[g\mathrm{AR}]$ $(0 < g < m)$
2. $[i\mathrm{AR}]$ occurs $j-1$ times in $[\mathrm{TC}_{i,1}], \ldots, \mathrm{TC}_{i,l}]$ $(0 < j)$

then

$$[1\mathrm{GD}^j_{\vec{c}}](([\mathrm{TC}_{i,l\ldots1}](\psi))(\vec{y})) \Leftrightarrow ([\mathrm{TC}_{i,l\ldots1}][m\mathrm{GD}_{\vec{c}}](\psi))(\vec{y})$$

where $\vec{y}$ is a sequence of variables of appropriate types

Proposition 1 is proved by induction on the number k of type changes applied to ϕ. Basic case: $k = 0$. In that case $[i\mathrm{AR}]$ does not occur in $[\mathrm{TC}_1]$, …, $[\mathrm{TC}_k]$. So, $j = 1$, $[1\mathrm{GD}^j_{\vec{c}}]$ is $[1\mathrm{GD}_{\vec{c}}]$, and, by means of fact 2.1:

$$([i\mathrm{AR}](\phi))(\vec{x})(T) \Leftrightarrow ([i\mathrm{VR}_{\langle\vec{c},t\rangle}](\phi))(\vec{x})(^\wedge[1\mathrm{GD}_{\vec{c}}](^\vee T))$$

Induction: $k > 0$. Given the first requirement in proposition 1, four cases can be distinguished. $[\mathrm{TC}_k]$ is either $[n\mathrm{VR}]$ or $[n\mathrm{GD}_{\vec{e}}]$ $(i < n)$, or $[f\mathrm{AR}]$ ($f < n$ and $i \neq f$), or $[i\mathrm{AR}]$. In the following equations $\Leftrightarrow^*$ refers to the induction hypothesis.

1. $[\mathrm{TC}_k] = [n\mathrm{VR}]$.
   $([n\mathrm{VR}][\mathrm{TC}_{(k-1)\ldots1}][i\mathrm{AR}](\phi))(\vec{x})(T) \Leftrightarrow$
   $([n\mathrm{VR}](\lambda\vec{x}\lambda T\ ([\mathrm{TC}_{(k-1)\ldots1}][i\mathrm{AR}](\phi))(\vec{x})(T)))(\vec{x})(T) \Leftrightarrow^*$
   $([n\mathrm{VR}](\lambda\vec{x}\lambda T\ ([\mathrm{TC}_{(k-1)\ldots1}][i\mathrm{VR}_{\langle\vec{c},t\rangle}](\phi))(\vec{x})([1\mathrm{GD}^j_{\vec{c}}](^\vee T))))(\vec{x})(T) \Leftrightarrow$
   $([n\mathrm{VR}][\mathrm{TC}_{(k-1)\ldots1}][i\mathrm{VR}_{\langle\vec{c},t\rangle}](\phi))(\vec{x})([1\mathrm{GD}^j_{\vec{c}}](^\vee T))$
2. $[\mathrm{TC}_k] = [n\mathrm{GD}_{\vec{e}}]$. Similar.
3. $[\mathrm{TC}_k] = [f\mathrm{AR}]$, where $i \neq f$. As in 1.
4. $[\mathrm{TC}_k] = [i\mathrm{AR}]$. Since $[i\mathrm{AR}]$ occurs $j-1$ times in $[\mathrm{TC}_1]$, …, $[\mathrm{TC}_k]$, $1 < j$ and $[i\mathrm{AR}]$ occurs $j-2$ times in $[\mathrm{TC}_1]$, …, $[\mathrm{TC}_{k-1}]$. The induction hypothesis is:

$([\mathrm{TC}_{(k-1)\ldots 2}][i\mathrm{AR}](\phi))(\vec{x})(T) \Leftrightarrow$
$([\mathrm{TC}_{(k-1)\ldots 2}][i\mathrm{VR}_{\langle\vec{c},t\rangle}](\phi))(\vec{x})(^\wedge[1\mathrm{GD}^{j-1}_{\vec{c}}](^\vee T))$

Now we find that:

$([i\mathrm{AR}][\mathrm{TC}_{(k-1)\ldots 2}][i\mathrm{AR}](\phi))(\vec{x})(T) \Leftrightarrow$
$\lambda\vec{z}\ ^\vee T(^\wedge\lambda t\ ([\mathrm{TC}_{(k-1)\ldots 2}][i\mathrm{AR}](\phi))(\vec{x})(t)(\vec{z})) \Leftrightarrow^*$
$\lambda\vec{z}\ ^\vee T(^\wedge\lambda t\ ([\mathrm{TC}_{(k-1)\ldots 2}][i\mathrm{VR}_{\langle\vec{c},t\rangle}](\phi))(\vec{x})(^\wedge[1\mathrm{GD}^{j-1}_{\vec{c}}](^\vee t))(\vec{z})) \Leftrightarrow^\dagger$
$\lambda\vec{z}\ ([1\mathrm{GD}^{j}_{\vec{c}}](^\vee T))(^\wedge\lambda t\ ([\mathrm{TC}_{(k-1)\ldots 2}][i\mathrm{VR}_{\langle\vec{c},t\rangle}](\phi))(\vec{x})(t)(\vec{z}))) \Leftrightarrow$
$([i\mathrm{AR}][\mathrm{TC}_{(k-1)\ldots 2}][i\mathrm{VR}_{\langle\vec{c},t\rangle}](\phi))(\vec{x})(^\wedge[1\mathrm{GD}^{j}_{\vec{c}}](^\vee T))$

where $\vec{z}$ is a sequence of variables and t a variable, all of appropriate types
† by the definition of $[1\mathrm{GD}^{j}_{\vec{c}}]$

That completes the proof of the first proposition.

Proof of the second proposition. Since $\vec{y}$ is a sequence of $m-1$ variables:

$[1\mathrm{GD}^{j}_{\vec{c}}](([\mathrm{TC}_{i,l\ldots 1}](\psi))(\vec{y})) \Leftrightarrow ([m\mathrm{GD}^{j}_{\vec{c}}][\mathrm{TC}_{i,l\ldots 1}](\psi))(\vec{y})$

So, it suffices to prove the equivalence:

$([m\mathrm{GD}^{j}_{\vec{c}}][\mathrm{TC}_{i,l\ldots 1}](\psi))(\vec{y}) \Leftrightarrow ([\mathrm{TC}_{i,l\ldots 1}][m\mathrm{GD}_{\vec{c}}](\psi))(\vec{y})$

which is proved by induction on the number l of type changes applied to ψ.
Basic case: $l = 0$. In that case $[m\mathrm{VR}]$ does not occur in $[\mathrm{TC}_{i,1}], \ldots, [\mathrm{TC}_{i,l}]$. So, $j = 1$, $[m\mathrm{GD}^{j}_{\vec{c}}]$ is $[m\mathrm{GD}_{\vec{c}}]$, and:

$([m\mathrm{GD}^{j}_{\vec{c}}][\mathrm{TC}_{i,l\ldots 1}](\psi))(\vec{y}) \Leftrightarrow ([m\mathrm{GD}_{\vec{c}}](\psi))(\vec{y}) \Leftrightarrow ([\mathrm{TC}_{i,l\ldots 1}][m\mathrm{GD}_{\vec{c}}](\psi))(\vec{y})$

Induction: $l > 0$. Given the first requirement in proposition 2, there are only two cases to be considered. $[\mathrm{TC}_{i,l}]$ is either $[f\mathrm{AR}]$ ($f < m$), or $[m\mathrm{VR}]$.

1. $[\mathrm{TC}_{i,l}] = [f\mathrm{AR}]$.

   $([m\mathrm{GD}^{j}_{\vec{c}}][f\mathrm{AR}][\mathrm{TC}_{i,(l-1)\ldots 1}](\psi))(\vec{y}) \Leftrightarrow$
   $([f\mathrm{AR}][m\mathrm{GD}^{j}_{\vec{c}}][\mathrm{TC}_{i,(l-1)\ldots 1}](\psi))(\vec{y}) \Leftrightarrow^*$
   $([f\mathrm{AR}][\mathrm{TC}_{i,(l-1)\ldots 1}][m\mathrm{GD}_{\vec{c}}](\psi))(\vec{y})$

2. $[\mathrm{TC}_{i,l}] = [m\mathrm{VR}]$. Since $[m\mathrm{VR}]$ occurs $j-1$ times in $[\mathrm{TC}_{i,1}], \ldots, [\mathrm{TC}_{i,l}]$, $1 < j$ and $[m\mathrm{VR}]$ occurs $j-2$ times in $[\mathrm{TC}_{i,1}], \ldots, [\mathrm{TC}_{i,(l-1)}]$. So, the induction hypothesis reads:

   $([m\mathrm{GD}^{j-1}_{\vec{c}}]([\mathrm{TC}_{i,(l-1)\ldots 1}](\psi)))(\vec{y}) \Leftrightarrow ([\mathrm{TC}_{i,(l-1)\ldots 1}][m\mathrm{GD}_{\vec{c}}](\psi))(\vec{y})$

   Since:

   $[m\mathrm{GD}^{j}_{\vec{c}}][m\mathrm{VR}](\phi) \Leftrightarrow \lambda\vec{x}\ [1\mathrm{GD}^{j}_{\vec{c}}][1\mathrm{VR}](\phi(\vec{x})) \Leftrightarrow$
   $\lambda\vec{x}\lambda R\ (\lambda S\ ^\vee S(^\wedge\phi(\vec{x})))(^\wedge\lambda p\ ^\vee R\ (^\wedge[1\mathrm{GD}^{j-1}_{\vec{c}}](^\vee p))) \Leftrightarrow$
   $\lambda\vec{x}\lambda R\ ^\vee R\ (^\wedge[1\mathrm{GD}^{j-1}_{\vec{c}}](\phi(\vec{x}))) \Leftrightarrow [m\mathrm{VR}][m\mathrm{GD}^{j-1}_{\vec{c}}](\phi)$

   we find that:

   $([m\mathrm{GD}^{j}_{\vec{c}}][m\mathrm{VR}][\mathrm{TC}_{i,(l-1)\ldots 1}](\psi))(\vec{y}) \Leftrightarrow$

$([m\mathrm{VR}][m\mathrm{GD}_{\vec{c}}^{j-1}][\mathrm{TC}_{i,(l-1)\ldots 1}](\psi))(\vec{y}) \Leftrightarrow^*$
$([m\mathrm{VR}][\mathrm{TC}_{i,(l-1)\ldots 1}][m\mathrm{GD}_{\vec{c}}](\psi))(\vec{y})$

That completes the proof of the second proposition, and, with the proof of proposition 1, of fact 5.2.

Derived translations

I will now show in more detail how certain readings of some sentence discussed in this chapter are derived with the type changing system. I start with two examples which are also discussed within the *DMG* framework. I only indicate the type shifts which are required to obtain the *DMG* or *DMG(2)* truth conditions of these examples.

(I.2) If a_i farmer owns a_j donkey, he_i beats it_j.
[2AR]([1AR]([3VR](*own'*))) [1DAR](*if'*)

(I.3) Every farmer who owns a_i donkey beats it_i.
[1AR]([2VR](*own'*)) [2AR]([3VR](*who'*)) [1DAR](*every'*)

(3) If a_i client comes in, you pamper him_i. You offer him_i a cup of coffee.
[1AR]([2VR](*come in'*)) [2VR](*pamper him$_i$'*)
[1DAR]([2GD$_{\langle\hat{}t,t\rangle}$](*if'*)) [1AR](*.'*)

(5) Either there is no_i bathroom here, or it_i is in a funny place. In any case, it_i is not on the ground floor.
[1VR$_{\langle\hat{}\langle\hat{}t,t\rangle,t\rangle}$]([2VR](*is here'*)) [1DGD$_{\langle\hat{}t,t\rangle}$](*no$_i$ bathroom'*)
[2VR](*is in a funny place'*) [1AR]([2GD$_{\langle\hat{}t,t\rangle}$](*or'*)) [1AR](*.'*)

(6) If there is a bathroom here, it is in a funny place. In any case, it is not on the ground floor. (Like example 3.)

(8) No client that comes in is offered coffee. He is directly sent up to me. (Like example 7.)

I now turn to a more detailed presentation of the flexible treatment of some complicated examples. I start with example 5:

(5) Every_i customer is offered coffee, that is, if he_i looks wealthy. He_i is sent up to me as quickly as possible. If he_i doesn't look wealthy, he_i can wait.

The expressions *customer*, *is offered coffee*, *look wealthy*, *is sent up to me as quickly as possible*, and *can wait* are associated with the basic translations $\lambda x\ \mathtt{cu}({}^{\vee}x)$, $\lambda x\ \mathtt{co}({}^{\vee}x)$, $\lambda x\ \mathtt{we}({}^{\vee}x)$, $\lambda x\ \mathtt{su}({}^{\vee}x)$, $\lambda x\ \mathtt{wa}({}^{\vee}x)$, all of type $\langle\hat{e},t\rangle$. The main constituents of example 5 have the following basic translations:

(a) every_i customer: $\lambda Q\ \forall d_i(\mathtt{cu}(d_i) \to {}^{\vee}Q({}^{\wedge}d_i))$
(b) is offered coffee: $\lambda x\ \mathtt{co}({}^{\vee}x)$
(c) if he_i looks wealthy: $\lambda q\ (\mathtt{we}(d_i) \to {}^{\vee}q)$
(d) he_i is sent up to me as quickly as possible: $\mathtt{su}(d_i)$
(e) if he_i doesn't look wealthy, he_i can wait: $(\neg\mathtt{we}(d_i) \to \mathtt{wa}(d_i))$

Furthermore, the following derived translations are used (variables are typed as follows: R: $\hat{}\langle\hat{}t,t\rangle$; $\mathcal{R}$: $\hat{}\langle\hat{}\langle\hat{}\langle\hat{}t,t\rangle,t\rangle,t\rangle$; T: $\hat{}\langle\hat{}\langle\hat{}e,t\rangle,t\rangle$; P: $\hat{}\langle\hat{}\langle\hat{}t,t\rangle,t\rangle$; $\mathcal{P}$: $\hat{}\langle\hat{}\langle\hat{}\langle\hat{}\langle\hat{}t,t\rangle,t\rangle,t\rangle,t\rangle$):

(b′) is offered coffee: $[1\text{AR}]([2\text{VR}]([2\text{VR}](\lambda x\ \texttt{co}({}^{\vee}x)))) \Leftrightarrow$
$\lambda T\lambda\mathcal{R}\ {}^{\vee}T({}^{\wedge}\lambda x\ {}^{\vee}\mathcal{R}({}^{\wedge}\lambda R\ {}^{\vee}R({}^{\wedge}\texttt{co}({}^{\vee}x))))$

(c′) if he$_i$ looks wealthy: $[1\text{AR}]([2\text{VR}]([1\text{GD}_{\hat{}\langle\hat{}t,t\rangle}](\lambda q\ (\texttt{we}(d_i) \to {}^{\vee}q)))) \Leftrightarrow$
$\lambda\mathcal{P}\lambda\mathcal{R}\ {}^{\vee}\mathcal{P}({}^{\wedge}\lambda P\ {}^{\vee}\mathcal{R}({}^{\wedge}\lambda R\ (\texttt{we}(d_i) \to {}^{\vee}P(R))))$

Finally, we use the following derived translations of the sequencing operation:

(.′) $[1\text{AR}]([3\text{VR}]([1\text{AR}](\lambda p\lambda q\ ({}^{\vee}p \wedge {}^{\vee}q)))) \Leftrightarrow$
$\lambda\mathcal{P}\lambda q\lambda R\ {}^{\vee}\mathcal{P}({}^{\wedge}\lambda P\ {}^{\vee}R({}^{\wedge\vee}P({}^{\wedge}\lambda p\ ({}^{\vee}p \wedge {}^{\vee}q))))$

(.″) $[1\text{AR}](\lambda p\lambda q\ ({}^{\vee}p \wedge {}^{\vee}q)) \Leftrightarrow$
$\lambda P\lambda q\ {}^{\vee}P({}^{\wedge}\lambda p\ ({}^{\vee}p \wedge {}^{\vee}q))$

Translation (b′) of *is offered coffee* is applied to the intension of translation (a) of the noun phrase *every$_i$ customer*. The resulting translation of *every$_i$ customer is offered coffee* can be reduced to the following expression:

(f) $\lambda\mathcal{R}\ \forall d_i(\texttt{cu}(d_i) \to {}^{\vee}\mathcal{R}({}^{\wedge}\lambda R\ {}^{\vee}R({}^{\wedge}\texttt{co}(d_i))))$

The translation of the first sentence of example 5 is obtained by applying translation (c′) of the relativizing phrase *if he looks wealthy* to the intension of (f). This translation can be reduced to the following expression:

(g) $\lambda\mathcal{R}\ \forall d_i(\texttt{cu}(d_i) \to {}^{\vee}\mathcal{R}({}^{\wedge}\lambda R\ (\texttt{we}(d_i) \to {}^{\vee}R({}^{\wedge}\texttt{co}(d_i)))))$

Translation (.′) of the sequencing operation can be applied, first, to the intension of translation (g) of the first sentence of example 5, and, second, to the intension of translation (d) of the second sentence. The result can be reduced to (h):

(h) $\lambda R\ \forall d_i(\texttt{cu}(d_i) \to {}^{\vee}R({}^{\wedge}(\texttt{we}(d_i) \to (\texttt{co}(d_i) \wedge \texttt{su}(d_i)))))$

Translation (.″) of the sequencing operation can be applied, next, to the intension of translation (h) of the first two sentences of example 5, and to the intension of translation (e) of the third sentence. The result is the following translation of example 5:

(i) $\forall d_i(\texttt{cu}(d_i) \to ((\texttt{we}(d_i) \to (\texttt{co}(d_i) \wedge \texttt{su}(d_i))) \wedge$
$(\neg\texttt{we}(d_i) \to \texttt{wa}(d_i))))$

The next example is 9:

(9) If a$_i$ farmer owns a$_j$ donkey, he$_i$ stones it$_j$. If he$_i$ leases it$_j$, he$_i$ pleases it$_j$.

The following translations of the main constituents are used (the variables introduced by the type shifts are typed as follows: T, U: $\hat{}\langle\hat{}\langle\hat{}e,t\rangle,t\rangle$; R: $\hat{}\langle\hat{}t,t\rangle$; P: $\hat{}\langle\hat{}\langle\hat{}t,t\rangle,t\rangle$):

(a) a$_i$ farmer: $\lambda Q\ \exists d_i(\texttt{fa}(d_i) \wedge {}^{\vee}Q({}^{\wedge}d_i))$
(b) a$_j$ donkey: $\lambda Q\ \exists d_j(\texttt{do}(d_j) \wedge {}^{\vee}Q({}^{\wedge}d_j))$
(c) own: $[2\text{AR}]([1\text{AR}]([3\text{VR}](\lambda y\lambda x\ \texttt{own}({}^{\vee}y)({}^{\vee}x)))) \Leftrightarrow$
$\lambda T\lambda U\lambda R\ {}^{\vee}U({}^{\wedge}\lambda x\ {}^{\vee}T({}^{\wedge}\lambda y\ {}^{\vee}R({}^{\wedge}\texttt{own}({}^{\vee}y)({}^{\vee}x))))$

(d) he$_i$ stones it$_j$: $\mathtt{stone}(d_j)(d_i)$
(e) if: [1DAR]([3VR]($\lambda p \lambda q\ ({}^{\vee}p \rightarrow {}^{\vee}q)$)) $\Leftrightarrow$
$\lambda P \lambda q \lambda R\ \neg {}^{\vee}P({}^{\wedge}\lambda p\ \neg {}^{\vee}R({}^{\wedge}({}^{\vee}p \rightarrow {}^{\vee}q)))$
(f) if he$_i$ leases it$_j$, he$_i$ pleases it$_j$: $\mathtt{lease}(d_j)(d_i) \rightarrow \mathtt{please}(d_j)(d_i)$
(.″) [1AR]($\lambda p \lambda q\ ({}^{\vee}p \wedge {}^{\vee}q)$) $\Leftrightarrow$
$\lambda P \lambda q\ {}^{\vee}P({}^{\wedge}\lambda p\ ({}^{\vee}p \wedge {}^{\vee}q))$

The translation of the antecedent of the first sentence of example 9 is obtained by applying translation (c) to the intensions of translations (b) and (a). The result can be reduced to the following expression:

(g) $\lambda R\ \exists d_i(\mathtt{fa}(d_i) \wedge \exists d_j(\mathtt{do}(d_j) \wedge {}^{\vee}R({}^{\wedge}\mathtt{own}(d_j)(d_i))))$

The translation of the first sentence of example 9 consists in the application of (e) to the intensions of (g) and (d), which can be reduced to the following expression:

(h) $\lambda R\ \neg\exists d_i(\mathtt{fa}(d_i) \wedge \exists d_j(\mathtt{do}(d_j) \wedge \neg {}^{\vee}R({}^{\wedge}\mathtt{own}(d_j)(d_i) \rightarrow \mathtt{stone}(d_j)(d_i)))) \Leftrightarrow$
$\lambda R\ \forall d_i(\mathtt{fa}(d_i) \rightarrow \forall d_j(\mathtt{do}(d_j) \rightarrow {}^{\vee}R({}^{\wedge}\mathtt{own}(d_j)(d_i) \rightarrow \mathtt{stone}(d_j)(d_i))))$

Finally, (.″) is applied to the intensions of (h) and (f), the result of which gives us the following formula as the translation of example 9:

(i) $\forall d_i(\mathtt{farmer}(d_i) \rightarrow \forall d_j(\mathtt{donkey}(d_j) \rightarrow ((\mathtt{own}(d_j)(d_i) \rightarrow \mathtt{stone}(d_j)(d_i)) \wedge$
$(\mathtt{lease}(d_j)(d_i) \rightarrow \mathtt{please}(d_j)(d_i)))))$

The last example is 10:

(10) No$_i$ farmer beats a donkey, if he$_i$ isn't insane. He$_i$ will not yell at it either. If he$_i$ is insane, then he$_i$ might beat a donkey.

As was indicated in section 3, this example is structurally similar to example 5. However, the example is treated differently, since the subject noun phrase *no$_i$ farmer*, which has wide scope, is downward monotonic. I only show the part of the treatment which deviates from the treatment of example 5.

The following translations are used (I use τ as an abbreviation of the type $\langle\hat{}\langle\hat{}\langle\hat{}t,t\rangle,t\rangle,t\rangle$; variables are typed as in the treatment of example 5, the type of $\mathcal{T}$, moreover, is $\hat{}\langle\hat{}\langle\hat{}e,\langle\hat{}\tau,t\rangle\rangle,\langle\hat{}\tau,t\rangle\rangle$ and the type of $\mathcal{Q}$ is $\hat{}\langle\hat{}e,\langle\hat{}\tau,t\rangle\rangle$):

(a) beats a donkey: [1VR$_{\langle\hat{}\tau,t\rangle}$]([2VR]([2VR]($\lambda x\ \mathtt{bad}({}^{\vee}x)$))) $\Leftrightarrow$
$\lambda \mathcal{T} \lambda \mathcal{R}\ {}^{\vee}\mathcal{T}({}^{\wedge}\lambda x \lambda \mathcal{R}\ {}^{\vee}\mathcal{R}({}^{\wedge}\lambda R\ {}^{\vee}R({}^{\wedge}\mathtt{bad}({}^{\vee}x))))(\mathcal{R})$
(b) no$_i$ farmer: [1DGD$_{\hat{}\tau}$]($\lambda Q\ \neg\exists d_i(\mathtt{fa}(d_i) \wedge {}^{\vee}Q({}^{\wedge}d_i))$) $\Leftrightarrow$
$\lambda \mathcal{Q} \lambda \mathcal{R}\ \neg\exists d_i(\mathtt{fa}(d_i) \wedge {}^{\vee}\mathcal{Q}({}^{\wedge}d_i)(\mathcal{R}^*))$

The application of (a) to the intension of (b) gives the following translation of the antecedent of the example:

(c) $\lambda \mathcal{R}\ \forall d_i(\mathtt{fa}(d_i) \rightarrow {}^{\vee}\mathcal{R}({}^{\wedge}\lambda R\ {}^{\vee}R({}^{\wedge}\neg\mathtt{bad}(d_i))))$[24]

---

24. $(\lambda \mathcal{T} \lambda \mathcal{R}\ {}^{\vee}\mathcal{T}({}^{\wedge}\lambda x \lambda \mathcal{R}\ {}^{\vee}\mathcal{R}({}^{\wedge}\lambda R\ {}^{\vee}R({}^{\wedge}\mathtt{bad}({}^{\vee}x))))(\mathcal{R}))({}^{\wedge}\lambda \mathcal{Q} \lambda \mathcal{R}\ \neg\exists d_i(\mathtt{fa}(d_i) \wedge {}^{\vee}\mathcal{Q}({}^{\wedge}d_i)(\mathcal{R}^*))) \Leftrightarrow$
$\lambda \mathcal{R}\ \neg\exists d_i(\mathtt{fa}(d_i) \wedge {}^{\vee}\mathcal{R}^*({}^{\wedge}\lambda R\ {}^{\vee}R({}^{\wedge}\mathtt{bad}(d_i)))) \Leftrightarrow$

Proceeding, next, as in the treatment of example 5, we get the following translation of example 10:

(d) $\forall d(\mathtt{farmer}(d) \rightarrow ((\mathtt{sane}(d) \rightarrow (\neg\mathtt{bad}(d) \wedge \neg\mathtt{yell\_at\_it}(d))) \wedge (\neg\mathtt{sane}(d) \rightarrow \mathtt{maybe\_bad}(d))))$

$\lambda\mathcal{R}\ \neg\exists d_i(\mathtt{fa}(d_i) \wedge \neg{}^{\vee}\mathcal{R}({}^{\wedge}\lambda R\ {}^{\vee}R({}^{\wedge}\neg\mathtt{bad}(d_i)))) \Leftrightarrow$
$\lambda\mathcal{R}\ \forall d_i(\mathtt{fa}(d_i) \rightarrow {}^{\vee}\mathcal{R}({}^{\wedge}\lambda R\ {}^{\vee}R({}^{\wedge}\neg\mathtt{bad}(d_i))))$

Chapter 4

Existential disclosure

The work of Kamp [1981] and Heim [1982] in the early eighties has started a new branch of semantic theorizing within the format of discourse representation theory (*DRT*). More recently, compositional, dynamic reformulations of the *DRT* framework have been given that enhance comparison of *DRT* with more classical semantic theories, in particular, Montague grammar, and that enable an integration of results (Barwise [1987], Rooth [1987], Asher and Wada [1988], Zeevat [1989], Groenendijk and Stokhof [1990a], Muskens [1990]). Groenendijk and Stokhof [1990a] in particular formulates a dynamic Montague grammar (*DMG*), in which the paradigmatic Montague grammar of the seventies is adapted in order to incorporate *DRT*-results.

In this chapter I want to show how existing treatments of relational nouns, adverbial modification and tense in discourse can be formulated within such dynamic frameworks. The choice of these topics is not arbitrary. All three seem to involve the notion of an implicit argument. Relational nouns appear to have implicit object arguments which can be specified by complement phrases. Many adverbs can be interpreted as predicates that range over events which are implicitly referred to by verb phrases. And in temporal discourse reference times implicitly referred to in one sentence get related to the ones referred to in subsequent sentences.

A dynamic semantics provides a natural framework for the treatment of these phenomena. In a dynamic semantics nouns and verbs with implicit arguments can be interpreted, on a par with nouns and verbs without implicit arguments, as functions from individuals to sentence denotations, that is, to context change potentials. The expressions which carry implicit arguments can be taken to introduce objects to the context which are available for optional adnominal, adverbial or temporal specification.

The proposals made in this chapter are programmatic, compositional reformulations of existing treatments of relational nouns, adverbs and tense. The point is to show that a compositional system of dynamic interpretation provides a natural framework for the description of the phenomena involved. Although the reformulations are cast within the framework of *DMG*, such reformulations are not restricted to this particular framework. As, I hope, the following sections show, a completely parallel treatment of the phenomena at issue is possible in any compositional reformulation of original *DRT*. *DMG* is used exemplary here, but, also, because it is relatively easy to use.

This chapter is organized as follows. In section 1, I review very shortly the rudimentary but compositional dynamic reformulation of *DRT* into *DMG* proposed by Groenendijk and Stokhof. In this section I show that dynamic interpretation comes along with the possibility of what is called 'existential disclosure', the possibility to address (dynamic) existentially closed (implicit) arguments as if they were free variables after all. The subsequent sections 2–4 show how existential disclosure can be employed to model the specification of implicit arguments of nouns and verbs by means of adnominal modification, adverbial modification and temporal operators, respectively.

1 Implicit arguments in dynamic semantics

Certain nouns and nominalized constructions come with implicit arguments which can, but need not, be specified by complement phrases. For instance, the relational noun *sister* may be used to denote a set of sisters without indicating the individuals of which they are a sister. Still, these individuals may be specified by a complement phrase as in *sister of John*. Similarly, we can talk about the destruction of the city without explicitly mentioning the destructive agents, but the agents can be specified, as in *the destruction of the city by the extra-terrestrials*.

Verbs, too, have been argued to come with implicit arguments which license optional specification. For instance, the sentence *The pigeon flew* has been taken to describe a certain event or change of location the source and goal of which can be specified by (optional) adverbial phrases like *from Sevilla* and *to Amsterdam*. Similarly, the time of such an event can be specified by a phrase like *last month*, or referred back to, as in the continuation *And it flew back to Sevilla afterwards*.

Implicit arguments are indefinite objects, that is, they are assumed to be, eventually, existentially closed arguments. So, a sister is supposed to be a sister of *someone*, a destruction is conceived to be a destruction of *something* by *someone* (or *something*), and the statement that the pigeon flew appears to assert the 'existence' of *some* event of flying with *some* source and goal location at *some* time. However, we just saw that complement phrases, adverbial phrases or temporal anaphors may give a further specification of these indefinite, implicit arguments or impose further conditions on them. The question how to account for this does not have a straightforward answer in static theories.

In static theories of interpretation, an existentially closed argument is not available for further specification. So, if it is stated that Mary is a sister of someone, or if we consider the set of individuals which are a sister of somebody, then, in classical theories, there is hardly any non-ad hoc means of referring back to the individual(s) of which Mary is a sister, or which the previously mentioned set is the set of sisters of. This is where a dynamic semantics comes in. In a dynamic semantics, indefinites objects, that is, objects which are existentially quantified over, *are* available for further specification.

In a dynamic semantics, a sentence like *There is a thief in the house* is assigned an interpretation which allows the indefinite thief to be specified further in subsequent discourse. So, if this sentence is followed by the sentence *He came in through the window*, the resulting interpretation is equivalent with that of *There is a thief in the house who came in through the window*. Clearly, also if a *noun* or *verb* comes with an implicit argument which is existentially quantified over, then, in a dynamic semantics, the argument remains available for further specification or restriction.

Before I show how a dynamic treatment can be given of the specification of implicit arguments, I first give a sketch of some of the basic properties of such a dynamic semantics, *DMG*, which is required for a proper understanding of the sequel. I use a formulation of *DMG* which slightly, but not principally, differs from the one presented by Groenendijk and Stokhof (cf., chapter 1).

Dynamic Montague grammar

Dynamic Montague grammar (*DMG*) is a version of Montague grammar (*MG*, Montague [1973]) which employs the formal apparatus of dynamic intensional logic (*DIL*, cf., Janssen [1986]), a variant of Montague's intensional logic *IL*. In *DMG* intensionality in the ordinary sense of the word is ignored. The apparatus of (dynamic) intensional logic merely serves the purpose of giving a compositional formulation of the dynamics of interpretation.

In *DMG* a proper subset of the types of *DIL* is used, the basic types $\varepsilon = \langle s, e \rangle$ and $\tau = \langle s, \langle\langle s, t\rangle, t\rangle\rangle$, and functional types derived from these two types. Sentences are assigned denotations in the type τ, which are functions from states (assignments) and propositions (sets of assignments) to truth values. Such functions can be conceived of as context change potentials. Typically, the denotation of a sentence S is a function that holds of a state s and a proposition p if, in the terminology of *DRT*, p contains a state (assignment) that verifies S with respect to state (assignment) s.

Like in *MG*, expressions of a fragment of natural language are translated into expressions of some logical language which has a well-defined interpretation. This language is built up from variables and from lifts $\uparrow c$ of extensional *DIL* constants

c. (The lift $\uparrow c$ of such a constant c is of a type obtained from the type of c by replacing all occurrences of e and t in that type by ε and τ, respectively.) Among the lifted constants of type e a set of discourse markers $d, d', \ldots$ is distinguished, whose lift has type ε. The language employs λ-abstraction, application and dynamic counterparts of identity ($\tilde{=}$), negation ($\sim$), existential quantification ($\mathcal{E}d$, where d is a discourse marker) and conjunction (;). I will also use lifted extensional variables and existential quantification over the values of these variables.

Sentences in *DMG* are assigned dynamic translations with dynamic interpretations. The *static* contents of the sentences are given by the closure operation $\downarrow$. In many cases the *static* contents of a *dynamic* translation can be determined by turning the dynamic translation into an ordinary *DIL* expression using the following equivalences ($[\alpha/x]\beta$ is obtained from β by substituting all free occurrences of x in β by α):

DMG reduction

1. (λ-conversion)
   $(\lambda x\ \beta)(\alpha) \quad \Leftrightarrow [\alpha/x]\beta$ (provided all free variables in α are free for x in β)
2. ($\uparrow$-export)
   $(\uparrow\beta)(\alpha) \quad \Leftrightarrow \uparrow(\beta(\downarrow\alpha))$
   $\alpha \mathrel{\tilde{=}} \beta \quad \Leftrightarrow \uparrow(\downarrow\alpha = \downarrow\beta)$
   $\sim\Phi \quad \Leftrightarrow \uparrow\neg\downarrow\Phi$
3. ($\downarrow$-import)
   $\downarrow\uparrow\phi \quad \Leftrightarrow \phi$
   $\downarrow\mathcal{E}d\Phi \quad \Leftrightarrow \exists d\downarrow\Phi$
   $\downarrow[\uparrow\phi\,;\Psi] \quad \Leftrightarrow (\downarrow\uparrow\phi \wedge \downarrow\Psi)$
4. (associativity)
   $[\mathcal{E}d\Phi\,;\Psi] \quad \Leftrightarrow \mathcal{E}d[\Phi\,;\Psi]$
   $[[\Phi\,;\Psi]\,;\Upsilon] \quad \Leftrightarrow [\Phi\,;[\Psi\,;\Upsilon]]$

By means of λ-conversion an expression can be meaning-preservingly reduced under the ordinary conditions of an extensional type theory. The application of the lift of an expression β to a dynamic argument α involves the lift of the application of β to the static content of α. The dynamic equation of α and β involves the equation of the static content of α and β and the dynamic negation of Φ involves the negation of Φ's static content, i.e., of Φ's truth conditions. The import equivalences allow us to replace the other *DMG* operators by their static counterparts. The first associativity equivalence is characteristic for the system of interpretation. The equivalence holds without proviso. So $[\mathcal{E}d\Phi\,;\Psi] \Leftrightarrow \mathcal{E}d[\Phi\,;\Psi]$ also if d occurs free in Ψ. Of course, if t is a variable (not a discourse marker) the equivalence between $[\mathcal{E}t\Phi\,;\Psi]$ and $\mathcal{E}t[\Phi\,;\Psi]$ only holds if t does not occur free in Ψ.

An example

Some basic aspects of *DMG* are illustrated by the *DMG* treatment of the following example:

(1) A man walks in the park. He whistles.

The common noun *man* and the intransitive verbs *walk* and *whistle* translate into the lift of the constants `man`, `walk`, and `whistle`, respectively. These constants are of type $\langle e, t\rangle$, and their lifts are of type $\langle \varepsilon, \tau\rangle$, which is also the type of the variables P and Q. The pronoun he_i is a noun phrase with an interpretation of type $\langle\langle \varepsilon, \tau\rangle, \tau\rangle$. The indefinite article a_i belongs to the category of determiners which is associated with the type $\langle\langle \varepsilon, \tau\rangle, \langle\langle \varepsilon, \tau\rangle, \tau\rangle\rangle$, the category of expressions that make up a noun phrase when combined with a common noun to their right. The combination of a noun phrase with an intransitive verb makes up a sentence. The noun phrases *a man* and *he* are coindexed in order to indicate their anaphoric relationship.

Basic fragment

- a_i $\rightsquigarrow \lambda P \lambda Q\ \mathcal{E} d_i[P(\uparrow d_i)\,;Q(\uparrow d_i)]$
- *man* $\rightsquigarrow \uparrow\mathtt{man}$
- *walks* $\rightsquigarrow \uparrow\mathtt{walk}$
- he_i $\rightsquigarrow \lambda P\ P(\uparrow d_i)$
- *whistles* $\rightsquigarrow \uparrow\mathtt{whistle}$

Three standard applications yield the following translations of the two sentences in example 1:

$$(\lambda P \lambda Q\ \mathcal{E} d_i[P(\uparrow d_i)\,;Q(\uparrow d_i)])(\uparrow\mathtt{man})(\uparrow\mathtt{walk})$$
$$(\lambda P\ P(\uparrow d_i))(\uparrow\mathtt{whistle})$$

These expressions can be reduced in the following way:

$$(\lambda P \lambda Q\ \mathcal{E} d_i[P(\uparrow d_i)\,;Q(\uparrow d_i)])(\uparrow\mathtt{man})(\uparrow\mathtt{walk}) \Leftrightarrow \text{ (λ-conversion)}$$
$$\mathcal{E} d_i[\uparrow\mathtt{man}(\uparrow d_i)\,;\uparrow\mathtt{walk}(\uparrow d_i)] \Leftrightarrow \text{ ($\uparrow$-export)}$$
$$\mathcal{E} d_i[\uparrow(\mathtt{man}(\downarrow\uparrow d_i))\,;\uparrow(\mathtt{walk}(\downarrow\uparrow d_i))] \Leftrightarrow \text{ ($\downarrow$-import)}$$
$$\mathcal{E} d_i[\uparrow(\mathtt{man}(d_i))\,;\uparrow(\mathtt{walk}(d_i))]$$

$$(\lambda P\ P(\uparrow d_i))(\uparrow\mathtt{whistle}) \Leftrightarrow \text{ (λ-conversion)}$$
$$\uparrow\mathtt{whistle}(\uparrow d_i) \Leftrightarrow \text{ ($\uparrow$-export)}$$
$$\uparrow(\mathtt{whistle}(\downarrow\uparrow d_i)) \Leftrightarrow \text{ ($\downarrow$-import)}$$
$$\uparrow(\mathtt{whistle}(d_i))$$

The translation of a sequence of two sentences $S.T$ consists in the dynamic conjunction $[S'\,;T']$ of the translations S' and T' of the two sentences. The (reduced) translation of example 1 therefore reads as follows:

$$[\mathcal{E} d_i[\uparrow(\mathtt{man}(d_i))\,;\uparrow(\mathtt{walk}(d_i))]\,;\uparrow(\mathtt{whistle}(d_i))]$$

By associativity this equals:

$$\mathcal{E}d_i[\uparrow(\mathtt{man}(d_i))\,;[\uparrow(\mathtt{walk}(d_i))\,;\uparrow(\mathtt{whistle}(d_i))]]$$

The truth-conditions of example 1 are given by the closure of the above formula:

$$\downarrow\mathcal{E}d_i[\uparrow(\mathtt{man}(d_i))\,;[\uparrow(\mathtt{walk}(d_i))\,;\uparrow(\mathtt{whistle}(d_i))]]$$

Employing $\downarrow$-import, this equals the following formula:

$$\exists d_i(\mathtt{man}(d_i)\wedge(\mathtt{walk}(d_i)\wedge\mathtt{whistle}(d_i)))$$

The example turns out to mean that there is a man who walks and who whistles. We see that the dynamic semantics accounts for the anaphoric relationship in example 1 which holds between the indefinite noun phrase *a man*, and the coindexed pronoun *he*. Notice that the anaphoric relationship in this example is accounted for by assembling, by means of dynamic conjunction, the meanings of the independently interpreted sentences *A man walks in the park* and *He whistles*. Thus, *DMG* gives a truly compositional formulation of the phenomenon that indefinites in one sentence may bind pronouns in subsequent ones.

This concludes the exposition of *DMG*.

Existential disclosure

In *DMG* the emphasis has been on the potential of indefinite and other noun phrases to bind coindexed pronouns. However, the same phenomena can be described as the potential of pronouns to refer to objects introduced in the context. By attaching a certain index to a pronoun it is made to address a preceding coindexed existential quantifier, if there is any. Clearly, the addressed existential quantifiers need not be introduced by explicit noun phrases, since they also may come along with the translations of (relational) nouns and (eventive) verbs.

It is expedient to elaborate a little more upon the ability to address previously introduced arguments in *DMG*. Suppose that Φ contains an 'active' occurrence of the quantifier $\mathcal{E}d_i$. (An occurrence of $\mathcal{E}d_i$ is active in Φ if it is not followed by another active occurrence of $\mathcal{E}d_i$ in Φ and if it is not in the scope of a negation sign.) Then a *free* variable x can be made to play the role of the bound discourse marker d_i in Φ by conjoining Φ with the formula that asserts the identity of x and d_i. Let us write $\{\uparrow x/d_i\}\Phi$ for such a conjunction $[\Phi\,;(\uparrow x \,\tilde{=}\, \uparrow d_i)]$. The following example shows that in the closure of $\{\uparrow x/d_i\}\Phi$ the quantification over the value of d_i in effect disappears:

$$\downarrow\{\uparrow x/d_i\}\mathcal{E}d_i[\uparrow\mathtt{man}(\uparrow d_i)\,;\uparrow\mathtt{walk}(\uparrow d_i)] =$$
$$\downarrow[\mathcal{E}d_i[\uparrow\mathtt{man}(\uparrow d_i)\,;\uparrow\mathtt{walk}(\uparrow d_i)]\,;(\uparrow x \,\tilde{=}\, \uparrow d_i)] \Leftrightarrow \text{ (associativity)}$$
$$\downarrow\mathcal{E}d_i[\uparrow\mathtt{man}(\uparrow d_i)\,;[\uparrow\mathtt{walk}(\uparrow d_i)\,;(\uparrow x \,\tilde{=}\, \uparrow d_i)]] \Leftrightarrow \text{ ($\uparrow$-export)}$$
$$\downarrow\mathcal{E}d_i[\uparrow(\mathtt{man}(\downarrow\uparrow d_i))\,;[\uparrow(\mathtt{walk}(\downarrow\uparrow d_i))\,;\uparrow(\downarrow\uparrow x \,\tilde{=}\, \downarrow\uparrow d_i)]] \Leftrightarrow \text{ ($\downarrow$-import)}$$
$$\exists d_i(\mathtt{man}(d_i)\wedge(\mathtt{walk}(d_i)\wedge(x=d_i))) \Leftrightarrow \text{ (classically)}$$
$$(\mathtt{man}(x)\wedge\mathtt{walk}(x))$$

Here we see that $\{x/d_i\}\Phi$ involves something like the reverse of an operation of existential closure and I will therefore call it the *existential disclosure* of $\mathcal{E}d_i$ in

Φ. As the above example shows, *DMG* enables us to associate natural language indefinites with free variables, like Lewis [1975] and Heim [1982] do, by translating them using existential quantifiers in the first place.

It may be noticed that the disclosure of $\mathcal{E}d_i$ in Φ not fully dissolves an active quantification over the values of d_i, since if Φ contains an active occurrence of $\mathcal{E}d_i$, this quantifier remains active in $\{\uparrow x/d_i\}\Phi$. So, if one really wants to make such a quantifier vacuous, then one has to 'reset' the value of d_i after the disclosure of d_i in Φ to the value d_i had before the processing of Φ. This can be done as follows:

$$\mathcal{E}t[(\uparrow t \doteq \uparrow d_i)\,;\{x/d_i\}\Phi\,;\mathcal{E}d_i(\uparrow t \doteq \uparrow d_i)]$$

provided t does not occur free in Φ.

In what follows, the possibility to address values existentially quantified over in *DMG*, and, more in particular, existential disclosure, will be employed in a compositional treatment of the specification of implicit arguments. Adnominal and adverbial modifiers will be taken to address arguments which are implicitly added to the context by the nouns and verbs on which they operate. These arguments of nouns and verbs make their appearance in the translation of these nouns and verbs as supplementary arguments which are (dynamic) existentially closed and they thus remain available for further specification. It may be noticed that this way of storing information about implicit arguments is comparable to the way in which such information is encoded in Bartsch' [1987] lexical representations of nouns and verbs. Bartsch presents a more comprising treatment of similar phenomena, but in a less compositional way.

Of course, in order to guarantee proper results we have to make sure that adverbial and adnominal modifiers address the *right* arguments. For this purpose I will employ two slightly different methods. The first method is based on the assumption that certain adnominal (and adverbial) phrases tend to address implicit arguments with specific thematic roles or cases. A specific use of the preposition *of* addresses what may be called the implicit object of relational nouns, as in *sister of John*, or *captain of the ship*. Similarly, *by*-phrases appear to select implicit subjects, as in *the destruction of the city by the enemy*. In order to indicate the kinds of arguments that are addressed by these phrases I will use distinguished discourse markers which each 'label' specific implicit thematic roles and which are referred to by the respective ad-phrases. This way of establishing the connection between complement phrases and addressed arguments is illustrated by the treatment of relational nouns below.

The second way of achieving the same aim consists in an extension of the indexing procedure. For instance, (certain) verbs are assumed to come with argument slots for events which are filled by arbitrary indexed discourse markers which range over events. These events then are available for further specification by coindexed adverbial phrases.

It must be clear that we also have to make sure that the ad-phrases eventually *do* address implicit arguments. For this purpose I assume a meta-rule that forbids vacuous addressing, which prevents adnominals to combine with nouns which lack the addressed argument and which makes sure that the indices on verbs and adverbs coincide. Such a prohibition can be conceived of as a dynamic analogue of Kratzer's [1988] prohibition against vacuous quantification by means of adverbs of quantification. Probably, such a rule can be implemented in a more compositional, semantic fashion, but I will not attempt to do so here. The required restatement of such a rule would be preferrably cast within a more general theory of indexing and pronoun resolution, and the issue, therefore, is left for further research.

A final remark concerns possible anaphoric relationships with implicit arguments. Although the noun *captain*, in certain contexts at least, comes with implicit ships which the captains are captains of, we cannot refer back to such ships with a pronoun. So, the pronoun *it* in *A captain whistles. It is ready to leave* seems to be not, or hardly, acceptable. On the other hand, if a sentence describes an event, the event can be referred back to, as in *Mary hit John. He laughed about it*. For this reason, we will simply assume that discourse marker that are used to label thematic cases, unlike those which refer to events, are not used in the translations of pronouns.

2 Relational nouns

Relational nouns, like *mother* and *captain*, are nouns that have been considered 'unsaturated' semantically. Every mother is the mother of someone, and every captain is the captain of some ship (in the default case). However, as is argued in de Bruyn and Scha [1988], this does not imply that relational nouns should be treated syntactically as being subcategorized for certain prepositional phrases. De Bruyn and Scha point out that the syntactic behaviour of relational nouns is very similar to that of 'ordinary' nouns, and they claim that the "overt realization of the arguments of a 'transitive' noun is always optional" (p. 26). Therefore, they propose a treatment of relational nouns which is on a par with that of ordinary nouns in the syntax, and which accounts for the idiosyncratic properties of relational nouns in the semantics. In this section, I subscribe to these general features of de Bruyn and Scha's proposal and give a more uniform elaboration of them in the format of *DMG*.

In de Bruyn and Scha's proposal the complement phrases that combine with relational nouns are considered to be modifiers of the nouns syntactically. Semantically, these complements address arguments of the relations expressed. The semantic part of de Bruyn and Scha's proposal comes down to the following. Relational nouns denote genuine relations, that is, sets of pairs of objects. The relational noun *sisters*, for instance, denotes the set of pairs each pair in which consists of an individual and a sister of that individual. Expressions like *Peter's* in *Peter's sisters*, or *of Peter* in *sisters of Peter*, restrict such sets of pairs to the sets consisting of pairs of which the first element is Peter. Notice that these phrases still denote sets of pairs of individuals. So, in order to let these expressions combine properly with verbs and other expressions, a meaning postulate is invoked that links up the semantic role of a set of pairs of objects with a corresponding set of objects (a projection of the set of pairs). In the example at hand, the set of all pairs consisting of Peter and a sister of his is linked up with the set of individuals which are Peter's sisters.

So, according to de Bruyn and Scha's proposal, nouns may denote sets of individuals as well as relations between individuals, and verbal predicates are made to apply both to plural noun phrases which denote sets of individuals, as well as to relational noun phrases which denote relations between individuals. For the purpose of giving an account of relational nouns, such complications can be avoided within a dynamic framework. Restricting ourselves to (singular) relational nouns, we can preserve both a uniform syntax, like de Bruyn and Scha, and a uniform, compositional semantics.

Sketch of a dynamic treatment of relational nouns

In *DMG* all nouns can be assigned the type of dynamic properties (type $\langle\varepsilon,\tau\rangle$), the type of functions from individual concepts to context change potentials. Relational nouns have interpretations in this type which contribute implicit arguments to the context. Complement phrases simply address the contributed arguments and give it a further specification. So, for instance, the relational noun *sister* can be interpreted as the property of being the sister of somebody and the syntactic modifier *of John* may turn this property into the property of being the sister of somebody who is John, that is, into the property of being a sister of John. Clearly, since both the nouns *sister* and *sister of John* denote dynamic properties, they may immediately combine with a determiner to form a noun phrase without assuming type-polymorphism or without the need of applying some operation of existential closure. So, within a dynamic semantics like *DMG* we can treat the prepositional phrase *of John* as an ordinary noun modifier syntactically, whereas it behaves semantically as a specifier of an implicit argument of the noun.

A dynamic treatment of relational nouns can be developed further along the following lines. The treatment of the noun *captain* exemplifies that of relational nouns in general. The expressions *ship* and *captain* belong to the category of common nouns (*CN*), associated with the type $\langle\varepsilon,\tau\rangle$ of dynamic properties. The name *the SS. Enterprise* belongs to the category of noun phrases (*NP*). We use two dynamic determiners, a_i and $every_j$. Finally, we have a complement preposition of_2 which belongs to the category *(CN\CN)/NP*, the category of expressions that, when combined with a noun phrase to their right belong to the category of *CN*-modifiers.

The translations of these expressions are the following (again, P and Q are variables of type $\langle\varepsilon,\tau\rangle$, x and y are variables of type ε, and T is a variable of type $\langle\langle\varepsilon,\tau\rangle,\tau\rangle$; as usual, $\mathcal{A}d\Phi$ is defined as $\sim\mathcal{E}d\sim\Phi$, and $[\Phi \Rightarrow \Psi]$ is defined as $\sim[\Phi\,;\sim\Psi]$):

Translations in a fragment with relational nouns

- a_i $\rightsquigarrow \lambda P\lambda Q\ \mathcal{E}d_i[P(\uparrow d_i)\,;Q(\uparrow d_i)]$
- *every*$_j$ $\rightsquigarrow \lambda P\lambda Q\ \mathcal{A}d_j[P(\uparrow d_j) \Rightarrow Q(\uparrow d_j)]$
- *ship* $\rightsquigarrow \uparrow\mathtt{ship}$
- *captain* $\rightsquigarrow \lambda x\ \mathcal{E}d_2\uparrow\mathtt{captain\_of}(\uparrow d_2)(x)$
- *of*$_2$ $\rightsquigarrow \lambda T\lambda P\lambda x\ \uparrow\downarrow T(\lambda y\ \{y/d_2\}P(x))$
- *the SS. Enterprise* $\rightsquigarrow \lambda Q\ Q(\uparrow\mathtt{ss.e})$

This tiny fragment generates sentences like *Every captain whistles*, *A captain of*$_2$ *the SS. Enterprise whistles* and *Every captain of*$_2$ *a ship whistles*. The (reduced) translations of these expressions are, respectively:

(2) Every captain whistles
$\mathcal{A}d_j[\mathcal{E}d_2\uparrow(\mathtt{captain\_of}(d_2)(d_j)) \Rightarrow \uparrow(\mathtt{whistle}(d_j))]$

(3) A captain of$_2$ the SS. Enterprise whistles
$\mathcal{E}d_i[\uparrow(\mathtt{captain\_of}(\mathtt{ss.e})(d_i))\,;\uparrow(\mathtt{whistle}(d_i))]$

(4) Every captain of$_2$ a ship whistles
$\mathcal{A}d_j[\uparrow\exists d_i(\mathtt{ship}(d_i) \wedge \mathtt{captain\_of}(d_i)(d_j)) \Rightarrow \uparrow(\mathtt{whistle}(d_j))]$

I now show in some more detail how example 3 is dealt with.

The translation of the expression *of*$_2$ *the SS. Enterprise* consists in the application of the translation of *of*$_2$ to that of *the SS. Enterprise*. By means of λ-conversion this translation can be reduced as follows:

$$(\lambda T\lambda P\lambda x\ \uparrow\downarrow T(\lambda y\ \{y/d_2\}P(x)))(\lambda Q\ Q(\uparrow\mathtt{ss.e})) \Leftrightarrow$$
$$\lambda P\lambda x\ \uparrow\downarrow(\lambda Q\ Q(\uparrow\mathtt{ss.e}))(\lambda y\ \{y/d_2\}P(x)) \Leftrightarrow$$
$$\lambda P\lambda x\ \uparrow\downarrow(\lambda y\ \{y/d_2\}P(x))(\uparrow\mathtt{ss.e}) \Leftrightarrow$$
$$\lambda P\lambda x\ \uparrow\downarrow\{\uparrow\mathtt{ss.e}/d_2\}P(x)$$

If we apply this expression to the translation of *captain* we get the translation of the complex common noun *captain of*$_2$ *the SS. Enterprise*, which, again by means of λ-conversion, can be reduced in the following way:

$$(\lambda P\lambda x\ \uparrow\downarrow\{\uparrow\mathtt{ss.e}/d_2\}P(x))(\lambda x\ \mathcal{E}d_2\uparrow\mathtt{captain\_of}(\uparrow d_2)(x)) \Leftrightarrow$$
$$\lambda x\ \uparrow\downarrow\{\uparrow\mathtt{ss.e}/d_2\}(\lambda x\ \mathcal{E}d_2\uparrow\mathtt{captain\_of}(\uparrow d_2)(x))(x) \Leftrightarrow$$
$$\lambda x\ \uparrow\downarrow\{\uparrow\mathtt{ss.e}/d_2\}\mathcal{E}d_2\uparrow\mathtt{captain\_of}(\uparrow d_2)(x)$$

By the definition of existential disclosure, this expression is equivalent with the following expression, and with the subsequent reductions of it:

$$\lambda x\ \uparrow\downarrow[\mathcal{E}d_2\uparrow\mathtt{captain\_of}(\uparrow d_2)(x)\,;\uparrow\mathtt{ss.e} \mathrel{\tilde{=}} \uparrow d_2] \Leftrightarrow \text{(associativity)}$$
$$\lambda x\ \uparrow\downarrow\mathcal{E}d_2[\uparrow\mathtt{captain\_of}(\uparrow d_2)(x)\,;\uparrow\mathtt{ss.e} \mathrel{\tilde{=}} \uparrow d_2] \Leftrightarrow (\uparrow\text{-export})$$
$$\lambda x\ \uparrow\downarrow\mathcal{E}d_2[\uparrow(\mathtt{captain\_of}(d_2)(\downarrow x))\,;\uparrow(\downarrow\uparrow\mathtt{ss.e} = \downarrow\uparrow d_2)] \Leftrightarrow (\downarrow\text{-import})$$

$$\lambda x \uparrow\exists d_2(\texttt{captain\_of}(d_2)(\downarrow x) \wedge \texttt{ss.e} = d_2) \Leftrightarrow$$
$$\lambda x \uparrow(\texttt{captain\_of}(\texttt{ss.e})(\downarrow x))$$

The sentence *A captain of*$_2$ *the SS. Enterprise whistles* has a (reduced) translation which consists in the application of the translation of *a* to the translations of *captain of*$_2$ *the SS. Enterprise* and *whistles*, respectively. The translation of the sentence can be reduced in the following way, again by means of λ-conversion:

$$(\lambda P \lambda Q\ \mathcal{E}d_i[P(\uparrow d_i)\,;Q(\uparrow d_i)])(\lambda x \uparrow(\texttt{captain\_of}(\texttt{ss.e})(\downarrow x)))(\uparrow\texttt{whistle}) \Leftrightarrow$$
$$\mathcal{E}d_i[(\lambda x \uparrow(\texttt{captain\_of}(\texttt{ss.e})(\downarrow x)))(\uparrow d_i)\,; \uparrow\texttt{whistle}(\uparrow d_i)] \Leftrightarrow$$
$$\mathcal{E}d_i[\uparrow(\texttt{captain\_of}(\texttt{ss.e})(\downarrow\uparrow d_i))\,; \uparrow\texttt{whistle}(\uparrow d_i)]$$

Using $\uparrow$-export and $\downarrow$-import, the translation of example 3 turns out to be equivalent with the following expression:

$$\mathcal{E}d_i[\uparrow(\texttt{captain\_of}(\texttt{ss.e})(d_i))\,; \uparrow(\texttt{whistle}(d_i))]$$

This formula is true iff there is an object z such that the pair of z and the denotation of `ss.e` are in the extension of `captain_of` and z is in the extension of `whistle`. Furthermore, under this dyanmic translation the captain who whistles can be referred back to by a pronoun as in a continuation of the example with the sentence *He is glad to leave*.

The rudimentary fragment above shows how determiners may be combined with ordinary common nouns, relational nouns and relational nouns with complement phrases, this in a uniform fashion. The treatment of the adnominals crucially relies on the possibility to address objects which are implicitly introduced by relational nouns. As was argued above, however, it must have been made sure that these phrases do not fail to address arguments, this, by means of the meta-rule which prohibits vacuous addressing. A complement phrase like *of*$_2$ *John* is only allowed to apply to nouns that come with implicit object arguments, that is, to nouns whose translations contain an active occurrence of the quantifier $\mathcal{E}d_2$. This prohibition against vacuous addressing expels anomalous constructions like, for instance, *a ship about syntax*.

Notice that the above translation of the preposition *of*$_2$ involves an operation of static closure by means of $\uparrow\downarrow$. This closure operation annihilates the dynamic potential of implicit arguments and indefinite noun phrases which occur in constructions of the form *CN of*$_2$ *NP*. There are two reasons for choosing to adopt this closure. In the first place, an indefinite noun phrase *a man* in a construction like *sister of a man* does not seem to be eligible as an antecedent for subsequent pronominal coreference. In the second place, if an implicit argument of a relational noun has been specified by an adnominal *of*$_2$-phrase, the argument does not seem to be available for further adnominal specification either. So, since in the translation of the complex common noun *brother of*$_2$ *Mary* no quantifiers are active, the rule against vacuous addressing excludes the construction of a complex noun *brother of*$_2$

John of$_2$ *Mary*. It must be noticed, however, that the translation of functionally used prepositions with an operation of static closure may be too restrictive to account for noun phrases such as *the destruction of*$_2$ *the city by the enemy* and *a letter from John to Harry about this issue*. Suitable adaptations have to be made in order to deal with these examples, but going into the required amendations would lead us too far from the (programmatic) purposes of this chapter.

3 Non-temporal adverbial phrases

The modification of verbs by adverbial phrases (among which I include prepositional phrases for simplicity) is in some respects similar to the modification of nouns. In contrast with direct and indirect object phrases, for which verbs may be subcategorized, adverbial phrases behave syntactically like verb (phrase) modifiers. (I disregard sentential adverbs, or 'ad-sententials', here.) However, many adverbial phrases are not arbitrary modifiers semantically, but they appear to have a genuine contribution to the information expressed by the verbs they modify.

Consider the following examples.

(5) Harry walks from Amsterdam to Budapest.
(6) Mary hits John with a hammer on his head.

In Montague [1970a] phrases like *from Amsterdam* and *with a hammer* express functions from properties to properties. For instance, sentence 5 gets translated, roughly, as `to(b)(from(a)(walk))(h)`. Such an approach, however, does not validate two inference schemes discussed in Parsons [1989], which seem to give a correct characterization of the logical behaviour of many adverbial phrases:

Scope entailment

$$ADV_1(ADV_2(\phi)) \rightarrow ADV_2(ADV_1(\phi))$$

Diamond entailment

$$\begin{array}{ccccc} & \nearrow & ADV_1(\phi) & \searrow & \\ ADV_1(ADV_2(\phi)) & & & & \phi \\ & \searrow & ADV_2(\phi) & \nearrow & \end{array}$$

Adverbial phrases for which the scope entailment holds can be reordered in a sentence without changing the sentence's meaning. For instance, *Harry walks to Budapest from Amsterdam* and *Mary hits John on the head with a hammer* have the same truth conditions as the examples 5 and 6 respectively. (That is, disregarding other aspects of meaning, such as, for instance, that of topic/focus.) Likewise, the deletion of adverbial phrases which respect to the diamond entailment weakens a sentence's truth conditions. So, example 5 entails that Harry walked from Amsterdam, that Harry walked to Budapest, and that Harry walked. (At first glance one might think that all adverbial phrases respect these entailment schemes, but this is not true. Montague [1970a] offers the example *in a dream*.)

Montague himself has observed this inferential behaviour of many ad-phrases, and he proposes to account for it by assigning such phrases the 'intersection-property' ([1970a, pp. 211–213]). This does not seem to be completely satisfactory. On the intersective interpretation of an adverbial phrase, the phrase expresses a property of the subject of the verb modified. But if the adverbial phrases in the examples above express properties, then it can hardly be properties of the subject of the sentences. For instance, if the intersection property were to hold for the adverbials in 6, then 6 together with the premiss that Mary sings would allow us to conclude that Mary sings with a hammer on John's head. This, of course, should not be.

Alternative analyses of adverbial modification have been given by, among others, Davidson [1967], Bartsch [1972], Parsons [1989, 1991] and Dowty [1989]. In these analyses (certain) verbs are taken to express properties of events (states, processes, space/time regions, ...), or to express relations between events (...) and some number of other arguments. Adverbial phrases are taken to be verb phrase modifiers which impose further restrictions on the events described by the verbs phrases to which they apply.

For instance, the verb phrase *hits John* is interpreted as the relation between individuals and events that holds of an individual x and an event e iff e is an event of x's hitting John. An adverbial phrase like *with a hammer* is analyzed as a verb modifier that, roughly, requires the event associated with the verb to which it applies to be with a hammer, or to be performed with a hammer. Similarly, the adverbial phrase *on the head* is taken to impose the condition that the event of hitting is a hitting on the head. Passing over details, the sentence *Mary hits John with a hammer on the head* is eventually associated with the following translation:

$$\exists e(\texttt{hit}(e)(\texttt{j})(\texttt{m}) \wedge \texttt{with\_a\_hammer}(e) \wedge \texttt{on\_the\_head}(e))$$

which states that there is an event of Mary's hitting John such that the hitting is performed with a hammer and such that the hitting is on the head (of John).

It is easily seen that if adverbial phrases impose conditions on events in the way sketched then they validate the aforementioned diamond and scope entailments. These entailments correspond to conjunction reduction and conjunction commutation respectively.

Clearly, matters are simplified here and the authors mentioned above have made many refinements, of course. To mention two, the condition that the event e is 'with a hammer' has been further analyzed as the condition that the 'instrument' relation holds between e and a hammer. Furthermore, the condition that e is an event of Mary's hitting John has been analyzed as the condition that e is a hitting event, and that the relation 'being the agent of' holds of Mary and e and the relation 'being the direct object of' holds of John and e. Complications like these, and simplifications like mine, however, are not relevant for the present discussion.

It may be noticed that in these Davidsonian treatments of adverbial modification an operation of existential closure has to apply at some stage in the process of interpretation. In order for the events described by verbs phrases to be available for adverbial modification they are indicated by free variables, the values of which may be abstracted over in the translations of the verb phrases. However, a finite sentence is generally taken to state the existence of an event of the kind described by the sentence. For this reason an operation of existential closure has to apply somewhere in the process of interpretation. In the proposals at issue, which are cast within static frameworks, such a closure has drawbacks. The closure precludes the possibility of anaphoric reference to described events, despite the fact that such anaphors to events make up one of the arguments for positing the existence of events in the first place.

Sketch of a dynamic treatment of adverbial modification

Analogous to the preceding treatment of relational nouns and their complements, the treatments of adverbial modification can be recast within the framework of *DMG*, without the need to postulate an additional operation of existential closure. As we have seen, such intermediary closures can be dispensed with in a dynamic semantics, since the arguments involved can already be assumed to be existentially closed in the translation of the (finite) verbs themselves. Due to the dynamics of the system of interpretation, these arguments can still be addressed by adverbial phrases.

We can adopt the analyses of adverbs sketched above within *DMG*, roughly, in the following way. A finite form of the verb *walk* belongs to the category IV. The prepositions *from* and *to* belong to the category *(IV\IV)/NP*, the category of expressions that form an intransitive verb phrase when combined with a noun phrase to their right and, next, with an intransitive verb phrase to their left. The names *Harry*, *Amsterdam* and *Budapest* belong to the category of noun phrases. The model is assumed to be extended with some appropriate domain of events, and the translation language is extended with discourse markers $e_i, e_j \ldots$ which range over events. The lexical expressions involved can be translated in the following way (variables are typed as above; the):

Translations in a fragment with adverbial phrases

- *Harry* $\rightsquigarrow \lambda Q\ Q(\uparrow\mathtt{h})$
- *walks$_i$* $\rightsquigarrow \lambda x\ \mathcal{E}e_i{\uparrow}\mathtt{walk}(\uparrow e_i)(x)$
- *from$_i$* $\rightsquigarrow \lambda T\lambda P\lambda x\ T(\lambda y\ [P(x)\ ;\ {\uparrow}\mathtt{from}(y)(\uparrow e_i)])$
- *Amsterdam* $\rightsquigarrow \lambda Q\ Q(\uparrow\mathtt{a})$

- *to_i* $\rightsquigarrow \lambda T \lambda P \lambda x\ T(\lambda y\ [P(x)\,;\uparrow\mathtt{to}(y)(\uparrow e_i)])$
- *Budapest* $\rightsquigarrow \lambda Q\ Q(\uparrow\mathtt{b})$

The constant `walk` is interpreted as a relation between individuals and events, to be understood as the relation which holds of an object x and an event e iff e is an event of x's walking. In the translation of the expression *walks*, the event argument slot of `walk` is closed by a dynamic existential quantifier. A sentence like *Harry walks* thus introduces an event of Harry's walking which is available for subsequent anaphoric reference and adverbial modification.

An adverbial phrase like *from Amsterdam* addresses the event introduced by the verb to which the phrase applies. Recall that we have assumed a rule prohibiting vacuous addressing. So, with regards to the verb phrase *walks from Amsterdam*, the verb *walks* and the preposition *from* are required to be coindexed. The verb phrase therefore denotes the dynamic property which can be paraphrased as the property of being an object that figures in an event of walking which starts from Amsterdam (A' indicates the translation of A):

$$\begin{aligned}
&\mathit{from_i}'(\mathit{Amsterdam}')(\mathit{walks_i}') = \\
&\lambda T\lambda P\lambda x\ T(\lambda y\ [P(x)\,;\uparrow\mathtt{from}(y)(\uparrow e_i)])(\lambda Q\ Q(\uparrow\mathtt{a}))(\lambda x\ \mathcal{E}e_i\uparrow\mathtt{walk}(\uparrow e_i)(x)) \Leftrightarrow \\
&\lambda x\ (\lambda Q\ Q(\uparrow\mathtt{a}))(\lambda y\ [(\lambda x\ \mathcal{E}e_i\uparrow\mathtt{walk}(\uparrow e_i)(x))(x)\,;\uparrow\mathtt{from}(y)(\uparrow e_i)]) \Leftrightarrow \\
&\lambda x\ (\lambda y\ [\mathcal{E}e_i\uparrow\mathtt{walk}(\uparrow e_i)(x)\,;\uparrow\mathtt{from}(y)(\uparrow e_i)])(\uparrow\mathtt{a}) \Leftrightarrow \\
&\lambda x\ [\mathcal{E}e_i\uparrow\mathtt{walk}(\uparrow e_i)(x)\,;\uparrow\mathtt{from}(\uparrow\mathtt{a})(\uparrow e_i)] \Leftrightarrow \\
&\lambda x\ \mathcal{E}e_i[\uparrow\mathtt{walk}(\uparrow e_i)(x)\,;\uparrow\mathtt{from}(\uparrow\mathtt{a})(\uparrow e_i)]
\end{aligned}$$

The translation of the sentence *Harry $walks_i$ $from_i$ Amsterdam* is obtained by applying the translation of *Harry* to that of *$walks_i$ $from_i$ Amsterdam*, the result of which reduces as follows:

$$\begin{aligned}
&(\lambda Q\ Q(\uparrow\mathtt{h}))(\lambda x\ \mathcal{E}e_i[\uparrow\mathtt{walk}(\uparrow e_i)(x)\,;\uparrow\mathtt{from}(\uparrow\mathtt{a})(\uparrow e_i)]) \Leftrightarrow \\
&\mathcal{E}e_i[\uparrow\mathtt{walk}(\uparrow e_i)(\uparrow\mathtt{h})\,;\uparrow\mathtt{from}(\uparrow\mathtt{a})(\uparrow e_i)] \Leftrightarrow \\
&\mathcal{E}e_i[\uparrow(\mathtt{walk}(e_i)(\mathtt{h}))\,;\uparrow(\mathtt{from}(\mathtt{a})(e_i))]
\end{aligned}$$

The sentence turns out to mean that there is an event of Harry's walking which is from Amsterdam.

The event introduced by the verb *$walk_i$* remains available for further specification or modification also after the adverbial phrase *$from_i$ Amsterdam* has applied to it. Hence, a second adverbial modifier *to_i Budapest*, with the same index again, may apply to the phrase *$walks_i$ $from_i$ Amsterdam*, and a sentence like *Harry $walks_i$ $from_i$ Amsterdam to_i Budapest* turns out to mean that there is an event of Harry's walking which is from Amsterdam and which is to Budapest:

$$\begin{aligned}
&[[\mathcal{E}e_i\uparrow(\mathtt{walk}(e_i)(\mathtt{h}))\,;\uparrow(\mathtt{from}(\mathtt{a})(e_i))]\,;\uparrow(\mathtt{to}(\mathtt{b})(e_i))] \Leftrightarrow \\
&\mathcal{E}e_i[\uparrow(\mathtt{walk}(e_i)(\mathtt{h}))\,;[\uparrow(\mathtt{from}(\mathtt{a})(e_i))\,;\uparrow(\mathtt{to}(\mathtt{b})(e_i))]]
\end{aligned}$$

Clearly, the sentences are assigned the truth-conditions argued for, this in a uniform way. The present exposition may serve as an indication of how to give a straighforward reformulation in *DMG* of the analyses of adverbial modification referred to above, whatever their details are.

McConnell-Ginet [1982] and Larson [1988] have presented alternative analyses in which adverbial phrases express properties of objects that are derived from the verb interpretation in some way or other. For instance, according to the proposal of McConnell-Ginet the interpretation of the verb *walks* (a property) can be turned into a two place relation which holds between walkers and walking rates. An adverb like *quickly* can be applied to such an inflated interpretation of the verb, and impose the condition that the 'added' rate is quick. In Larson's proposal, which is cast within a situation theoretic framework, a walking event is taken to 'involve' an event in which an agent changes position. An adverbial like *to Budapest* then imposes the condition that Budapest is the goal of the change of location which is involved by the situation described by verb.

Again passing over all kinds of details, I think the essence of the last mentioned treatments of adverbials can be wormed from their theoretical frameworks and cast in the mould of a dynamic semantic framework. McConnell-Ginet says: "Ad-verbs typically have a dual function: they augment the order of the verb on which they operate, and they specify the value(s) of the added argument place(s)" (p. 168). In the format of *DMG* we can restate this function of adverbial phrases as their potential to disclose (and specify) implicit arguments in the verbs on which they operate. So, where McConnell-Ginet would have an augmentation of a verb with an added rate (manner, ...) argument place, we could propose an initial rate (manner, ...) argument place that is existentially closed in the translation of the verb. The essence of Larson's proposal, as far as the semantics is concerned, may be clarified if we take constraints to be meaning postulates (disregarding the fact that they were *not* intended to be taken that way), and introduce the involved situations or involved objects in the lexical translations of the involving verbs themselves.

It must be remarked, however, that it is not a trivial matter to elaborate a comprehensive treatment of adverbial phrases along these lines. The facts at issue are much more entangling then appears from the present discussion, of course. Bartsch [1972], in particular, presents a large number of observations which a full-blown theory of adverbs should account for, about the kinds of objects which are available for adverbial modification, and about the particular properties of all different kinds of adverbial modification. The ways in which one extends the present treatments of adverbial modification will probably depend on the ontological commitments one wishes to make and on what one considers to be an economical way to account for the semantic facts involved. For the present aims, however, it is immaterial how one should want to proceed. What I merely aimed to demonstrate in this section is that the treatments of adverbial modification discussed here can be perspicuously formulated within the framework of a dynamic system of interpretation.

4 Tense in discourse

The last application of existential disclosure concerns temporal adverbs and tense in discourse. Since the syntactic category of (temporal) adverbs is most likely that of verb (phrase) modifiers, one again might be tempted to treat them also as modifiers at the semantic level. As regards tense one might think of a Priorian analysis of temporal operators as operators that quantify over times of evaluation. We can use the operator P of classical tense logic to translate verbs in the past tense. A past-tensed sentence is then translated as $P(\phi)$, where $P(\phi)$ is true at some time of evaluation iff ϕ is true at some earlier time of evaluation. Likewise, the adverb *yesterday* can be translated as the operator Y which shifts back the time of evaluation to the day before the original evaluation time. Dowty [1982] (among others) has argued that this is not a tenable analysis.

Consider the following example:

(7) Bill left yesterday.

On a Priorian analysis this sentence translates either as $Y(P(\texttt{leave}(\texttt{b})))$ or as $P(Y(\texttt{leave}(\texttt{b})))$. Neither translation gives the right truth conditions, since both are verified if Bill did not leave yesterday, but, for instance, two days ago.

Reference time, speech time and event time

Alternative approaches to tense have been offered by Bäuerle [1979], Dowty [1982], [1986], Kamp and Rohrer [1983], Hinrichs [1986] and Partee [1984], to name just a few. Common to all of these approaches is that they elaborate upon Reichenbach's distinction between speech time, reference time and event time. This distinction can be illustrated by the sentence *Bill had gone*. This sentence describes a state of affairs at a reference time before the speech time such that at the reference time it holds that Bill left at an event time before that (cf., Reichenbach [1947, pp. 287–298]).

In all the approaches mentioned, tenses express conditions on the ordering of the three points or intervals of time, and temporal adverbs impose conditions on the reference time. For instance, a simple past (future) indicates that the reference time lies before (after) the speech time and the adverb *yesterday* imposes the condition that the reference time is located the day before the speech time. The past perfect furthermore locates the (current) reference time in the past, and assigns the tensed verb a reference or event time before the current reference time.

Reference times also play an important role in the temporal connectedness of 'temporal', or 'narrative', discourse. It is commonly assumed that narrative discourses describe successions of events. A restatement of this is that the reference time in such a discourse moves forward sentence by sentence. (We can take this to be the default case. There are quite a lot of adjustments to be made here, but these need not interfere with the present discussion.) Dowty [1986] formulates it as follows:

"The reference time of a sentence is interpreted to be a time consistent with the definite time adverbials, if there are any; otherwise, a time which immediately follows the reference time of the previous sentence." This is what he calls the 'temporal discourse interpretation principle (TDIP)'.

In the proposal of Kamp and Rohrer, Hinrichs and Partee, which are cast within the framework of *DRT*, something like the TDIP is formulated in the discourse representation construction algorithm. The picture is a bit more involved here, since the proposals make a distinction between eventive and stative sentences. Eventive sentences are taken to describe events which occur at a time contained in the current reference time. These sentences are supposed to move the reference time forward within a discourse. Stative sentences, on the other hand, describe states which obtain at a stretch of time which itself contains the reference time. These sentences are assumed to keep the reference time the same.

There is one difference between the analyses of the last mentioned authors and the one by Dowty which is worth mentioning. In the *DRT* proposals the reference time is eventually existentially quantified over. In Dowty's proposal, on the other hand, reference times are definite parameters of interpretation. The adoption of a definite reference time is, among others, motivated by Partee's example *I didn't turn of the stove*. The idea is that if reference times are existentially quantified over, then the example would mean either that there is a time in the past where the speaker did not turn of the stove (and be trivially true under common circumstances), or that there is no time in the past where she turned of the stove (and be trivially false then). On Dowty's account the example will be probably taken to mean that the speaker did not turn off the stove at reference time. It appears to me that something like Dowty's interpretation can be also be attained by means of restricted existential quantification over reference times. If the reference time, which is existentially quantified over, is required to be found within a restricted (temporal) domain of quantification, then both analyses may turn out equivalent if the domain chosen in the existential analysis coincides with the choice of the reference time in Dowty's analysis.

In the approaches mentioned, the interpretation of tensed verbs is in some way or other related to (event) times. We need not go into the required ontology of events and times here, but simply assume some suitable domain of events which comes with a temporal order. Verbs can be taken to have argument slots for events with associated event times, and temporal adverbs and tense operators are taken to impose conditions on the times of the events that fill these slots, for instance, by relating them in certain specific ways to the speech and/or reference time.

Again, the semantic contribution of the respective verbs, temporal adverbs and temporal operators can be perspicuously formulated within the framework of *DMG*. In what follows I will partly reformulate the proposals of Partee, which are by and large based upon the work of Hinrichs. I will pass over the details of the proposals, and we will solely be concerned with stretches of simple, linear narrative discourse.

Sketch of a dynamic treatment of tense

In the *DMG* treatment of tense and temporal predication it is expedient to have at our disposal three distinguished discourse markers d_s, d_a and d_b. The first one of these (d_s) is used to indicate the speech time. The two other discourse markers (d_a and d_b) can be used to refer to the current reference time, and, if necessary, to reset it. Initially, the reference time is existentially quantified over.

In the translation of a sentence with an eventive verb the current reference time labeled by d_a is pushed forward by requiring it to precede the next reference time labeled by d_b. As we will see presently, the value of d_b will figure as the reference time for a subsequent sentence, since in the conjunction of two sentences S and T in a temporal discourse the (input) reference time for T (i.e., the value of the occurrence of d_a in T) is equated with the (output) reference time delivered by S (i.e., the value of the occurrence of d_b in S). The translation of a stative verb leaves the reference time untouched by equating the 'input' reference time with the 'output' reference time.

In the following fragment the translation of the intransitive verb *arrive* illustrates the interpretation of eventive verbs, and the translation of *sleep* that of stative verbs. The past tense affix *–ed* is defined as an operator that turns an untensed intransitive verb into a tensed intransitive verb. I have added a temporal sentential connective *when* and a sequencing operator '$._t$' which is used in temporal or narrative discourse. (Think of $._t$ as the child's connective *And then* ....)

Translations in a fragment with temporal operators

- *arrive*$_i$ $\rightsquigarrow \lambda x\ \mathcal{E}d_a\mathcal{E}d_b\mathcal{E}e_i\ [(\uparrow e_i \mathrel{\tilde{\subseteq}} \uparrow d_a)\ ;\ [\uparrow\mathtt{arrive}(\uparrow e_i)(x)\ ;\ (\uparrow e_i \mathrel{\tilde{<}} \uparrow d_b)]]$
- *sleep*$_i$ $\rightsquigarrow \lambda x\ \mathcal{E}d_a\mathcal{E}d_b\mathcal{E}e_i\ [(\uparrow d_a \mathrel{\tilde{\subseteq}} \uparrow e_i)\ ;\ [\uparrow\mathtt{sleep}(\uparrow e_i)(x)\ ;\ (\uparrow d_a \mathrel{\tilde{=}} \uparrow d_b)]]$
- *-ed* $\rightsquigarrow \lambda P\lambda x\ [P(x)\ ;\ (\uparrow d_a \mathrel{\tilde{<}} \uparrow d_s)]$
- $S._t T$ $\rightsquigarrow \mathcal{E}t\ [\{\uparrow t/d_b\}S'\ ;\ \{\uparrow t/d_a\}T']$
- *when* S, $T \rightsquigarrow \mathcal{E}t\ [\{\uparrow t/d_a\}S'\ ;\ \{\uparrow t/d_a\}T']$

The temporal relation expressions $\tilde{\subseteq}$ and $\tilde{<}$ are the *DMG* counterparts of *DIL* expressions $\subseteq$ and $<$ which express the relation of being temporally contained in and that of temporal precedence, respectively. Formulas containing these operators observe the $\uparrow$-export equivalences which also apply to identity statements:

$\alpha \mathrel{\tilde{\subseteq}} \beta \Leftrightarrow \uparrow(\downarrow\alpha \subseteq \downarrow\beta)$
$\alpha \mathrel{\tilde{<}} \beta \Leftrightarrow \uparrow(\downarrow\alpha < \downarrow\beta)$

The verb *arrive* holds of objects which arrive at an event time which is contained in the reference time and the verb *sleep* holds of objects which are in a state of sleep at the reference time. When we use the past tense of a verb, the past tense affix requires that the implicit reference time of the verb is located in the past. In the temporal sequencing of two sentences S and T, the input reference time of the second sentence T is equated with the output reference time of the first. If the two sentences are combined by means of the connective *when*, then they are both evaluated with respect to the current reference time. In both the sentential constructions $S._t T$ and *when* S, T the existential quantifier $\mathcal{E}t$ binds a variable, not a discourse marker. These two instances of existential quantification, hence, are *not* dynamic.

These translations merely formalize the treatment of tense as it was described above. As an illustration of the present fragment I indicate how it deals with an example discussed in Partee [1984, pp. 258ff].

An example

Partee discusses the following example:

(8) Mary turned the corner. When John saw her, she crossed the street.

The three verbs in this example are assumed to be eventive. So, the first sentence in this example, and the two constituent sentences of the second sentence are assigned the following (simplified) translations (I have skipped the clauses which locate the events in the past):

Mary turned$_i$ the corner.
$\mathcal{E}d_a\mathcal{E}d_b\mathcal{E}e_i\ [(\uparrow e_i \mathrel{\tilde{\subseteq}} \uparrow d_a)\,;\uparrow\mathtt{turn}(\uparrow e_i)(\uparrow\mathtt{m})\,;(\uparrow e_i \mathrel{\tilde{<}} \uparrow d_b)]$

John saw$_j$ her.
$\mathcal{E}d_a\mathcal{E}d_b\mathcal{E}e_j\ [(\uparrow e_j \mathrel{\tilde{\subseteq}} \uparrow d_a)\,;\uparrow\mathtt{see}(\uparrow e_j)(\uparrow\mathtt{m})(\uparrow\mathtt{j})\,;(\uparrow e_j \mathrel{\tilde{<}} \uparrow d_b)]$

She crossed$_k$ the street.
$\mathcal{E}d_a\mathcal{E}d_b\mathcal{E}e_k\ [(\uparrow e_k \mathrel{\tilde{\subseteq}} \uparrow d_a)\,;\uparrow\mathtt{cross}(\uparrow e_k)(\uparrow\mathtt{m})\,;(\uparrow e_k \mathrel{\tilde{<}} \uparrow d_b)]$

By means of $\uparrow$-export and $\downarrow$-import the translations of the three constituent sentences can be reduced to the following formulas:

$$\mathcal{E}d_a\mathcal{E}d_b\mathcal{E}e_i\ [\uparrow(e_i \subseteq d_a)\,;\uparrow(\mathtt{turn}(e_i)(\mathtt{m}))\,;\uparrow(e_i < d_b)]$$
$$\mathcal{E}d_a\mathcal{E}d_b\mathcal{E}e_j\ [\uparrow(e_j \subseteq d_a)\,;\uparrow(\mathtt{see}(e_j)(\mathtt{m})(\mathtt{j}))\,;\uparrow(e_j < d_b)]$$
$$\mathcal{E}d_a\mathcal{E}d_b\mathcal{E}e_k\ [\uparrow(e_k \subseteq d_a)\,;\uparrow(\mathtt{cross}(e_k)(\mathtt{m}))\,;\uparrow(e_k < d_b)]$$

The last two sentences are combined with the connective *when*, and the result of this is conjoined with the first sentence by means of the temporal sequencing operator $._t$. Spelling out the involved existential disclosures, we arrive at the following translation of example 8:

$$\mathcal{E}t\ [\mathcal{E}d_a\mathcal{E}d_b\mathcal{E}e_i\ [\uparrow(e_i \subseteq d_a)\,;\uparrow(\mathtt{turn}(e_i)(\mathtt{m}))\,;\uparrow(e_i < d_b)]\,;\uparrow(t = d_b)\,;$$
$$\mathcal{E}t'\ [\mathcal{E}d_a\mathcal{E}d_b\mathcal{E}e_j\ [\uparrow(e_j \subseteq d_a)\,;\uparrow(\mathtt{see}(e_j)(\mathtt{m})(\mathtt{j}))\,;\uparrow(e_j < d_b)]\,;\uparrow(t' = d_a)\,;$$
$$\mathcal{E}d_a\mathcal{E}d_b\mathcal{E}e_k\ [\uparrow(e_k \subseteq d_a)\,;\uparrow(\mathtt{cross}(e_k)(\mathtt{m}))\,;\uparrow(e_k < d_b)]\,;\uparrow(t' = d_a)]\,;\uparrow(t = d_a)]$$

Using the associativity equivalences, this formula turns out equivalent with:

$$\mathcal{E}t\mathcal{E}d_a\mathcal{E}d_b\mathcal{E}e_i\ [\uparrow(e_i \subseteq d_a)\,;\uparrow(\mathtt{turn}(e_i)(\mathtt{m}))\,;\uparrow(e_i < d_b)\,;\uparrow(t = d_b)\,;$$
$$\mathcal{E}t'\mathcal{E}d_a\mathcal{E}d_b\mathcal{E}e_j\ [\uparrow(e_j \subseteq d_a)\,;\uparrow(\mathtt{see}(e_j)(\mathtt{m})(\mathtt{j}))\,;\uparrow(e_j < d_b)\,;\uparrow(t' = d_a)\,;$$
$$\mathcal{E}d_a\mathcal{E}d_b\mathcal{E}e_k\ [\uparrow(e_k \subseteq d_a)\,;\uparrow(\mathtt{cross}(e_k)(\mathtt{m}))\,;\uparrow(e_k < d_b)\,;\uparrow(t' = d_a)\,;\uparrow(t = d_a)]]]$$

Applying the closure operator $\downarrow$ we get the truth conditions of example 8. By means of $\downarrow$-import, the result can be reduced to the following formula:

$$\exists t\exists d_a\exists d_b\exists e_i((e_i \subseteq d_a) \wedge \mathtt{turn}(e_i)(\mathtt{m}) \wedge (e_i < d_b) \wedge (t = d_b) \wedge$$
$$\exists t'\exists d_a\exists d_b\exists e_j((e_j \subseteq d_a) \wedge \mathtt{see}(e_j)(\mathtt{m})(\mathtt{j}) \wedge (e_j < d_b) \wedge (t' = d_a) \wedge$$
$$\exists d_a\exists d_b\exists e_k((e_k \subseteq d_a) \wedge \mathtt{cross}(e_k)(\mathtt{m}) \wedge (e_k < d_b) \wedge (t' = d_a) \wedge (t = d_a))))$$

Using some standard reductions, this is equivalent with the following formula:

$$\exists t\exists d_a\exists e_i(e_i \subseteq d_a \wedge \mathtt{turn}(e_i)(\mathtt{m}) \wedge e_i < t \wedge$$
$$\exists d_b\exists e_j(e_j \subseteq t \wedge \mathtt{see}(e_j)(\mathtt{m})(\mathtt{j}) \wedge e_j < d_b \wedge$$
$$\exists d_b\exists e_k(e_k \subseteq t \wedge \mathtt{cross}(e_k)(\mathtt{m}) \wedge e_k < d_b)))$$

In other words, example 8 is true iff Mary turns the corner at an original reference time (labeled by d_a), and if after Mary's turning there is a reference time (t) at which John sees her and at which she crosses the street. Together with the condition that all three events are to be located before the speech time, these are the truth-conditions that Partee ends up with. Furthermore, like in Partee, after example 8 has been processed the reference time is located after the time of Mary's crossing the street.

We see that the *DRT* treatment of tense can be formulated relatively perspicuously within the *DMG* framework. The reformulation can even be argued to be an improvement. The following observation comes from Partee: "(. . .) Hinrich's rules refer to 'the current reference time r_p', which changes in the course of construction (. . .); only the most recent of them is in effect at any given point in the construction of the representation. (. . .) The resulting DRS is in a sense then a dynamic representation rather than a static one." (Partee, [1984, p. 258]) This observation implies that for a treatment of example 8 along the present lines, *DRT* is in need of further dynamification. Notice that a compositional reformulation of *DRT* is not only seen to be easily extended with a *DRT*-like treatment of tense, but that such an extension does not require such a further dynamification.

5 Conclusions

Implicit arguments are generally assumed to be genuine semantic arguments of nouns and verbs which need not be realized syntactically. The translations of such nouns and verbs are conceived of as having an added argument slot, for complements, events, or times, and these can, but need not, be addressed by optional complement or other adjoining phrases. In most existing proposals it is assumed that these argument slots eventually are saturated and that, somewhere in the process of interpretation, they are existentially closed.

Within a dynamic Montague grammar the implicit arguments of nouns and verbs can be assumed to be existentially quantified over right from the start, i.e., in the translation of the nouns and verbs themselves. Since existential quantification in *DMG* is dynamic, such closed arguments can still be addressed afterwards and made subject to further adnominal or adverbial specification. Despite the programmatic nature of the proposals made in this chapter, they demonstrate the usefulness of a dynamic approach to interpretation also at the intrasentential level. The benefits of using a dynamic framework is that it enables us to deal with implicit arguments merely by lexical specification and not by complicating the syntax/semantics interface with optional closure operations, and, furthermore, that it enables us to give a uniform treatment of relational and non-relational nouns, and of verbs that do, and verbs that do not describe events.

Of course, the present proposals may seem a bit laborious. In what is only a small Montagovian fragment of natural language elaborated along the present lines, quite some encoding (and concomitant technical book-keeping) has to be done in the translation of the lexical items. I don't think this is a real objection to the spirit of the proposals. Whichever way one turns, when implicit arguments are addressed they must somehow be there. Of course, one might choose to await their appearance in a 'context' whose formulation is not expedient from a considered semantic point of view. But one can also take them for what they appear to be, viz., objects brought to the fore by the linguistic context. Clearly, we may ask ourselves whether the emergence of these objects had better be dealt with in some pragmatic rather than in a purely semantic part of a theory of interpretation. However, the semantics/pragmatics distinction has to be rethought within a theory of dynamic interpretation anyhow, if it is eventually tenable at all. With the reformulations above I just have tried not to be hampered by such a distinction.

The third point I want to make is that compositionality pays off.[1] Adhering to this (methodological) principle, one is forced to observe ultimate explicitness and generality when making proposals concerning the semantics of natural language expressions and operators. This brought Groenendijk and Stokhof to their, at first glance seemingly technical, but explicitly formalized, proposals concerning the meaning of dynamic natural language noun phrases. The pay off of it is that their rigourous reformulation of *DRT*'s treatment of anaphoric relationships has shown to provide for a framework broad enough to deal with other *DRT* results and with other, relatively different, semantic phenomena.

Here, again, it must be emphasized that it is not just *DMG* that has this extended potential, but that any compositionalization of *DRT* probably equals *DMG* in this expressive force. In the introduction I mentioned alternative compositional

1. This chapter is the written version of a talk held at the Third Symposium on Logic and Language, the topic of which was "Compositionality, representationalism and dynamic theories of meaning".

reformulations, some of which are more faithful to original *DRT*, one that is more *IL*-oid, and another *TY2*-ic. I cannot think of any substantial reason here for preferring any one over the other. As for logical issues, Muskens' [1990] *TY2*-like variant might be preferable, since his language of translation is more explicit about what is going on technically speaking. As a tool for describing natural language phenomena, on the other hand, one might prefer *DMG*, since the translations of natural language expressions within the language of *DFL* have a structure which more closely resembles the structure of the natural language expressions themselves. The choice between anyone of these systems may be guided by personal preference and purposes. The main point is that such a choice does not affect potential empirical coverage.

One last thing remains to be concluded. I argued that a compositional dynamic semantics is quite powerful and that the adherence to compositionality pays off. However, the major pay off is an integration and further compositionalization of other, differently oriented, semantic proposals such as has been sketched in this chapter. And in fact, what we have done in this chapter was nothing more than giving a programmatic, compositional reformulation of independently proposed partial descriptions of the semantics of natural language phenomena. Our final conclusion, therefore, cannot be other than that non-compositionality apparently pays off as well (at least sometimes).

Chapter 5

Updates in dynamic semantics

Groenendijk and Stokhof, in [1990b], characterize the meaning of a sentence in a *static semantics* as the set of indices at which the sentence is true. Such a set of indices, also called an information state, defines the information content of the sentence. In a *dynamic semantics*, it is not the information content, but the information change potential of a sentence that is regarded as constituting its meaning. In a dynamic semantics the meaning of a sentence is, or can be considered to be, a function from information states to information states. Groenendijk and Stokhof compare two examples of such a dynamic semantics, namely dynamic predicate logic (*DPL*, Groenendijk and Stokhof [1991]) and update semantics (*US*, Veltman [1990]).

DPL and *US* constitute two different examples of a dynamic semantics. The two systems formalize different aspects of the dynamics of discourse and in their present formulations they have conflicting logical properties. *DPL* accounts for the context change potential of indefinite noun phrases to set up discourse referents for subsequent anaphoric pronouns, and *US* formalizes the update of information about the world that results from interpreting successive sentences in a discourse. Furthermore, as Groenendijk and Stokhof point out, *DPL* interpretation is distributive and non-eliminative and interpretation in *US* is eliminative and non-distributive. As a result, the two systems employ different notions of truth and entailment which should not be substituted for one another.

In this chapter, I will move towards an integration of the two systems, by reformulating *DPL* as an update semantics. The resulting system, *EDPL*, defines the meaning of the formulas of predicate logic as genuine update functions that update information about the values of variables. Thus, the *US* notions of truth and entailment become applicable to *DPL*, and this enhances the combination of the two systems. Pay off of this reformulation of *DPL* is that it is easily extended with a uniform and perspicuous treatment of nominal and adverbial quantification and, moreover, that it constitutes a substantial step towards the development of a broader theory of information *exchange* that comprises both *DPL* and *US*.

I will proceed as follows. In the first section I sketch the two theories of dynamic semantics at issue. I present *DPL* in its so-called functional formulation, and *US* is sketched in its most rudimentary form. I then compare the two systems and argue for the desirability of an update-style formulation of *DPL*. Section 2 presents and discusses the reformulation of *DPL* as the update semantics *EDPL*, and in section 3 this system is extended with quantifiers. Finally, section 4 offers a more thorough study of the notion of information in *EDPL*. This section introduces information about the world, studies the structure of information states, and it offers some observations concerning the modelling of phenomena of information exchange within the framework of *EDPL*.

1 Two theories of dynamic semantics

1.1 Introduction

A dynamic semantics formalizes the insight that the meaning of a sentence is its potential to change information, rather than to express it. As Veltman [1990] puts it, the idea is that "you know the meaning of a sentence if you know the change it brings about in the information state of anyone who accepts the news conveyed by it". The dynamic conception of meaning contrasts with the standard, static, conception of meaning according to which "you know the meaning of a sentence if you know the conditions under which it is true". The dynamic notion of meaning dates back to Stalnaker [1974] and is adopted by quite a number of authors.

That there is more to a truly dynamic semantics than complying with this slogan can be observed as follows. It is easy to give static systems of interpretation a dynamic twist. For instance, we can give the language of propositional logic the following dynamic interpretation. Let I be a singleton set of indices $\{\emptyset\}$, and let the set of information states S be the powerset $\mathcal{P}(I)$ of I. There are two information states then, $\{\emptyset\}$ and $\emptyset$, the true and the false state, respectively. Furthermore, let V be a valuation function that assigns the proposition letters of the language sets of indices as interpretation (i.e., truth values in the present case). The interpretation function $[\![\ ]\!]$ then assigns to each formula an update function on the set of information states. It is defined as follows ($s[\![\phi]\!]_V$ indicates the result of updating an information state s with ϕ with respect to V, that is, of applying the function $[\![\phi]\!]_V$ to s):

(1) $s[\![p]\!]_V = \{i \in s \mid i \in V(p)\}$
$s[\![\neg\phi]\!]_V = s - s[\![\phi]\!]_V$
$s[\![\phi \wedge \psi]\!]_V = s[\![\phi]\!]_V[\![\psi]\!]_V$

(2) ϕ is true in s with respect to V iff $s \subseteq s[\![\phi]\!]_V$

Of course, what we find here is a completely trivial dynamic semantics, since it is just propositional logic formulated in a more complicated way. A formula ϕ is true in all $s \in S$ with respect to V in this dynamic propositional logic iff ϕ is true with respect to V in ordinary, static, propositional logic.

Nevertheless, this tiny exercise in dynamic semantics gives us a taste of systems of dynamic interpretation and it may serve to figure as a more general leg up to the two non-trivial examples of dynamic semantics which make up the subject of this chapter, viz., *DPL* and *US*. Information states in dynamic propositional logic are sets of indices (the present example is a borderline case, in the sense that there is only one possibility, viz., truth) and the interpretation of a formula is a function on the domain of information states. Atomic formulas are assigned an information content that intersects with the input information state. Negation is associated with set or state subtraction and sentence conjunction is associated with function composition. If we interpret the conjunction of two formulas ϕ and ψ in a state s, we first interpret ϕ in s and next interpret ψ in the state that results from the update with ϕ.

1.2 Dynamic predicate logic

DPL gives a dynamic interpretation of the language of first order predicate logic that accounts, among other things, for intersentential anaphoric relationships as found in *A cowgirl meets a boy. She slaps him*. Like in discourse representation theory and file change semantics, in *DPL* natural language noun phrases are associated with variables, or discourse markers, and information states determine what values they can have given the conditions imposed on them in the course of a discourse.

For instance, if we interpret *A cowgirl meets a boy* and associate *a cowgirl* with a variable x and *a boy* with a variable y, then we end up in a state which contains the information that the value of x is a cowgirl who meets a boy who is the value of y (if there is such a cowgirl, otherwise the sentence is just false). This state is precisely the kind of state we need to interpret a continuation with *She slaps him*, where *she* is associated with x again, and *him* with y. This second sentence then adds the information that x slaps y, and the state that results from interpreting the sequence of two sentences contains the information that the value of x is a cowgirl who meets and slaps a boy who is the value of y.

DPL semantics

Since variable assignments are useful ways to carry around information about variables and their reference, information states in *DPL* are thought of as sets of variable assignments. So if D is a domain of individuals, and V the set of variables we use, then any subset $s \subseteq D^V$ of the set of variable assignments is an information state and S, the set of all information states, is the powerset of the set of variable assignments: $S = \mathcal{P}(D^V)$. The set of information states contains a minimal state $s = D^V$, in which all variable assignments are still possible, an absurd state $s' = \emptyset$, which excludes all possibilities, and maximal information states $\{i\}$, for $i \in D^V$, which completely determine the value of all variables.

The language of *DPL* is that of predicate logic, but for ease of exposition I disregard individual constants. The semantics is defined with respect to a model $M = \langle D, F \rangle$ consisting of a non-empty set of individuals D and an interpretation function F that assigns sets of n-tuples of objects to n-ary relation expressions. (I omit reference to M whenever this does not lead to confusion.) The interpretation of formulas is a function on the domain of information states ($i[x/d]$ is the assignment j such that j agrees with i on the values of all the variables except possibly x, and such that $j(x) = d$):

Definition 1.1 (Semantics of DPL)

- $s[\![Rx_1 \dots x_n]\!] = \{i \in s \mid \langle i(x_1), \dots, i(x_n) \rangle \in F(R)\}$
- $s[\![x = y]\!] = \{i \in s \mid i(x) = i(y)\}$
- $s[\![\neg\phi]\!] = s - \downarrow[\![\phi]\!]$
- $s[\![\phi \wedge \psi]\!] = s[\![\phi]\!][\![\psi]\!]$
- $s[\![\exists x\phi]\!] = s[x][\![\phi]\!]$

where

$\downarrow[\![\phi]\!] = \{i \mid \{i\}[\![\phi]\!] \neq \emptyset\}$
$s[x] = \{i[x/d] \mid i \in s \;\&\; d \in D\}$

The interpretation of an atomic formula in a state s involves the intersection of s with the set of assignments with respect to which the formula is true in a classical sense. The negation of ϕ subtracts those i in s which constitute a context $\{i\}$ with respect to which the interpretation of ϕ does not produce the absurd state. Putting it the other way around, the negation of ϕ preserves those i in s with respect to which ϕ produces the absurd information state. Conjunction is just function composition.

The characteristic clause concerns the interpretation of the existential quantifier. If we interpret a formula $\exists x\phi$ in a state s, we take into consideration all values for x and then interpret ϕ. The mediating state $s[x]$ contains the same information as s about the values of all variables except x. About x, $s[x]$ has no information whatsoever: for each i in s (and for each d in D) there is a j in $s[x]$ that is like i except possibly with respect to the value assigned to x (and such that $j(x) = d$).

I will not work out the *DPL* interpretation of various examples here. (For a detailed account the reader is referred to Groenendijk and Stokhof [1990b] and [1991].) For the present purposes it suffices to point out some characteristic facts about *DPL* interpretation.

Some characteristic facts

The interpretation of an existentially quantified formula $\exists x\phi$ can be conceived of as involving the composition of two operations on states: $[x]$ and $[\![\phi]\!]$, respectively.

As we saw, also conjunction amounts to function composition. Since composition is an associative operation, the following equivalences hold in *DPL*. The first one is a classical conjunction, but the second one distinguishes *DPL* from static theories (the arrow $\Leftrightarrow$ indicates identity of meaning):

Fact 1.1 (Donkey equivalences (1))

- $((\phi \wedge \psi) \wedge \chi) \Leftrightarrow (\phi \wedge (\psi \wedge \chi))$
- $(\exists x\phi \wedge \psi) \Leftrightarrow \exists x(\phi \wedge \psi)$

It is typical of *DPL* that the second equivalence also holds if the variable x is free in ψ. These equivalences therefore allow *DPL* to deal with the following textbook example (which explains the equivalences' label):

(3) A farmer owns a donkey. He beats it.
$(\exists x(Fx \wedge \exists y(Dy \wedge Oxy)) \wedge Bxy) \Leftrightarrow$
$\exists x(Fx \wedge \exists y(Dy \wedge (Oxy \wedge Bxy)))$

This sequence of sentences turns out to be equivalent with the sentence *A farmer owns a donkey that he beats*. So, although the two sentences *A farmer owns a donkey* and *He beats it* are assigned an interpretation of their own, the intersentential anaphoric relationships (between *a farmer* and *he*, and between *a donkey* and *it*) get established when we combine the two in a conjunction.

The donkey equivalences have a nice corollary. Given the usual definitions of universal quantification and implication in terms of existential quantification, negation and disjunction, we also have the following facts (again, the first one is a classical equivalence and the second one a typical *DPL* equivalence):

Fact 1.2 (Donkey equivalences (2))

- $((\phi \wedge \psi) \rightarrow \chi) = \neg((\phi \wedge \psi) \wedge \neg\chi) \Leftrightarrow \neg(\phi \wedge (\psi \wedge \neg\chi)) = (\phi \rightarrow (\psi \rightarrow \chi))$
- $(\exists x\phi \rightarrow \psi) = \neg(\exists x\phi \wedge \neg\psi) \Leftrightarrow \neg\exists x(\phi \wedge \neg\psi) = \forall x(\phi \rightarrow \psi)$

This enables *DPL* to deal with the museum piece donkey sentences:

(4) If a farmer owns a donkey he beats it.
$(\exists x(Fx \wedge \exists y(Dy \wedge Oxy)) \rightarrow Bxy) \Leftrightarrow$
$\forall x(Fx \rightarrow \forall y((Dy \wedge Oxy) \rightarrow Bxy))$

(5) Every farmer who owns a donkey beats it.
$\forall x((Fx \wedge \exists y(Dy \wedge Oxy)) \rightarrow Bxy) \Leftrightarrow$
$\forall x(Fx \rightarrow \forall y((Dy \wedge Oxy) \rightarrow Bxy))$

These sentences are assigned their so called 'strong' readings. Both sentences state that every farmer beats every donkey he owns. It is the merit of *DPL* that it gives a compositional treatment of these examples, which before, in *DRT* (see Kamp [1981]), had supplied motivation for a representational and non-compositional treatment.

Another typical property of *DPL* is that it has a non-eliminative semantics. It is not generally true that $s[\![\phi]\!] \subseteq s$. Interpretation in *DPL* does not merely involve the elimination of possibilities, but it may also involve the introduction of new possibilities. For instance, the interpretation of a formula $\exists x\phi$ in a state s may contain assignments that assign a cowgirl to x, whereas all assignments in s assign a man to x. The existential quantifier $\exists x$ sets up a new discourse referent, so to speak, by resetting the value of a variable x, and it is precisely this sort of 'act' in *DPL*, together with the fact that these acts affect the interpretation of variables in successive formulas, that makes *DPL* distinct from static theories.

Related to its non-eliminativity is the fact that *DPL* conjunction is not commutative ($\not\Leftrightarrow$ indicates the possibility of non-identity of meaning):

Fact 1.3 (Non-commutativity)

- $\phi \wedge \psi \not\Leftrightarrow \psi \wedge \phi$

An example of a non-commutative conjunction is $\exists xMx \wedge Wx$, a *DPL* paraphrase of a sequence like *A man walks in the park. He whistles.* In *DPL*, and also intuitively, the meaning of this formula differs from that of $Wx \wedge \exists xMx$, a paraphrase of *He whistles. A man walks in the park*.

The last property of *DPL* that is discussed here is that the *DPL* interpretation of any formula is a distributive function:

Fact 1.4 (Distributivity)

- $s[\![\phi]\!] = \bigcup_{i \in s}\{i\}[\![\phi]\!]$

Distributivity means that in the update of a state s with a formula ϕ only properties of the individual elements of s count, and not global properties of the state as a whole. This implies that *DPL* interpretation can be given a definition in a lower type, viz., as a relation between assignments instead of as a function on sets of assignments. And indeed the original formulation of *DPL* in [1991] is the relational one.

Notice that distributivity is, as it were, assumed in the clause that defines the interpretation of the negation of a formula ϕ. Since the interpretation of ϕ in a state s is completely characterized by the update of singleton subsets of s by ϕ, the negation of ϕ in s can be defined appropriately in terms of the update by ϕ of each of these singleton subsets.

Truth and entailment

I now turn to truth and entailment in *DPL*:

Definition 1.2 (Truth and entailment in DPL)

- ϕ is true in s with respect to M, $s \models_M \phi$, iff $s \subseteq \downarrow[\![\phi]\!]_M$
- $\phi_1, \ldots, \phi_n \models \psi$ iff $\forall M, s$: $s[\![\phi_1]\!]_M \ldots [\![\phi_n]\!]_M \models_M \psi$

A formula ϕ is true in a state s iff all possible variable assignments $i \in s$ consitute a state $\{i\}$ which can be succesfully updated with ϕ (i.e., in which the interpretation of ϕ does not produce the absurd state). A conclusion ψ follows from a sequence of premises $\phi_1, \ldots \phi_n$ if the update of any information state s with $\phi_1, \ldots, \phi_n$, successively, produces a state in which ψ is true.

DPL licenses the deduction theorem:

Fact 1.5

- $\phi_1, \ldots, \phi_n \models \psi$ iff $\phi_1, \ldots, \phi_{n-1} \models \phi_n \rightarrow \psi$

From the deduction theorem and the donkey equivalences it follows that entailment in *DPL* is dynamic. For instance, $\exists x Fx$ entails Fx. This fact corresponds to the following line of elementary reasoning:

(6) If a man comes from Rhodes, he likes pineapple-juice. A man I met yesterday comes from Rhodes. So, he likes pineapple-juice.
$\exists x(Mx \wedge Rx) \rightarrow Lx, \exists x(Mx \wedge Rx) \models Lx$

1.3 Update semantics

I now turn to Veltman's first example of an update semantics in [1990]. It is a (dynamic) propositional logic with an additional sentential operator $\diamond$. (In fact, this system only gives the rudiments of a much more interesting system. However, since the basic system of *US* already employs notions of truth and entailment which are crucially different from the ones used in *DPL*, it seems natural to compare *DPL* with the rudimentary system in the first place.)

The kind of information *US* deals with is information about the world. Information states are modeled as subsets of the set of possible worlds W. For someone in information state s, each world in s might correspond to the real world. (Veltman doesn't really use worlds, but sets of atomic sentences which "might give a correct picture of reality". I neglect this difference in what follows.) The minimal information state is again the set of all possibilities, W this time, the absurd state is the empty set, and a state of maximal information is any singleton subset of W.

An atomic sentence p in *US* updates an information state s by eliminating the worlds in s which are inconsistent with p, and negation and conjunction are interpreted as set subtraction and composition, respectively. So far the semantics of *US* closely corresponds to the dynamic formulation of the semantics of propositional logic given at the start of this section. The interesting bit comes in with the operator $\diamond$. In an information state s, $\diamond\phi$ tests whether ϕ is still consistent with s. Like its natural language counterpart *might*, it reflects upon the present information state and expresses that that state can be consistently updated with ϕ. So, if ϕ is acceptable in a state s, then $\diamond\phi$ is true in that state; however, if you already have the information that ϕ is false, then $\Diamond\phi$ has to be rejected and its interpretation returns the absurd state.

The semantics of *US* is defined with respect to a model $M = \langle W, V\rangle$ consisting of a set of worlds W and an interpretation function V that assigns sets of worlds to proposition letters. (Again, reference to M is omitted when that does not lead to confusion.) The interpretation of a formula is defined as an update function on states:

Definition 1.3 (Update semantics)

- $s[\![p]\!] = \{i \in s \mid i \in V(p)\}$
- $s[\![\neg\phi]\!] = s - s[\![\phi]\!]$
- $s[\![\phi \wedge \psi]\!] = s[\![\phi]\!][\![\psi]\!]$
- $s[\![\Diamond\phi]\!] = \{i \in s \mid s[\![\phi]\!] \neq \emptyset\}$

As is evident from the semantics of *US*, the result of interpreting a formula in a state s is always a subset of s. Interpretation can only *eliminate* possibilities:

Fact 1.6 (Eliminativity)

- $s[\![\phi]\!] \subseteq s$

The eliminativity of *US* interpretation implies that interpretation guarantees update of information. In *US*, the interpretation of any formula in a state s always produces a state which contains at least as specific information about the world as s.

The $\Diamond$-operator reflects upon the current stage in the process of information growth. The interpretation of $\Diamond\phi$ in a state s returns s if ϕ is acceptable in s, and the absurd state if ϕ is not acceptable in s. Since *US* interpretation is eliminative, a formula $\Diamond\phi$ is 'unstable', this in the sense that, in the gradual update of information, at some stage $\Diamond\phi$ may be true (if ϕ is not excluded at that stage), whereas at a later stage it is false (if the possibility that ϕ has been excluded in the meantime).

The unstability of $\Diamond\phi$ causes conjunction in *US* to be non-commutative:

Fact 1.7 (Non-commutativity)

- $\phi \wedge \psi \not\Leftrightarrow \psi \wedge \phi$

An example of a non-commutative conjunction is $\Diamond p \wedge \neg p$. This conjunction, with this order of conjuncts, is consistent. For instance, let s be a state which is undecided about the truth or falsity of p, i.e., a state in which both p and $\neg p$ are acceptable. In that case, the update of s with $\Diamond p$ is s, since p is acceptable in s, and further update with $\neg p$ is not unacceptable. On the other hand, the commutation $\neg p \wedge \Diamond p$ of this conjunction is inconsistent. The interpretation of the formula $\neg p$ produces an information state in which p is false and, consequently, $\Diamond p$ unacceptable. So, we see that $\neg p \wedge \Diamond p$, which is inconsistent, is not equivalent with $\Diamond p \wedge \neg p$, which is consistent.

The following pair of examples exemplifies this pattern (granted that we know that John is not Mary):

(7) Somebody is knocking at the door. .... It might be John. .... It's Mary.
(8) Somebody is knocking at the door. .... It's Mary. .... *It might be John.

If somebody hears someone knocking at the door, he may of course be curious who it is and not exclude the possibility that it is, say, John. Still, in that situation it is perfectly possible for him to find out that it is Mary who is knocking, not John. On the other hand, once he has found out that Mary is knocking on the door, it is excluded that it is John, and then it is quite absurd to say that, as far as he knows, it might be John who is knocking at that door. (Counterfactuals like *But it might have been John* are not discussed in this chapter.)

Another property of *US* interpretation worth pointing out is that it is not (always) distributive. A formula $\Diamond\phi$ tests a global property of a state s, viz., its consistency with ϕ, which does not need to hold of all singleton subsets of s. For instance, suppose $s = \{i, j\}$, and $\{i\}[\![\phi]\!] = \{i\}$ and $\{j\}[\![\phi]\!] = \emptyset$. Then $s[\![\Diamond\phi]\!] = s$. However, $\{i\}[\![\Diamond\phi]\!] = \{i\}$ and $\{j\}[\![\Diamond\phi]\!] = \emptyset$. So $\bigcup_{i\in s}\{i\}[\![\Diamond\phi]\!] = \{i\}$, which is different from $s[\![\Diamond\phi]\!] = \{i, j\}$.

I now turn to truth and entailment in *US*. In fact, Veltman does not speak of truth but of acceptance. If, as I will say it, ϕ is true in a state s, Veltman would say that ϕ is accepted in s. Truth and entailment are defined as follows (Veltman discusses two alternative notions of entailment, but I neglect those here):

Definition 1.4 (Truth and entailment in update semantics)

- ϕ is true in s with respect to M, $s \models_M \phi$, iff $s \subseteq s[\![\phi]\!]_M$
- $\phi_1, \ldots, \phi_n \models \psi$ iff $\forall M, s$: $s[\![\phi_1]\!]_M \ldots [\![\phi_n]\!]_M \models_M \psi$

A formula ϕ is true in s if after updating s with ϕ we still envisage the possibilities we envisaged in state s, i.e., if $s[\![\phi]\!]$ doesn't contain more information than s. A sequence of premises $\phi_1, \ldots, \phi_n$ entail a conclusion ψ if always, if you update your information with $\phi_1 \ldots \phi_n$, in that order, you arrive at a state of information to which update with ψ adds no more information.

US licenses the deduction theorem:

Fact 1.8

- $\phi_1, \ldots, \phi_n \models \psi$ iff $\phi_1, \ldots, \phi_{n-1} \models \phi_n \rightarrow \psi$

I leave it at this exposition of basic properties of rudimentary *US*. For more details, and for more interesting extensions of *US*, the reader is referred to Veltman [1990].

1.4 A comparison

DPL and *US* are genuinely dynamic systems as can be seen from the fact that in none of the two conjunction is commutative. The example $\Diamond\phi \wedge \neg\phi$ is a counterexample to commutativity in *US*, and in *DPL* a counterexample is $\exists xFx \wedge Gx$.

As Groenendijk and Stokhof observe, the respective properties of non-eliminativity and non-distributivity serve to distinguish the two systems from classical theories of interpretation. It is fairly easily shown that a dynamic semantics in which all sentences are interpreted as eliminative and distributive updates is not really dynamic after all.[1] However, as we have seen above, *DPL* is non-eliminative, since after interpreting an existentially quantified formula $\exists x\phi$ in a state s, the information s has about the value of x is lost. Furthermore, *US* is non-distributive, since the *might*-operator expresses global properties of an information state which do not need to hold of all singleton subsets of that state. So, the respective properties of non-eliminativity and non-distributivity distinguish *DPL* and *US* from static theories of interpretation.

It is important to notice that distinct properties distinguish *DPL* and *US* from static theories, non-eliminativity and non-distributivity respectively. These two different ways in which *DPL* and *US* depart from the static paradigm are reflected by a difference in the respective notions of truth (and, hence, of entailment). For ψ to be true in s in *DPL*, *each* singleton subset s must allow update with ψ. This notion of truth is that of a distributive system. Truth does not depend on inherently global properties of s, but only on properties of each of the individual members of s. On the other hand, the *US* notion of truth is that of an eliminative system. For ψ to be true in s, *US* requires that the update of s with ϕ does not eliminate possibilities in s. Given eliminativity, this says that ψ is true in s if s already contains the information conveyed by ψ.

From these remarks it may already be clear that the two different notions of truth and entailment should not be substituted for one another. *US*' notion of truth, which is that of an eliminative system, is adverse to *DPL*'s non-eliminativity, and *DPL*'s distributive notion of truth is adverse to *US*' non-distributivity. For instance, if we adopt the *US* notion of truth in *DPL*, then the *DPL*-valid entailment $\exists xPx \models \exists yPy$ would no longer be valid. The reason is that, on the *US* notion of truth, $\exists yPy$ is true in s iff Py is true in s, and, clearly, $\exists xPx \not\models Py$ on any of the two notions of entailment. On the other hand, if we adopt the (distributive) *DPL* notion of truth in *US*, then the *US*-valid entailment $\Diamond p \models \Diamond p$ would no longer go through. The reason is that, on the *DPL* notion of truth, $\Diamond p$ is true in s iff p is true

1. If a function τ on a domain of sets is distributive and eliminative, then for all sets s, $\tau(s) = s \cap {\downarrow}\tau$, where ${\downarrow}\tau$ is the characteristic set $\{x \mid \tau(\{x\}) = \{x\}\}$ (cf., Groenendijk and Stokhof [1990b, p. 57] and van Benthem [1986, p. 62] and [1991, p. 137]). Now suppose that two sentences ϕ and ψ are interpreted as eliminative and distributive updates τ and τ' (with characteristic sets ${\downarrow}\tau$ and ${\downarrow}\tau'$, respectively) and that sentence conjunction involves function composition. Then $s[\![\phi \wedge \psi]\!] = s[\![\psi \wedge \phi]\!]$ since $\tau'(\tau(s)) = \tau(\tau'(s)) = s \cap {\downarrow}\tau \cap {\downarrow}\tau'$. So, in a distributive and eliminative system of dynamic interpretation, conjunction amounts to (commutative) intersection.

in s, and $\Diamond p \not\models p$ on any of the two notions of entailment.

We see that *DPL* and *US* are two really different systems of dynamic interpretation with conflicting characteristic properties. This is not to say that the two are incompatible. As Groenendijk and Stokhof suggest, the two systems can be combined within a system that preserves the characteristic features of both and that, to some extent, gives a separate treatment of the two different kinds of (update or change of) information that the two systems deal with.

However, in this chapter I want to show that it is worthwhile to remove one of the differences between the two systems, by adapting the logic of one of the two (*DPL*) to the format of that of the other (*US*). Doing so, a combination of the two theories results which is more of an *integration*, since it allows us to employ one elementary notion of truth (and entailment) instead of the product of two, so to speak. Crucial to this reformulation is the development of a genuine notion of update of information about the values of variables, for it is there that *DPL*'s non-eliminativity resides.

Information about the values of variables

Both *DPL* and *US* formalize dynamic aspects of natural language interpretation by interpreting formulas as functions that change or update information states. (I use 'update' from now on to indicate, specifically, *addition* of information, not arbitrary *revision*.) Basically, *US* models update of information about the world brought about by processing sentences. Interpretation in *US* therefore involves a process of information *growth*. If we accept a discourse, our information about the world increases. This corresponds to the fact that interpretation in *US* is eliminative. The state that results from interpreting a formula in a state s contains at least the information that s contains, and maybe more.

The dynamics of *DPL* is of a quite different nature. "It [*DPL*, PD] (...) restricts the dynamics of interpretation to that aspect of the meaning of sentences that concerns their potential to 'pass on' possible antecedents for subsequent anaphors (...)."(Groenendijk and Stokhof, [1991, p. 43–4]) The dynamics of *DPL*, basically, serves to give a compositional account of the phenomenon that existential quantifiers (indefinite noun phrases) may bind variables (pronouns) occurring outside their proper scope. For this reason, the kind of information employed in *DPL* is only a means to establish (the semantics of) anaphoric links. *DPL*'s information states encode the information that is required at a certain stage of interpretation to establish the semantic relations between existential quantifiers that have occurred before and free variables yet to come. Hence, if a discourse has been interpreted from beginning to end, the information contained in the resulting state is superfluous information, since no more anaphoric relationships will be established afterwards.

Significant in this respect is *DPL*'s definition of truth. In order to assess whether a discourse δ is true in a state s in *DPL*, we do *not* look at the state $s[\![\delta]\!]$ that results from interpreting δ in s, but we check whether the interpretation of δ in all singleton subsets $\{i\}$ of s yields non-absurd output. In other words, it does not matter what information is contained in the state that results from interpreting δ (in s or in the singleton subsets of s), all that matters is that the interpretation of δ is possible in (all singleton subsets of) s.

Clearly, as it is used in *DPL*, the kind of information that *DPL* models should not be confused with the kind of information language users typically aim to exchange by uttering and interpreting sentences, this in contradistinction with *US*. Nevertheless, such a notion of update and exchange of information about the values of variables has some intuitive appeal. Human agents talk about 'indefinite' objects, objects which they have partial knowledge of. These objects are talked about, they are ascribed properties and people are informed about their existence. Typically, these 'objects' are not to be conceived of as classical objects. From the perspective of agents with partial information the ultimate identity of such objects may be left unresolved without this prohibiting their use as topics of information exchange. Such objects can be taken to be partial objects, or 'pegs' in the sense of Landman [1986]. They are things that don't have properties, but to which properties are ascribed and, similarly, things that don't have identity conditions, but that have identity conditions ascribed to them. As Landman puts it: "the essence of partial information is that it cannot justify certain distinctions, and the decision about the identity of certain pegs is a prime example of that" (Landman [1986, p. 126]). Crucial here is the idea that the process of information exchange involves things (pegs in Landman's data semantics, (values of) variables in the subsequent section), which "are the objects we assume in conversation, and which we follow through information growth" (Landman, [1986, p. 128]).

What is relevant, furthermore, is that natural language indefinites typically introduce new objects, or pegs, into the domain of objects we have information about. This idea traces back, in rudimentary form, to Karttunen's seminal papers [1968a, 1968b]. Karttunen interprets indefinites as *establishing* discourse referents, which are *novel individuals* in the discourse. This idea is further developed within the frameworks of discourse representation theory (*DRT*, Kamp [1981]) and file change semantics (*FCS*, Heim [1982, 1983]). In both *DRT* and *FCS*, indefinite noun phrases induce a genuine update of the discourse representation (or file, respectively) that constitutes the context of interpretation by the addition of a novel discourse referent to its domain. Hence, disregarding *DRT*'s intermediary representations, the information that is passed on when processing successive sentences in a discourse determines, first, which variables are under discussion, and, second, what properties these variables have. These two different aspects of information are associated with two different kinds of information update. One way of updating one's information consists in getting more informed about the objects (i.e., the values of variables) one has information about. Another way consists in extending this domain of objects.

Update of information about the values of variables

The aim of section 3 below is to give a reformulation of *DPL* in terms of *EDPL* that complies with the idea that sentence interpretation involves genuine update of information. Like in *DPL*, information in *EDPL* remains confined to information about the values of variables, but, in contradistinction with *DPL*, in *EDPL* this kind of information is not a mere means of establishing anaphoric relationships, but is understood to model a genuine form of information about partial objects which can be exchanged by uttering and interpreting sentences. In section 5, I will give more substance to this claim that the structure of information states employed in *EDPL* may serve to model exchange of information about partial objects. Before that, in sections 3 and 4, I will only focus on the formulation of a predicate logic update semantics. Section 3 provides this reformulation of *DPL* which is seen to be easily extended with quantifiers in section 4. The most striking properties of *EDPL* are the following.

Like in *DPL*, information about the values of variables is encoded in *EDPL* by sets of variable assignments. However, *EDPL* uses sets of *partial* variable assignments. For any subset X of the set of variables, an information state about the values of X is a set of assignments of objects to the variables in X. An information state in *EDPL* hence determines the two aspects of information addressed above. In the first place such a state s determines a domain of variables whose values are at issue, viz., the joint domain of the assignments in s. In the second place s contains information about the values of variables in that domain.

Since information states in *EDPL* model two aspects of information about variables, like in *DRT* and *FCS*, we can also distinguish two basic kinds of update of information. Update of information consists either in getting more information about the values of variables, or in getting information about the values of more variables. In other words, update consists either in restricting the set of partial variable assignments by elimination, or in extending the domain of partial variable assignments, or, of course, in a mixture of both. A state t, then, is considered a possible update of a state s, written as $s \leq t$, if the domain of t comprises that of s and if the assignments in t satisfy the restrictions s imposes on the values of variables in the domain of s. Clearly, on such a notion of update, t is an update of s if t at least contains the information that s has about the values of the variables in the domain of s. But, of course, t may contain more information about more variables. As will be seen in section 3.2, *EDPL* interpretation always yields information update.

Let us now turn to the details of *EDPL*.

2 EDPL, an update semantics for DPL

In this section, I will first introduce the notions of information and of information update in *EDPL* (section 3.1). In section 3.2 I define the semantics of *EDPL* as a partial update function on the domain of information states and discuss some general properties of *EDPL* interpretation. Section 3.3 discusses *EDPL* interpretation in more detail, and section 3.4 introduces the *EDPL* notion of entailment and compares *EDPL* with *DPL*.

2.1 Information states in EDPL

Following *DPL* I will define the interpretation of the language of predicate logic as a function on a domain of information states. Like in *DPL*, in *EDPL* these information states encode information about the values of variables in a discourse, but contrary to what is the case in *DPL*, in *EDPL* these states are sets of partial variable assignments.

If D is a domain of individuals and V the set of variables, then S^X, the set of information states about the values of $X \subseteq V$, and S, the set of all information states, are defined as follows:

Definition 2.1 (Information states)

- $S^X = \mathcal{P}(D^X)$
- $S = \bigcup_{X \subseteq V} S^X$

An information state about the values of a set of variables X is a set of assignments of individuals to the variables in X. Given such a state $s \in S^X$, I will refer to X as the domain of s, written as $D(s)$. Information states contain information about the values of the variables in their domain by restricting their valuations. An information state s is a set of valuations of the variables in the domain of s which are conceived possible in s, and, hence, excludes all other valuations to these variables. So, if x and y are in the domain of s, then s contains the information that x is a man iff no i in s maps x on an individual that is not a man, and s contains the information that x sees y, iff for all i in s, $i(x)$ sees $i(y)$.

With respect to a fixed domain of variables, the notions of minimal and maximal information states are as in *DPL* and *US*. For any domain of variables X, the minimal information state about the values of X is D^X, referred to as $\top^X$. A minimal information state has no information about the values of the variables in in its domain, since it considers all valuations of the variables possible. A maximal information state about the values of X is $\{i\}$ for any $i \in D^X$. A maximal state has total information about the values of the variables in its domain, since only one valuation of them is conceived possible. Furthermore, for any domain X, the absurd information state is $\emptyset$, referred to as $\bot^X$. An absurd information state excludes all assignments to the variables in its domain.[2]

A special set of information states is $S^{\emptyset}$, the possible states of information about the values of no variables. There are only two such states: the set containing the empty assignment, and the empty set. (In fact this is just the domain of truth values on their set-theoretic definition.) Interestingly, with respect to the empty domain the minimal information state and the maximal information state coincide. This reflects the fact that one can have no substantial information about the values of no variables. Notice, moreover, that S^V is the set of *(D)PL* information states, the set of sets of assignments to all variables. So, the set of states S encompasses the states of propositional logic ($\{1,0\}$) and those of (dynamic) predicate logic ($\mathcal{P}(D^V)$).

Since, in general, we will be dealing with assignments and states which may have different domains, it is useful to have at our disposal two notions which can be conceived to be generalizations of the notion of being an element of a set. These notions are defined as follows ($i \leq j$ says that j extends i, i.e., for all x in the domain $D(i)$ of i, x is in the domain $D(j)$ of j and $i(x) = j(x)$):

Definition 2.2 (Restriction and extension)

- i has a restriction in s, $i \gtrdot s$, iff $D(s) \subseteq D(i)$ and $\exists j \in s$: $j \leq i$
- i has an extension in t, $i \lessdot t$, iff $D(i) \subseteq D(t)$ and $\exists j \in t$: $i \leq j$

Assignment i has a restriction in s if i is an extension of some element of s. Assignment i has an extension in t if i is a restriction of some element of t. In the latter case, I will also say that i survives in t. Clearly, if the domain of s equals the domain of i, then $i \gtrdot s$ iff $i \in s$ iff $i \lessdot s$.

The two notions of having a restriction and having an extension give rise to two generalizations of the subset relation, the update relation and the substate relation respectively. I start with the update relation.

The fact that a state t contains more information than a state s, or, in other words, the fact that state t is an update of s is defined as follows:

Definition 2.3 (Update)

- t is an update of s, $s \leq t$, iff $D(s) \subseteq D(t)$ and $\forall i \in t$: $i \gtrdot s$

2. In an earlier version of this paper (Dekker [1992]) I have identified any two absurd states regardless of their domain. In that paper I confined myself to giving an update formulation of *DPL* and for that purpose there was no great use in distinguishing these states. The absurd states are all states irrevocably deemed to contain impossible information.
However, strictly speaking we must distinguish two absurd states that contain impossible information about the values of different sets of variables and I do so in this chapter. This amended treatment of absurd states does not interfere with the semantics of *EDPL* presented below except with respect to definedness phenomena, which turn out more perspicuous then in the former paper (cf., section 3.2). On the other hand, the refinement *is* crucial for the use that is made of the structure of information states in section 5.

If t is an update of s, then every assignment in t is an extension of some assignment in s. An update t of s contains only possible valuations of the variables in $D(s)$ which are possible in s. Hence, t contains at least the information that s contains about the variables in $D(s)$. Moreover, t may contain information about variables which s is silent about. The assignments in t may be proper extensions of assignments in s. So, the relation $\leq$ precisely captures the notion of update informally described in section 2.3.

The update relation $\leq$ induces a partial order of information states:

Fact 2.1 $\langle S, \leq \rangle$ is a partial order, i.e., $\leq$ is
- reflexive ($s \leq s$)
- transitive (if $s \leq s'$ and $s' \leq s''$ then $s \leq s''$)
- antisymmetric (if $s \leq s'$ and $s' \leq s$ then $s = s'$)

In fact, as we will see in section 5, the set of information states is a complete lattice.

The second generalization of the subset relation is the substate relation, which, in a sense, is the dual of the update relation. The substate relation plays a part, chiefly, in the semantics of *EDPL*, since it acts in the *EDPL* definition of the *DPL*-style (and *DRT*-style) notions of truth and entailment. The relation is defined as follows:

Definition 2.4 (Substate)
- s is a substate of t, $s \sqsubseteq t$, iff $D(s) \subseteq D(t)$ and $\forall i \in s$: $i \lessdot t$

State s is a substate of t iff all possibilities in s have an extension in t. This definition says that if $s \sqsubseteq t$ then s contains at least as much information as t about the variables in the domain of s. However, in contradistinction with the case in which s is an update of t, if s is a substate of t, t may contain information about variables which s is silent about. For s to be a substate of t, the crucial thing is that t does not reject valuations of the variables in $D(s)$ which are still possible in s.

The notions of update and substate are related to the notion of subset in the following way:

Fact 2.2
- If $D(s) = D(t)$, then $s \leq t$ iff $t \sqsubseteq s$ iff $t \subseteq s$
- $s \leq t$ and $t \sqsubseteq s$ iff $t \subseteq s$

Like the update relation, the substate relation is a partial order:

Fact 2.3 $\langle S, \sqsubseteq \rangle$ is a partial order

As concerns the ordering of S by $\sqsubseteq$ and $\leq$ I note the following. The weakest information state in the sense of $\sqsubseteq$ (i.e., the state t such that $\forall s$: $s \sqsubseteq t$) is $\top^V$, the state with no information about all variables. The strongest state in $\sqsubseteq$ is $\bot^\emptyset$, the absurd state of information about no variables ($\forall s$: $\bot^{\emptyset} \sqsubseteq s$). The strongest state in the sense of $\leq$ is $\bot^V$, the absurd state of information about all variables ($\forall s$: $s \leq \bot^V$). The weakest information state in $\leq$ (i.e., the state t such that $\forall s$: $s \leq t$) is $\top^{\emptyset}$, the state with no information about no variables. These observations can be pictured as follows (the ordering relations $\sqsubseteq$ and $\leq$ should be read as if rotated 90^o anticlockwise):

$$\sqsubseteq \quad \begin{array}{c} \top^V \\ \vdots \\ \bot^{\emptyset} \end{array} \qquad\qquad \leq \quad \begin{array}{c} \bot^V \\ \vdots \\ \top^{\emptyset} \end{array}$$

Notice that $\sqsubseteq$ ($\leq$) ranges from the bottom (top) of propositional logic to the top (bottom) of predicate logic.

2.2 Semantics of EDPL

We are now ready to turn to the semantics of *EDPL*. Like a *DPL* model, an *EDPL* model is a pair $M = \langle D, F \rangle$ consisting of a non-empty set of individuals D and an interpretation function F that assigns sets of n-tuples of objects to n-ary relation expressions. (Again, I will omit reference to M whenever this does not lead to confusion.) In the definition of the semantics of *EDPL* the following abbreviations are used ($i \leq_X j$ iff $j \in D^{D(i) \cup X}$ and $i \leq j$):

Definition 2.5 (State subtraction and domain extension)

- $s - t = \{i \in s \mid i \not\leq t\}$
- $s[x] \ = \{j \mid \exists i \in s\text{: } i \leq_{\{x\}} j\}$

Subtracting state t from state s we get all those assignments in s which do not have an extension in t. If the domain of s is a subset of the domain of t, which it is in all cases of state subtraction below, then $s - t$ is the set of assignments in s to the variables in the domain of s which are excluded by t. So, state $s - t$ contains the information that s has about the variables in the domain of s supplemented with the information excluded by t.

The state $s[x]$ at most differs from s in that it is defined for x. About the values of variables in the domain of s the new state $s[x]$ contains precisely the same information as s, and, if x is not in the domain of s, $s[x]$ is completely impartial about the value of x. In that case, for each i in s and for each z in D, there is an extension j of i in $s[x]$ that assigns z to x. What is added, one might say, is the information that x has a value. Clearly, if x is already in the domain of s, then $s[x] = s$. In the semantics of *EDPL*, however, it is excluded that $[x]$ operates on states s which are already defined for x (but, cf., section 5.2).

EDPL interpretation is defined as a *partial* update function on the domain of information states. If a formula contains a free variable x, it must be interpreted in a state that is defined for x. Conversely, for an indefinite noun phrase (existential quantifier) to extend the domain of a state s with a new variable x, x should not be in the domain of s.[3] In the following definition I only indicate the possibility of undefinedness in these so-called 'source' cases. Of course, since the interpretation of a compound formula is stated in terms of that of its constituent formulas, the interpretation of the compound must be understood to be undefined if the interpretation of one of its constituents is undefined. I will come back to this below.

The definition of the semantics of *EDPL* runs as follows:

Definition 2.6 (Semantics of EDPL)

- $s[\![Rx_1 \ldots x_n]\!] = \{i \in s \mid \langle i(x_1), \ldots, i(x_n)\rangle \in F(R)\}$ if $x_1, \ldots, x_n \in D(s)$
- $s[\![x = y]\!] = \{i \in s \mid i(x) = i(y)\}$ if $x, y \in D(s)$
- $s[\![\neg\phi]\!] = s - s[\![\phi]\!]$
- $s[\![\phi \wedge \psi]\!] = s[\![\phi]\!][\![\psi]\!]$
- $s[\![\exists x\phi]\!] = s[x][\![\phi]\!]$ if $x \notin D(s)$

The clauses in this definition correspond closely to those in the definitions of the semantics of *DPL* and *US*. The interpretation of atomic formulas is the same as in *DPL*, except for the use of partial, instead of total assignments. The interpretation of $Rx_1 \ldots x_n$ in s preserves the assignments in s that map the variables $x_1, \ldots, x_n$ onto individuals $z_1, \ldots, z_n$ that stand in the relation R, in that order. *EDPL* negation combines features of negation in *DPL* and in *US*. The requirement that assignments i in $s[\![\neg\phi]\!]$ do not have an extension in $s[\![\phi]\!]$ in fact corresponds to the *DPL* requirement that $\{i\}[\![\phi]\!]$ be empty. On the other hand, the result of interpreting $\neg\phi$ in s is defined solely in terms of s and $s[\![\phi]\!]$, like in *US*. Like in *DPL* and *US*, conjunction is interpreted as function composition. Furthermore, like in *DPL*, the existential quantifier $\exists x$ introduces arbitrary valuations of x, the difference being that domain extension is used, instead of reinstantiation. In section 3.3 below, it will be demonstrated in more detail how these clauses work out in practice. Before that, I discuss some general properties of *EDPL* interpretation in the remainder of this section.

Definedness conditions in EDPL

A major difference with *DPL*, and *US*, is the presence of side conditions, or definedness conditions, on *EDPL* interpretation. As was already said above, the interpretation of a formula can be undefined for certain states.

Undefinedness is generated by free variable and quantifier occurrences. If a formula contains a free variable for which a state s is undefined, then the interpre-

3. In *DRT*, for this reason, the so-called discourse representation algorithm is used to ensure that indefinites always associate with novel variables and pronouns with yet established ones, and in *FCS* felicity-conditions serve the same purpose. *EDPL* makes the same requirements on pain of undefinedness. In section 5 we will see that these requirements can be relaxed in certain cases.

tation of the formula is undefined for that state s; similarly, the interpretation of an existentially quantified formula $\exists x\phi$ is undefined for a state s which is already defined for x. In particular this last requirement expels the possibility of reinstantiation. Undefinedness persists in the following way. If (the interpretation of) ϕ is undefined for s, then $\neg\phi$ and $\phi \wedge \psi$ are undefined for s. Furthermore, if ϕ is undefined for $s[x]$, then $\exists x\phi$ is undefined for s, and if ψ is undefined for $s[\![\phi]\!]$, then $\phi \wedge \psi$ is undefined for s. Notice that these restrictions on the states for which formulas are defined are, basically, Heim's felicity conditions in *FCS*, reformulated here as definedness conditions. In fact, the semantics of *EDPL* closely resembles the version of *FCS* in Heim [1983].

In section 5 we will see that the side conditions on *EDPL* interpretation can be removed without allowing sentences to involve loss of information, and we will also find reason to do so in certain cases. This will be discussed in due course. However, for the present purpose of merely giving an update reformulation of *DPL*, it is expedient to adopt the side conditions and to keep to the partial interpretation function specified above.

The question whether the interpretation of an *EDPL* formula is defined for a state s completely depends on the domain of s. In other words, *EDPL* formulas carry presuppositions that restrict the domains of states with respect to which their interpretation is defined. These presuppositions can be calculated by the following function $\mathcal{D}$, which for any formula ϕ defines the domains of information states for which ϕ is defined, or, as I will also say, the domains of ϕ:

Definition 2.7 (Domains of a formula)

- $\mathcal{D}(Rx_1 \ldots x_n) \quad = \{X \mid x_1, \ldots, x_n \in X\}$
- $\mathcal{D}(\exists x\phi) \quad = \{X \mid x \notin X \text{ and } X \cup \{x\} \in \mathcal{D}(\phi)\}$
- $\mathcal{D}(\neg\phi) \quad = \mathcal{D}(\phi)$
- $\mathcal{D}(\phi \wedge \psi) \quad = \mathcal{D}(\phi) \cap \mathcal{D}(\psi)$ if ϕ is a test
  $\mathcal{D}(\exists x\phi \wedge \psi) \quad = \mathcal{D}(\exists x(\phi \wedge \psi))$
  $\mathcal{D}((\phi \wedge \psi) \wedge \chi) = \mathcal{D}(\phi \wedge (\psi \wedge \chi))$

A formula ϕ is a test if ϕ is an atomic formula $Rx_1 \ldots x_n$ or $x = y$ or a negation $\neg\psi$.

For any formula ϕ, $\mathcal{D}(\phi)$ is a set of sets of variables, i.e., it is a generalized quantifier over variables. Moreover, $\mathcal{D}(\phi)$ is, typically, a quantifier with the following properties:

Fact 2.4

- For any formula ϕ, $\mathcal{D}(\phi)$ is a continuous quantifier closed under $\cap$ and $\cup$

(A quantifier $\mathcal{Q}$ is continuous iff for any X and Y, if $X \in \mathcal{Q}$ and $Y \in \mathcal{Q}$, then for any Z, if $X \subseteq Z \subseteq Y$ then $Z \in \mathcal{Q}$.) Since $\mathcal{D}(\phi)$ is continuous and closed under $\cap$ and $\cup$, there are sets of variables X and Y such that $\mathcal{D}(\phi) = \{Z \mid X \subseteq Z \subseteq Y\}$. So, if $\mathcal{D}(\phi)$ is not empty, it has a smallest element X and a greatest element Y and all elements bigger than X and smaller than Y.

If $\mathcal{D}(\phi)$ is not empty, then the smallest element X in the domains $\mathcal{D}(\phi)$ of ϕ is the set of free variables in ϕ and the greatest element is the complement of the set of variables quantified over in ϕ. In that case, if we let A abbreviate *free variable in* ϕ, and B *variable quantified over in* ϕ, then $\mathcal{D}(\phi)$ is denoted by the noun phrase *Every* A *and no* B. (Notice that $\mathcal{D}(\phi)$ may be empty, even if the denotation of *Every* A *and no* B is not. This is the case if ϕ contains an existential quantifier to the right of a still 'active' quantifier ranging over the same variable x in ϕ.)

The next fact shows that $\mathcal{D}$ is defined properly:

Fact 2.5

- $D(s) \in \mathcal{D}(\phi)$ iff $s[\![\phi]\!]$ is defined

So, on the basis of syntactic properties of ϕ we can compute the function $\mathcal{D}(\phi)$ which tells us for which states ϕ is defined. Furthermore, if $\mathcal{D}(\phi)$ is non-empty, then we find that ϕ is defined for s iff every free variable in ϕ and no variable quantified over in ϕ is in the domain of s.

Update, truth and distributivity

The most important property of *EDPL* is that it is a genuine update semantics. Interpretation in *EDPL* always yields update of information:

Fact 2.6 (Update)

- $s \leq s[\![\phi]\!]$, if defined

(Fact 2.6 is proved by a simple induction on the complexity of ϕ.) The fact that the interpretation of any sentence in *EDPL* is an update function distinguishes *EDPL* from *DPL*. In *EDPL* no established information about the values of variables gets lost in the process of interpretation. The worst thing that can happen, one might say, is that an attempt to initialize a variable which one has already information about is encountered by an error message.

Like *DPL*, *EDPL* is not really eliminative. Update of information in *EDPL* not only consists in getting more informed about the values of variables one has already information about, but it may also consist in getting informed about the values of more variables. Hence, *EDPL* interpretation not merely involves elimination of assignments, it may also involve extension of assignments. However, as is expressed by fact 2.6, even if the assignments in the state that results from updating a state s with ϕ are not themselves elements of the original state s, as is the case in a real eliminative semantics, all these assignments are at least extensions of assignments in s.

The *DPL* definition of truth builds on fact 2.6 (the definition of entailment is postponed to section 3.4):

Definition 2.8 (Truth in EDPL)

- ϕ is true in s with respect to M, $s \models_M \phi$, iff $s \sqsubseteq s[\![\phi]\!]_M$
  ϕ is false in s with respect to M, $s =\!\!|_M \phi$, iff $s[\![\phi]\!]_M = \emptyset$

A formula ϕ is true in a state s iff s is a substate of the update of s with ϕ, i.e., iff the state $s[\![\phi]\!]$ does not contain *more* information than s about the variables in the domain of s. Notice that for the assessment of the truth of a formula ϕ in a state s we only need to compare the states s and $s[\![\phi]\!]$. Since *EDPL* interpretation always produces information update it is possible to define a proper notion of truth in this way. It is excluded that ϕ is true in a state s while in the processing of ϕ in s we acquire information not yet present in s and which is discarded afterwards. Given fact 2.6, such information downdate is simply excluded.

Clearly, a formula is true in a state s iff its negation is false in s, and vice versa. However, if a formula is defined for a state s, it need not be either true or false in s. Since we are dealing with partial information in *EDPL*, a formula may convey information about the values of variables that a state s does not yet have, but that is neither excluded by s. On the other hand, in case we are dealing with a maximal state of information $\{i\}$, then a formula is either true or false in $\{i\}$, that is, if it is defined for $\{i\}$. If we have that maximal state $\{i\} \models \phi$, then I will also say that i satisfies ϕ.

Like *DPL*, *EDPL* has a distributive semantics:

Fact 2.7 (Distributivity)

- $s[\![\phi]\!] = \bigcup_{i \in s}\{i\}[\![\phi]\!]$, if defined

(The fact is again proved by a straightforward induction on the complexity of ϕ.) In section 2.3, footnote 1, it is noticed that an eliminative and distributive dynamic semantics is equivalent to a static semantics, since an eliminative and distributive update of a state s corresponds to the intersection of s with a characteristic set. The fact that *EDPL*, besides distributivity, only licenses update (and not pure eliminativity) prevents it from collapsing into a static semantics.

Update and distributivity jointly do entail the following fact:

Fact 2.8

For all s, $i \in s$:

- $\{i\} \models \phi$ iff $i \lessdot s[\![\phi]\!]$

Fact 2.8 tells us that an assignment i in s satisfies ϕ iff i survives in the update of s with ϕ. So, the update of a state s with ϕ contains assignments that register, i.e., extend, the assignments in s that satisfy ϕ. Furthermore, by fact 2.6, $s[\![\phi]\!]$ contains *only* assignments which register assignments in s that satisfy ϕ. This fact shows, once more, the crucial difference with interpretation in *DPL*, where we can not in general conclude, considering the assignments in $s[\![\phi]\!]$, which assignments in s satisfy

ϕ.

With respect to the *EDPL* notion of truth, fact 2.8 tells us that ϕ is true in s iff all assignments i in s satisfy ϕ: $s \models \phi$ iff (definition of $\models$) $s \sqsubseteq s[\![\phi]\!]$ iff (definition of $\sqsubseteq$) $\forall i \in s$: $i < s[\![\phi]\!]$ iff (fact 2.8) $\forall i \in s$: $\{i\} \models \phi$. In this respect the notion of truth in *EDPL* is exactly like that in *DPL*, where $s \models \phi$ iff $\forall i \in s$: $\{i\}[\![\phi]\!] \neq \emptyset$. The difference with *DPL* is that the truth of ϕ in a state s in *EDPL* is defined, basically, in terms of s and $s[\![\phi]\!]$ only, whereas in *DPL* it is, crucially, defined in terms of s and the distribution of $[\![\phi]\!]$ over the singleton subsets of s.

2.3 Applications of EDPL

I will now turn to the more specific characteristics of *EDPL* interpretation. In this section I will simply assume definedness whenever this is unlikely to give rise to confusion.

Existential quantification

Since, as in *DPL*, the interpretation of an existentially quantified formula in *EDPL*, and that of a conjunction, involves the composition of two operations on states, *DPL*'s characteristic donkey equivalences are retained:

Fact 2.9 (Donkey equivalences (1))

- $(\exists x\phi \wedge \psi) \Leftrightarrow (\exists x(\phi \wedge \psi))$
- $((\phi \wedge \psi) \wedge \chi) \Leftrightarrow (\phi \wedge (\psi \wedge \chi))$

Like *DPL*, *EDPL* accounts for the fact that indefinite noun phrases (existential quantifiers) in one sentence may bind pronouns (free variables) in a successive sentence. Again, this is achieved in a completely compositional way. Both sentences *A man owns a donkey* (translated as $\exists x(Mx \wedge \exists y(Dy \wedge Oxy)))$ and *He beats it* (translated as Bxy) get assigned an interpretation of their own, and in the conjunction of the two into *A man owns a donkey. He beats it* $(\exists x(Mx \wedge \exists y(Dy \wedge Oxy)) \wedge Bxy)$, the respective anaphoric relationships get established.

Let us inspect the donkey example in a little more detail. The interpretation of $\exists x(Mx \wedge \exists y(Dy \wedge Oxy))$ in a state s produces the state $s[\![\exists x(Mx \wedge \exists y(Dy \wedge Oxy))]\!]$, which is $s[x][\![Mx]\!][y][\![Dy]\!][\![Oxy]\!]$. If X is the domain of s, and $Y = X \cup \{x, y\}$, this is the following set of assignments:

(9) $\{j \in D^Y \mid \exists i \in s\text{: } i \leq j \ \&\ j(x) \in F(M)\ \&$
$\qquad j(y) \in F(D)\ \&\ \langle j(x), j(y)\rangle \in F(O)\}$

This state, which will be referred to as state t for the time being, contains the information that the value of x is a man and the value of y is a donkey and that the value of x owns the value of y. Notice that for any $i \in s$ and for any man z and donkey z' owned by z, there is an assignment $j \in t$ such that $i \leq j$, $j(x) = z$ and $j(y) = z'$.

The formula $\exists x(Mx \wedge \exists y(Dy \wedge Oxy))$ is true in s iff $s \sqsubseteq t$, that is, iff, in the model, there is a man who owns a donkey. Notice that the formula $\exists x(Mx \wedge \exists y(Dy \wedge Oxy))$ can only be true or false in a state s. The formula contains no free variables, and, hence, conveys no information about the variables in the domain of s.

Now consider the update of t with Bxy, $t[\![Bxy]\!]$, which is the following set of assignments:

(10) $\{j \in D^Y \mid \exists i \in s\text{: } i \leq j \;\&\; j(x) \in F(M) \;\&$
$j(y) \in F(D) \;\&\; \langle j(x), j(y)\rangle \in F(O) \;\&\; \langle j(x), j(y)\rangle \in F(B)\}$

The state $t[\![Bxy]\!]$ contains the information that the value of x is a man and the value of y is a donkey and that the value of x owns and beats the value of y. The little discourse $\exists x(Mx \wedge \exists y(Dy \wedge Oxy)) \wedge Bxy$ then is true in s iff $s \sqsubseteq t[\![Bxy]\!]$, that is, iff, in the model, there is a man who owns and beats a donkey.

Notice that the formula Bxy is not in general true in a state t as described above. If someone has the information that the value of x is a man and the value of y a donkey the value of x owns, he need not have the information that the value of x beats the value of y of course. This would only be the case if he already had the information that if a man owns a donkey, he beats it.

In *EDPL* implication and universal quantification are defined, as usual, in terms of existential quantification, negation and conjunction:

Definition 2.9

- $\phi \rightarrow \psi = \neg(\phi \wedge \neg\psi)$
- $\forall x\phi = \neg\exists x\neg\phi$

Like *DPL*, as a corollary of the first donkey equivalences *EDPL* validates the second donkey equivalences:

Fact 2.10 (Donkey equivalences (2))

- $(\exists x\phi \rightarrow \psi) \Leftrightarrow (\forall x(\phi \rightarrow \psi))$
- $((\phi \wedge \psi) \rightarrow \chi) \Leftrightarrow (\phi \rightarrow (\psi \rightarrow \chi))$

So, the sentence *If a man owns a donkey he beats it* ($\exists x(Mx \wedge \exists y(Dy \wedge Oxy)) \rightarrow Bxy$) turns out equivalent with the sentence *Every man beats every donkey he owns* ($\forall x(Mx \rightarrow \forall y((Dy \wedge Oxy) \rightarrow Bxy))$). Before I show in more detail how examples like these are interpreted, it is expedient to take a closer look at *EDPL* negation.

Negation, universal quantification and implication

Like in *US*, the interpretation of $\neg\phi$ in a state s involves a form of 'state subtraction', the subtraction of state $s[\![\phi]\!]$ from state s. The subtraction of $s[\![\phi]\!]$ from s in *EDPL* involves the elimination of assignments in s that have an extension in $s[\![\phi]\!]$. The difference with ordinary set subtraction, which is employed in *US* negation, is that

$s - s[\![\phi]\!]$ in *EDPL* is not the set of assignments in s which are not *an element of* $s[\![\phi]\!]$, but the set of assignments in s which do not *survive in* $s[\![\phi]\!]$.

Fact 2.8 tells us that the interpretation of $\neg\phi$ in s in fact involves the elimination of assignments in s that satisfy ϕ. This is like in *DPL*. In *DPL*, $s[\![\neg\phi]\!]$ is the set $\{i \in s \mid \{i\} \not\models \phi\}$. In *EDPL*, $s[\![\neg\phi]\!]$ is the set $\{i \in s \mid i \not< s[\![\phi]\!]\}$, which, by fact 2.8, also is $\{i \in s \mid \{i\} \not\models \phi\}$. The difference with *DPL* is that $s[\![\neg\phi]\!]$ in *EDPL* is defined in terms of s and the update of s with ϕ solely, whereas in *DPL* $s[\![\neg\phi]\!]$ is defined in terms of s and the distribution of $[\![\phi]\!]$ over the singleton subsets of s.

Let us consider one simple example:

(11) No man sees her.
$\neg\exists y(My \wedge Syx)$

The interpretation of this sentence in a state s yields the state $s[\![\neg\exists y(My \wedge Syx)]\!]$, which is $s - s[y][\![My]\!][\![Syx]\!]$. The state $s[y][\![My]\!][\![Syx]\!]$ that is subtracted from the original state s is the following set of assignments (where X is the domain of s):

(12) $\{j \in D^{X \cup \{y\}} \mid \exists i \in s\text{: } i \leq_{\{y\}} j \;\&\; j(y) \in F(M) \;\&\; \langle j(y), j(x)\rangle \in F(S)\}$

The subtraction of this state from s, $s - s[y][\![My]\!][\![Syx]\!]$, produces a state that consists of the assignments in s that do not survive in $s[y][\![My]\!][\![Syx]\!]$:

(13) $\{i \in s \mid \neg\exists j\text{: } i \leq_{\{y\}} j \;\&\; j(y) \in F(M) \;\&\; \langle j(y), j(x)\rangle \in F(S)\}$

In other words, the interpretation of $\neg\exists y(My \wedge Syx)$ in s produces a state that contains the information that the value of x is an individual such that no other individual can be found that is a man and sees her.

Since the interpretation of a formula $\neg\phi$ in a state s in *EDPL* is purely eliminative it does not bring about an extension of the domain of s. This corresponds to the fact that, usually, if a noun phrase stands in the scope of a negation it can not serve as an antecedent for pronouns that occur outside the negation's scope (but, for exceptions, see Groenendijk and Stokhof [1990a] and chapter 2). As a consequence, the law of double negation doesn't hold in *EDPL*. The double negation of a formula involves a form of state restriction:

Fact 2.11

- $s[\![\neg\neg\phi]\!] = s - (s - s[\![\phi]\!]) = \{i \in s \mid i < s[\![\phi]\!]\}$

The interpretation of the double negation of ϕ in s preserves the assignments in s which survive the update of s with ϕ. The state $s[\![\neg\neg\phi]\!]$ consists of the assignments in s which have an extension in $s[\![\phi]\!]$, not the extensions themselves. So, $\neg\neg\phi$ imposes the same restrictions as ϕ imposes on the values of variables in the domain of a state s, but it cancels possible extensions of the domain of s brought about by ϕ.

Something essentially similar holds in *DPL*, where the double negation of a formula ϕ involves the same test on assignments in a state s as ϕ involves, but discards the reinstantiation of variables induced by ϕ. Since the double negation of

ϕ thus robs it of its context change potential, Groenendijk and Stokhof call it the static closure of ϕ. In *EDPL* I also adopt this terminology and, like Groenendijk and Stokhof, write $\downarrow\phi$ for the static closure $\neg\neg\phi$ of ϕ.

Given the definition of universal quantification and implication above, the semantics of $\forall x$ and $\rightarrow$ turns out to be as follows:

Fact 2.12

- $s[\![\forall x\phi]\!] = s - (s[x] - s[x][\![\phi]\!])$
  $= \{i \in s \mid \forall j \in s[x]\text{: if } i \leq j \text{ then } j \prec s[x][\![\phi]\!]\}$
- $s[\![\phi \rightarrow \psi]\!] = s - (s[\![\phi]\!] - s[\![\phi]\!][\![\psi]\!])$
  $= \{i \in s \mid \forall j \in s[\![\phi]\!]\text{: if } i \leq j \text{ then } j \prec s[\![\phi]\!][\![\psi]\!]\}$
- $s[\![\forall x(\phi \rightarrow \psi)]\!] = s - (s[x][\![\phi]\!] - s[x][\![\phi]\!][\![\psi]\!])$
  $= \{i \in s \mid \forall j \in s[x][\![\phi]\!]\text{: if } i \leq j \text{ then } j \prec s[x][\![\phi]\!][\![\psi]\!]\}$

The interpretation of $\forall x\phi$ in a state s preserves all the assignments i in s such that every extension of i to x survives in the update with ϕ. So, using fact 2.8, for any assignment i in $s[\![\forall x\phi]\!]$ it holds that every extension of i to x satisfies ϕ.

Similarly, the interpretation of $(\phi \rightarrow \psi)$ preserves the assignments i in s of which every extension in $s[\![\phi]\!]$ survives in the subsequent update with ψ. The interpretation of $\forall x(\phi \rightarrow \psi)$ in a state s preserves the assignments i in s every extension of which in $s[x][\![\phi]\!]$ has an extension in $s[x][\![\phi]\!][\![\psi]\!]$. Using fact 2.8 again, for any assignment i in $s[\![\forall x(\phi \rightarrow \psi)]\!]$ it holds that every extension of i in $s[x][\![\phi]\!]$ satisfies ψ.

Let us briefly consider two examples of universal quantification, the first one of which contains a free variable:

(14) Every man sees her.
$\forall y(My \rightarrow Syx)$

The interpretation of this sentence in a state s yields the state $s[\![\forall y(My \rightarrow Syx)]\!]$, which is $s - (s[y][\![My]\!] - s[y][\![My]\!][\![Syx]\!])$. This state consists of the assignments i in s such that on every extension of i to y such that the value of y is a man, the value of y sees the values of x. So, the resulting state consists of those assignments i in s which assign an individual to x which every man sees.

The second example is the universally quantified donkey sentence:

(15) Every man who owns a donkey beats it.
$\forall x((Mx \wedge \exists y(Dy \wedge Oxy)) \rightarrow Bxy)$

The interpretation of this sentence in a state s yields the set of assignments i in s such that every extension of i in $s[x][\![Fx]\!][y][\![Dy]\!][\![Oxy]\!]$ satisfies Bxy. So, the resulting state contains an assignment i in s iff on every extension j of i to x and y it holds that if the value of x is a man and the value of y is a donkey which the value of x owns, then the value of x beats the value of y. Clearly, this simply requires that every man beats every donkey he owns.

Digression

It has been argued, for instance by Schubert and Pelletier [1988], that the strong readings of donkey sentences are misguided, or, at least, are not the only readings these sentences have. Schubert and Pelletier's favourite example is the following sentence, which I label the 'dime implication':

(16) If I have a dime in my pocket, I'll put it in the parking meter.

On its most natural reading this sentence says that if I have one or more dimes in my pocket, then I will throw one in the meter. However, if we interpret the dime implication, like the donkey implication, as one of strong implication, then the sentence would imply that I throw *all* the dimes I have in my pocket in the meter. As concerns the present example, this strong reading seems quite odd.

It is possible to define a notion of weak implication that assigns conditional sentences the weak truth conditions that Schubert and Pelletier argue for, and that preserves the internal dynamics of the implication. This is, in fact, Chierchia's [1992] definition of implication. I use $\hookrightarrow$ to indicate the weak notion of implication:

Definition 2.10 (Weak implication)

- $\phi \hookrightarrow \psi = {\downarrow}\phi \rightarrow (\phi \wedge \psi)$

The interpretation of a weak implication $\phi \hookrightarrow \psi$ in a state s gives us those $i \in s$ such that if i has an extension in $s[\![\phi]\!]$, then i has an extension in $s[\![\phi]\!][\![\psi]\!]$. If we interpret the dime implication employing this weak notion of implication, then the sentence is true in a state s if I throw a dime I have in my pocket in the meter, that is, if I have dimes in my pocket at all.

The contrast between the strong donkey and the weak dime implications constitutes an argument, although some have argued that the strong donkey should be read weakly. Now this is not the place to discuss in detail whether natural language conditionals should be considered inherently ambiguous or whether we have to choose for one of the two notions of implication. Still, unlike Schubert and Pelletier, I think that it would be misguided to unconditionally reject any one of the two notions. Both the strong and the weak implication seem reasonable in certain contexts.

Convincing arguments for the viability of both readings are obtained by slight variations of apparent weak implications, in which their consequent clauses are negated. For instance, consider the following variant of the dime implication:

(17) If I have a dime in my pocket, I won't throw it in the meter.

Interpreting this sentence as one of weak implication seems to give just as odd results as the strong interpretation of the original dime sentence. On the weak reading of the present sentence, its truth conditions would only require that if I have one or more dimes in my pocket, I keep at least one dime but maybe throw all other dimes I have in the parking meter. On the other hand, on the strong reading it is required that I don't throw any dime I might have in my pocket in the meter, and that, I think, is as intuition would have it.

What makes this alternative dime sentence an interesting example is that it completely corresponds to the original dime sentence, that it can be uttered in precisely the same contexts as the original dime sentence, but that it, nevertheless, seems to require the strong interpretation of the conditional, whereas the original dime sentence requires the weak interpretation. In my opinion this supplies strong evidence against Schubert and Pelletier's claim about the strong reading of conditionals, that "What makes it seem correct in certain cases is its confounding with a 'generic' or 'habitual' or 'gnomic' understanding of these sentences" ([1988, p. 201]). There hardly seems to be any reason for assuming that the alternative dime implication (for which the strong reading has to be preferred) is in any sense more 'generic', 'habitual' or 'gnomic' then the original dime implication (the weak reading of which is preferred).

In section 4 I will come back to this issue, and there I will show that the weak and strong readings of conditionals in fact fit in a more general scheme of (universal) adverbial quantification. Up untill then, I will only use the strong implication ($\rightarrow$), which is defined in the usual way in terms of negation and conjunction, and which naturally corresponds to the notion of entailment given below.

This concludes the digression.

2.4 Entailment, DPL and EDPL

I conclude this section about *EDPL*'s semantics with a discussion about entailment and about the relation between *DPL* and *EDPL*.

EDPL entailment is relativized to a domain of variables X. A sequence of premises is defined to entail a conclusion with respect to a domain X, if the update with the premises of any state of information about the values of X yields a state in which the conclusion is true (the definition of truth is repeated for convenience):

Definition 2.11 (Truth and entailment in EDPL)

- $s \models_M \phi$, iff $s \sqsubseteq s[\![\phi]\!]_M$
- $\phi_1, \ldots, \phi_n \models_X \psi$ iff $\forall M, s \in S^X$: $s[\![\phi_1]\!]_M \ldots [\![\phi_n]\!]_M \models_M \psi$

If we, for the moment, pass over the restriction to states with a specific domain, the *EDPL* notion of entailment combines features of *DPL* and *US* entailment. *EDPL* entailment has the dynamics of *DPL* entailment. Free variables in the conclusion of an inference may refer back to objects introduced in the premises. So, like in *DPL*, we find that $\exists xFx \rightarrow Gx, \exists xFx \models Gx$, as in *If a man comes from Rhodes, he likes pineapple-juice. A man I met yesterday comes from Rhodes. So, he like pineapple-juice.* Like in *US*, *EDPL* entailment employs an update notion of truth. A sequence of premises entails a conclusion if, relative to a certain domain, the update with the premises of a state of information always produces a state containing information about variables about which subsequent update with the conclusion adds no more information.

The reason that entailment is defined relative to a domain of variables is that we should not exclude the possibility of undefinedness. Many valid inferences may be undefined for certain states of information and an inference may be valid even if, for some domain X, the conclusion of the inference is undefined for the state that results from the (defined) update of a state in S^X with the premises of the inference. For instance, for states with a domain Y such that $x \notin Y$, the interpretation of $x = x$ is undefined. However, we would want $x = x$ to be valid with respect to any domain X such that $x \in X$, and this indeed falls out of the present definition of entailment. Similarly, for all states $s \in S^Y$, if $x \notin Y$, then $\exists yFy$ is not necessarily defined in state $s[\![\exists xFx]\!]$. However, the entailment $\exists xFx \models_X \exists yFy$ should come out valid for any domain X such that $x, y \notin X$, and indeed it does.

So, disregarding undefinedness in some domains, an inference is judged valid iff (i) there is at least a domain in which the update with the premises and the conclusion of the inference is defined and (ii) the update with the premises of any state with such a domain produces a state in which the conclusion is true.[4]

Since *EDPL* inferences are relativized to a domain of variables, in order to assess whether an inference is valid relative to such a domain X, we only have to check for states s in S^X whether the conclusion is true in the update of s with the premises. However, it is important to notice that this relativization, or restriction, only discards occurring *undefinedness* of such inferences and that it does not corrupt the notion of entailment by discarding *counterexamples* to them. That is to say, if we have that $\phi \models_X \psi$, then there is no state s such that $s[\![\phi]\!][\![\psi]\!]$ is defined and $s[\![\phi]\!] \not\models \psi$. More generally we have the following fact:

Fact 2.13

- If $\phi_1, \ldots \phi_n \models_X \psi$, then for all s: $s[\![\phi_1]\!] \ldots [\![\phi_n]\!] \models \psi$, if defined

4. In order to allow inferences to be undefined in some domains, we might, alternatively, have required that every update with the premises, if defined, licenses the conclusion, if defined. However, such a (weaker) notion of entailment would generate inferences that owe their validity to being necessarily undefined, and this goes against the spirit of *EDPL*. For instance, in that case $x = x$ would entail $\exists x(x \neq x)$, which is objectionable.

In the proof of fact 2.13, I use the following lemma ($i \approx j$ says that for all x such that $i(x)$ and $j(x)$ are defined $i(x) = j(x)$):

Lemma 1

- If $i \approx j$ then $\forall k \in \{i\}[\![\phi]\!]\ \exists! l \in \{j\}[\![\phi]\!]$: $k \approx l$, if defined[5]

For reasons of readibility I show the proof of fact 2.13 for single premise entailments only, that is, I show that if $\phi \models_X \psi$ and $s[\![\phi]\!][\![\psi]\!]$ is defined, then $s[\![\phi]\!] \models \psi$, this by contraposition.
Suppose (i) $\phi \models_X \psi$, (ii) $s[\![\phi]\!][\![\psi]\!]$ is defined, and (iii) $s[\![\phi]\!] \not\models \psi$. By assumptions (ii) and (iii) there is an assignment j in $s[\![\phi]\!]$ such that $j \not\leq s[\![\phi]\!][\![\psi]\!]$ and (by update and distributivity) there is an assignment i in s such that $j \in \{i\}[\![\phi]\!]$ and $\{j\}[\![\psi]\!] = \emptyset$. Clearly, there is an assignment i' in D^X such that $i \approx i'$ and, using assumption (i), $\{i'\}[\![\phi]\!][\![\psi]\!]$ is defined. Using lemma 1 we find that there is an assignment j' in $\{i'\}[\![\phi]\!]$ such that $j \approx j'$, and, using lemma 1 again, that $\{j'\}[\![\psi]\!] = \emptyset$. But then $\{i'\}[\![\phi]\!] \not\models \psi$, which contradicts assumption (i). So, by contraposition, if $\phi \models_X \psi$ and $s[\![\phi]\!][\![\psi]\!]$ is defined, then $s[\![\phi]\!] \models \psi$. That concludes the proof.

Fact 2.13 states that if an inference is a valid entailment with respect to a domain of variables X, the inference is valid with respect to any domain, as long as it is defined. So, if $\phi_1, \ldots, \phi_n$ entail ψ with respect to a domain X, and if $\phi_1, \ldots, \phi_n, \psi$ is defined for states with domain Y, then $\phi_1, \ldots, \phi_n$ entail ψ with respect to domain Y. For this reason it is proper to say that $\phi_1, \ldots, \phi_n \models \psi$ iff there is a domain X such that $\phi_1, \ldots, \phi_n \models_X \psi$, and I will do so below.

5. This lemma is proved by induction on the complexity of ϕ. The proof is simplified by treating existentially quantified formulas as conjunctions: $\exists x\phi = \exists x \wedge \phi$ where $s[\![\exists x]\!]$ is $s[x]$ if $x \notin D(s)$. Atomic formulas and existential quantifiers then constitute the basic cases of the induction.

1. For atomic formulas the proof of the lemma is straightforward.
2. Suppose that $i \approx j$, that $\{i\}[\![\exists x]\!]$, $\{j\}[\![\exists x]\!]$ are defined and that $k \in \{i\}[\![\exists x]\!]$, i.e., $i \leq_{\{x\}} k$. Clearly, there is an assignment l in $\{j\}[\![\exists x]\!]$ (i.e., $j \leq_{\{x\}} l$) such that $k(x) = l(x)$. Since $i \approx j$, $i \leq_{\{x\}} k$, $j \leq_{\{x\}} l$ and $k(x) = l(x)$, we find that $k \approx l$. Now suppose that an assignment l' is in $\{j\}[\![\exists x]\!]$ and $k \approx l'$. Then $j \leq_{\{x\}} l'$ and $k(x) = l'(x)$ and, hence, $l' = l$.
3. Suppose that $i \approx j$, that $\{i\}[\![\neg\phi]\!], \{j\}[\![\neg\phi]\!]$ are defined and that $k \in \{i\}[\![\neg\phi]\!]$, i.e., $k = i$ and $\{i\}[\![\phi]\!] = \emptyset$. Using the induction hypothesis, $\{j\}[\![\phi]\!] = \emptyset$. So, $j \in \{j\}[\![\neg\phi]\!]$, and, since $i \approx j$ and $k = i$, $k \approx j$. Now suppose $j' \in \{j\}[\![\neg\phi]\!]$, then, automatically, $j' = j$.
4. Suppose that $i \approx j$, that $\{i\}[\![\phi\wedge\psi]\!], \{j\}[\![\phi\wedge\psi]\!]$ are defined and that $k \in \{i\}[\![\phi\wedge\psi]\!] = \{i\}[\![\phi]\!][\![\psi]\!]$. By distributivity there is an assignment f in $\{i\}[\![\phi]\!]$ such that $k \in \{f\}[\![\psi]\!]$. By induction there is an assignment g in $\{j\}[\![\phi]\!]$ such that $g \approx f$, and, again by induction, there is an assignment $l \in \{g\}[\![\psi]\!]$ such that $k \approx l$. By distributivity again, such an $l \approx k$ is in $\{j\}[\![\phi]\!][\![\psi]\!] = \{j\}[\![\phi\wedge\psi]\!]$. Now suppose that an assignment $l' \approx k$ is in $\{j\}[\![\phi]\!][\![\psi]\!]$. Then there is an assignment $g' \in \{j\}[\![\phi]\!]$ such that $l' \in \{g'\}[\![\psi]\!]$. Since (by update) $f \leq k$, $g \leq l$ and $g' \leq l'$, and since $l \approx k$ and $l' \approx k$, we find that $f \approx g$ and $f \approx g'$. Since $i \approx j$, by induction on ϕ, $g = g'$, and, by induction on ψ, $l = l'$.

Deduction and transitivity

The deduction theorem holds in *EDPL* :

Fact 2.14

- $\phi_1, \ldots, \phi_n \models \psi$ iff $\phi_1, \ldots, \phi_{n-1} \models \phi_n \to \psi$

This fact is proved in two steps.
(i) For any s, $s[\![\phi_1]\!] \ldots [\![\phi_n]\!] \models \psi$ iff (fact 2.8) $\forall j \in s[\![\phi_1]\!] \ldots [\![\phi_n]\!]$: $\{j\} \models \psi$ iff (fact 2.7) $\forall i \in s[\![\phi_1]\!] \ldots [\![\phi_{n-1}]\!]$ $\forall j \in \{i\}[\![\phi_n]\!]$: $\{j\} \models \psi$ iff (by definition of $\to$) $\forall i \in s[\![\phi_1]\!] \ldots [\![\phi_{n-1}]\!]$: $\{i\} \models (\phi_n \to \psi)$ iff (fact 2.8) $s[\![\phi_1]\!] \ldots [\![\phi_{n-1}]\!] \models (\phi_n \to \psi)$.
(ii) $\phi_1, \ldots, \phi_n \models \psi$ iff there is an X such that $\phi_1, \ldots, \phi_n \models_X \psi$ iff there is an X such that $\forall s \in S^X$: $s[\![\phi_1]\!] \ldots [\![\phi_n]\!] \models \psi$ iff (by (i)) there is an X such that $\forall s \in S^X$: $s[\![\phi_1]\!] \ldots [\![\phi_{n-1}]\!] \models (\phi_n \to \psi)$ iff there is an X such that $\phi_1, \ldots, \phi_{n-1} \models_X (\phi_n \to \psi)$. iff $\phi_1, \ldots, \phi_{n-1} \models (\phi_n \to \psi)$. That concludes the proof.

Like *DPL*-entailment *EDPL*-entailment is not transitive. The counterexample in *DPL* is also a counterexample in *EDPL*. Whereas $\exists xFx$ (as usual) entails $\exists yFy$ and $\exists yFy$ (dynamically) entails Fy, $\exists xFx$ does not entail Fy. Transitivity fails in this example because the goal conclusion Fy critically refers back to the object introduced in the mediating formula $\exists yFy$. However, if we restrict ourselves to inferences defined on a shared domain, transitivity does hold:

Fact 2.15 (Restricted transitivity)
If $\mathcal{D}(\phi \wedge \psi) \cap \mathcal{D}(\phi \wedge \chi) \neq \emptyset$ then:

- If $\phi \models \psi$ and $\psi \models \chi$, then $\phi \models \chi$

For the proof of fact 2.15, notice that, as a corollary of lemma 1, if $i \approx j$ and $\{i\}[\![\phi]\!]$, $\{j\}[\![\phi]\!]$ are defined, then $\forall k \in \{i\}[\![\phi]\!]$: $k \approx j$. (Proof. Suppose $i \approx j$, $\{j\}[\![\phi]\!]$ is defined and $k \in \{i\}[\![\phi]\!]$. By lemma 1, there is an assignment l in $\{j\}[\![\phi]\!]$ such that $k \approx l$. By fact 2.6, $j \leq l$, and, hence, $k \approx l$.)

The proof of fact 2.15 then runs as follows. Suppose (i) $X \in \mathcal{D}(\phi \wedge \psi) \cap \mathcal{D}(\phi \wedge \chi)$, (ii) $\phi \models \psi$ and (iii) $\psi \models_Y \chi$. Using assumptions (i) and (ii) and fact 2.13, $\phi \models_X \psi$. Now consider an assignment k in $s[\![\phi]\!]$, for any $s \in S^X$. Since $\phi \models_X \psi$, $\{k\} \models \psi$. Next consider an assignment k' in D^Y such that $k \approx k'$. By lemma 1 and assumption (iii), $\{k'\} \models \psi$ and for any assignment l in $\{k'\}[\![\psi]\!]$: $\{l\} \models \chi$. By the corollary of lemma 1, for any such l: $k \approx l$, hence (by lemma 1) $\{k\} \models \chi$, if defined. Since, by assumption (i) $\{k\}[\![\chi]\!]$ is defined, $\{k\} \models \chi$. This shows that for any $s \in S^X$ for any $k \in s[\![\phi]\!]$: $\{k\} \models \chi$, and, hence, that $s[\![\phi]\!] \models \chi$. So, $s[\![\phi]\!] \models_X \chi$, and, consequently, $s[\![\phi]\!] \models \chi$. That concludes the proof of fact 2.15.

Fact 2.15 implies that if ϕ entails ψ with respect to X, and ψ entails χ, then ϕ entails χ with respect to X iff defined relative to X. The restriction on transitivity effectively excludes cases where the mediating formula ψ introduces variables which are free in χ, and it also guarantees definedness of the goal inference from ϕ to χ

with respect to some domain X.

DPL and EDPL

I now turn to the relation between *EDPL* and *DPL*. Let s^V be the total extension of s, i.e., $s^V = \{g \in D^V \mid g \geqslant s\}$, and let t^X be the restriction of t to X, i.e., $t^X = \{i \in D^X \mid i \lessdot t\}$. Then truth in *DPL* with respect to a model M and truth in *EDPL* with respect to M are related in the following way:

Fact 2.16
If $t \in S^V$ and $X, D(s) \in \mathcal{D}(\phi)$, then
1. $t \models_{M,dpl} \phi$ iff $t^X \models_{M,edpl} \phi$
2. $s \models_{M,edpl} \phi$ iff $s^V \models_{M,dpl} \phi$

So, if ϕ is true in t in *DPL*, then ϕ is true in *EDPL* in the restriction of t to a domain for which ϕ is defined. Furthermore, if ϕ is true in s in *EDPL*, then ϕ is true in the total extension of s in *DPL*.

To prove fact 2.16 I use the following lemma ($g \geq_t i$ iff g is a total extension of i):

Lemma 2
If $\{i\}[\![\phi]\!]$ is defined and $g \geq_t i$, then
- $\{i\} \models_{M,edpl} \phi$ iff $\{g\} \models_{M,dpl} \phi$[6]

Using this lemma, the two clauses of fact 2.16 are proved as follows. With respect to the first clause, suppose $t \in S^V$ and $X \in \mathcal{D}(\phi)$, i.e., $t^X[\![\phi]\!]$ is defined. Then $t \models_{dpl} \phi$ iff (definition of $\models_{dpl}$) $\{g\} \models_{dpl} \phi$ for all g in t iff (lemma) $\{i\} \models_{edpl} \phi$ for all i in t^X iff (distributivity) $t^X \models_{edpl} \phi$.

With respect to the second clause of fact 2.16, suppose $D(s) \in \mathcal{D}(\phi)$, i.e., $s[\![\phi]\!]$ is defined. Then $s \models_{edpl} \phi$ iff (distributivity) $\{i\} \models_{edpl} \phi$ for all i in s iff (lemma) $\{g\} \models_{dpl} \phi$ for all $g \in s^V$ iff (definition of $\models_{dpl}$) $s^V \models_{dpl} \phi$. That concludes the proof.

---

6. The lemma is proved by induction on the complexitity of the normal binding form of ϕ (the normal binding form of a formula ϕ is obtained by substituting all occurences of subformulas of the form $(\exists x\phi_1 \wedge \phi_2)$ and $((\phi_1 \wedge \phi_2) \wedge \psi_3)$ in ϕ by $(\exists x(\phi_1 \wedge \phi_2))$ and $(\phi_1 \wedge (\phi_2 \wedge \psi_3))$ respectively):

– If $\{i\}[\![Rx_1 \dots x_n]\!]$ is defined and $g \geq_t i$, then $\{i\} \models_{M,edpl} Rx_1 \dots x_n$ iff $\{g\} \models_{M,dpl} Rx_1 \dots x_n$, since $i(x_1) = g(x_1)$, …, $i(x_n) = g(x_n)$ and $F_{edpl}(R) = F_{dpl}(R)$.

– If $\{i\}[\![\neg\phi]\!]$ is defined and $g \geq_t i$, then $\{i\}[\![\phi]\!]$ is defined and, hence, $\{i\} \models_{edpl} \neg\phi$ iff $\{i\} \dashv_{edpl} \phi$ iff (by induction) $\{g\} \dashv_{dpl} \phi$ iff $\{g\} \models_{dpl} \neg\phi$.

– If $\{i\}[\![\exists x\phi]\!]$ is defined and $g \geq_t i$, then $\forall j \geq_{\{x\}} i$: $\{j\}[\![\phi]\!]$ is defined. Hence, $\{i\} \models_{edpl} \exists x\phi$ iff $\exists j \geq_{\{x\}} i$: $\{j\} \models_{edpl} \phi$ iff $\exists h[x]g\ \exists j$: $h \geq_t j \geq_{\{x\}} i$ and $\{j\} \models_{edpl} \phi$ iff (by induction) $\exists h[x]g$: $\{h\} \models_{dpl} \phi$ iff $\{g\} \models_{dpl} \exists x\phi$.

– If $\{i\}[\![\phi \wedge \psi]\!]$ is defined and $g \geq_t i$, then (since ϕ is a test) $\{i\}[\![\phi]\!]$ and $\{i\}[\![\psi]\!]$ are defined and, furthermore, $\{i\} \models_{edpl} \phi \wedge \psi$ iff $\{i\} \models_{edpl} \phi$ and $\{i\} \models_{edpl} \psi$ iff (by induction) $\{g\} \models_{dpl} \phi$ and $\{g\} \models_{dpl} \psi$ iff $\{g\} \models_{dpl} \phi \wedge \psi$.

3 Quantification in EDPL

In this section I introduce quantifiers in *EDPL* and show that we can give a perspicuous and uniform interpretation of adnominal and adverbial quantifiers, symmetric as well as asymmetric. The quantifiers are assigned a so-called 'internally dynamic' interpretation, one that accounts for anaphoric relationships between antecedent noun phrases in the restriction of the quantifiers and anaphoric pronouns in their nuclear scope. I do not discuss 'externally dynamic' readings, in which antecedent noun phrases in the restriction or in the scope of a quantifier license anaphora beyond the scope of the quantifier (cf., for instance, van den Berg [1990, 1991] and chapter 2). I start with adverbs of quantification.

3.1 Adverbs of quantification (unselective)

Lewis [1975] argues that in many cases adverbs of quantification (like *always*, *sometimes*, *usually*) unselectively quantify over the values of 'free variables' (often stemming from indefinite noun phrases) in their restrictive clause. The examples Lewis discusses are of the form *Sometimes/usually/always if x is a man, if y is a donkey, and if x owns y, x beats y* and, clearly, these are paraphrases in the logicians idiom of sentences like:

(18) Sometimes/usually/always if a man owns a donkey, he beats it.

Lewis points out that the quantifying adverbs *sometimes/usually/always* in fact quantify over the 'cases' that verify the restrictive clause. These cases are the admissible assignments of values to the variables that are free in the restriction, or, equivalently, the tuples of individuals that are possible values of these variables. So, the cases that verify the restriction *x is a man, y is a donkey and x owns y* are the maps from x to a farmer and from y to a donkey the farmer owns (i.e., the pairs consisting of a farmer and a donkey he owns). The adverbial quantifier quantifies over these cases, i.e., pairs of individuals in the examples above. If the head is *sometimes*, as in *Sometimes, if a farmer owns a donkey, he beats it*, the sentence says that some pairs consisting of a farmer and a donkey owned, are pairs of which the first element beats the second. If *always* is the head, it is said that every pair of a farmer and a donkey he owns is a pair that stands in the beat relation. And with *usually*, we get that most pairs that consist of a farmer and a donkey owned, stand in the beat relation. (Clearly, if in a quantified construction $A(\phi)(\psi)$ the restriction ϕ contains three (or n) free variables or indefinites, the unselective adverbs quantify over triples (or n-tuples) of individuals.)

This part of the story about adverbs of quantification is formalized quite elegantly in *FCS*, *DRT* and *DPL*. In these systems, unselective quantification about the values of any number of free variables is cast in terms of quantification about verifying (output) assignments. In *DPL*, for instance, the formula *Always*$(\phi)(\psi)$ tests, given an initial assignment g, whether all assignments that verify ϕ with respect to g also satisfy ψ. Similarly, the formula *Sometimes*$(\phi)(\psi)$ tests whether some assignments that verify ϕ with respect to g satisfy ψ, and the formula *Never*$(\phi)(\psi)$ tests whether no assignment that verifies ϕ with respect to g satisfies ψ.

EDPL, too, allows a straightforward interpretation of unselectively quantifying adverbs. We restrict ourselves to the adverbs (and determiners, cf., section 4.3) that satisfy the constraints of extension, quantity and conservativity.[7] For any such adverb of quantification A, with its usual set-theoretic interpretation $[A]$, the interpretation is defined as follows:

Definition 3.1 (Adverbs of quantification (symmetric))

- $s[\![A(\phi)(\psi)]\!] = \{i \in s \mid [A](\{j \mid i \leq j \ \& \ j \in s[\![\phi]\!]\})(\{j \mid j \lessdot s[\![\phi]\!][\![\psi]\!]\})\}$

In a state s, for each assignment i in s, a symmetric adverbial quantifier A tests whether $[A]$ applies, first, to the set of extensions of i in the update of s with the restriction of the adverb, and, second, to the set of assignments that also verify the nuclear scope of the adverb. In effect, the adverb quantifies over the values of the variables introduced in the restriction.

So, if we interpret *If a farmer owns a donkey he always beats it* in a state s in *EDPL*, we get $s[\![Always(\exists x(Fx \wedge \exists y(Dy \wedge Oxy)))(Bxy)]\!]$. This is the set of assignments i in s such that on every extension of i to x and y, if the value of x is a farmer who owns a donkey which is the value of y, then the value of x beats the value of y. In other words, this formula tests whether all pairs of a farmer and a donkey he owns are pairs of which the first element beats the second element. A second example is *If a man gives her a present, she usually thanks him for it*, formalized as *Usually*$(\exists y(My \wedge \exists z(Pz \wedge Gyzx)))(Txyz)$. Interpreted in a state s, this example gives all those i in s that assign an individual to x that renders thanks in most cases in which a man gives her a present.

There are some interesting correspondences between the sentential connectives of *EDPL* and adverbs of quantification (as before, $\downarrow\phi$ indicates the static closure $\neg\neg\phi$ of ϕ):

Fact 3.1

- *Sometimes*$(\phi)(\psi) \Leftrightarrow \ \downarrow(\phi \wedge \psi)$

---

7. Let Q be a quantifier that assigns any domain of individuals E a binary relation Q_E between subsets of E. Then Q satisfies extension iff for all E, E' and for all $A, B \subseteq E \subseteq E'$: $Q_E(A)(B)$ iff $Q_{E'}(A)(B)$. Q satisfies quantity iff for all E, E', if π is a bijection from E to E', then for all $A, B \subseteq E$: $Q_E(A)(B)$ iff $Q_{E'}(\{\pi(a) \mid a \in A\})(\{\pi(b) \mid b \in B\})$. Q is conservative iff for all E and $A, B \subseteq E$, $Q_E(A)(B)$ iff $Q_E(A)(A \cap B$. Van Benthem [1986, Ch. 1,2] uses these constraints to single out the determiners that qualify as '(logical) quantifiers'. Adverbs and determiners that do not observe all three constraints can be treated within the *EDPL* framework, but they deserve a special treatment.

- *Always*$(\phi)(\psi)$ $\Leftrightarrow$ $(\phi \rightarrow \psi)$
- *Never*$(\phi)(\psi)$ $\Leftrightarrow$ $\neg(\phi \wedge \psi)$

Conjunction (disregarding its external dynamics) and implication appear to fit in the more general scheme of adverbial quantification. A sentence *Sometimes, if a farmer owns a donkey he beats it* has the same truth conditions as the conjunction *A farmer owns a donkey. He beats it.*[8] Furthermore, the sentence *Always, if a farmer owns a donkey he beats it* turns out equivalent with the sentence *Every farmer beats every donkey he owns*, and the sentence *If a farmer owns a donkey he never beats it* turns out equivalent with the sentence *No farmer who owns a donkey beats it*.

3.2 Adverbs of quantification (asymmetric)

Adverbs of quantification do not always unselectively quantify over the values of *all* variables introduced in their restriction. Several authors (Bäuerle and Egli [1985], Root [1986], Rooth [1987] and Kadmon [1987, 1990], see also Heim [1990] and Chierchia [1992]) have discussed examples in which adverbial quantifiers involve quantification over the values of a proper *subset* of the introduced variables. Following Rooth and Kadmon, I call this kind of quantification *asymmetric*. We find it in the following sentences:

(19) If a farmer owns a donkey, he is usually rich.
(20) If a DRUMMER lives in an apartment complex, it is usually half empty.
(21) If a drummer lives in an APARTMENT COMPLEX, it is usually half empty.

On its most natural reading, the adverb *usually* in the first example quantifies over farmers who own a donkey and not over farmer-donkey pairs. The sentence says that most farmers who own a donkey are rich. (If the adverb is taken to quantify unselectively, we get the different reading that for most pairs consisting of a farmer and a donkey he owns, it holds that the farmer is rich.) The second and third example are different for a similar reason. In the second example, with focal stress on *drummer*, we (may) find quantification over apartment complexes in which a drummer houses. The example then states that most apartment complexes where a drummer lives are usually half-empty. In the third example, where we find focal stress on *apartment complex*, the adverb may be taken to quantify over drummers. On this reading, the sentence says that most drummers that live in an apartment complex live in an half empty apartment complex.

---

8. Some features of the adverbially quantified sentence and of the indefinite donkey conjunction are neglected here. Intuitively, an adverbially quantified sentence *Sometimes*$(\phi)(\psi)$ requires more than one of the cases that verify ϕ to satisfy ψ, a plurality condition that is absent from the (singular) donkey conjunction. On the other hand, such a singular donkey conjunction may have a 'specific' flavour, in the sense that it may be taken to talk about a specific farmer and donkey known to the speaker. The adverbially quantified sentence lacks such a reading. In section 5.2 I will slightly address specific indefinites. Plurals fall beyond the scope of this chapter.

Examples like these pose a problem for the *DRT* and *DPL* analysis of adverbs of quantification, a problem dubbed the 'proportion problem' by Kadmon [1987]. Of course, it is easy to *annul* the introduction of certain variables in a restriction and to quantify over the values of the remaining variables. So, the sentence *If a man owns a donkey he is usually rich* can be (successfully) interpreted by means of the translation $\mathit{Usually}(\exists x(Mx \wedge \downarrow\exists y(Dy \wedge Oxy)))(Rx)$. However, this approach runs into problems when a pronoun in the nuclear scope is anaphorically related to an indefinite description in the restrictive clause that does not participate in the adverbial quantification. For instance, if we want the example *If a drummer lives in an* APARTMENT COMPLEX, *it is usually half empty* to quantify over drummers, and if, therefore, the interpretation of the restriction only yields assignments varying with respect to the drummers, then the pronoun *it* in the nuclear scope remains unbound.

Of course, in case of asymmetric quantification we don't really want to *annul* the introduction of certain variables in the restriction of a quantifying adverb. When the values of other variables are selected for quantification, we just want the adverb to *neglect* the (possibly different) values of the other ones. For this reason, Root [1986] suggests that asymmetric adverbial quantifiers do not discriminate between assignments that only differ from one another in their assignment to unselected variables. This implies that asymmetric quantifiers in fact quantify over equivalence classes of assignments, each one of which consists of assignments that agree on the values of the selected variables. For instance, in the example *If a drummer lives in an* APARTMENT COMPLEX, *it is usually half empty*, the adverb *usually* selects (the variable associated with) *a drummer*, and in effect quantifies over a set of sets of cases, each element of which is a set of pairs any first element of which is one and the same drummer and any second element any apartment complex the drummer lives in.

In *EDPL*, asymmetric adverbs more naturally fit into the general scheme of adverbial quantification. Since *EDPL* has the update property, we can (unselectively) quantify over the assignments that satisfy the restriction ϕ of an adverb by considering all extensions of i in $s[\![\phi]\!]$, for any assignment i in an input state s. Now, in case of asymmetric quantification, we only need to take into account extensions of i that *survive* in $s[\![\phi]\!]$, and test whether these extensions also survive in further update with the nuclear scope of the adverb.

For instance, if we interpret *If a man owns a donkey he is usually rich* asymmetrically in a state s, $s[\![\mathit{Usually}_{\{x\}}(\exists x(Mx \wedge \exists y(Dy \wedge Oxy)))(Bxy)]\!]$, we test, for any $i \in s$ whether most assignments j: $i \leq_{\{x\}} j$ that survive in $s[\![\exists x(Mx \wedge \exists y(Dy \wedge Oxy))]\!]$ also survive in $s[\![\exists x(Mx \wedge \exists y(Dy \wedge Oxy))]\!][\![Rx]\!]$. In fact this tests whether most donkey owning man are rich. Notice that in the present example the nuclear scope Rx of the adverb is interpreted with respect to a state that is not only defined for x, the values of which are quantified over, but also for y, the possibly different values of which do not interfere with the quantification. In other words, we can quantify in *EDPL* over the values of *some* variables introduced in the restriction of an adverb, neglect the valuation of others, without denying the others the ability to serve as an antecedent for anaphoric pronouns in the nuclear scope.

So, assume that an asymmetric adverb of quantification comes with a set of selection indices X the values of which the adverb quantifies over. Its interpretation then is defined as follows:

Definition 3.2 (Adverbs of quantification (asymmetric))
If $X \subseteq (D(s[\![\phi]\!]) - D(s))$

- $s[\![A_X(\phi)(\psi)]\!] = \{i \in s \mid [A](\{j \mid i \leq_X j \;\&\; j < s[\![\phi]\!]\})(\{j \mid j < s[\![\phi]\!][\![\psi]\!]\})\}$

The side condition that $X \subseteq (D(s[\![\phi]\!]) - D(s))$ guarantees that an asymmetric adverbial quantifier A_X effectively quantifies over the values of all the variables in X, this, again, on pain of undefinedness.

Let us briefly consider two mutually related examples.

(22) If a man gives her a PRESENT, she usually thanks him for it
$\mathit{Usually}_{\{y\}}(\exists y(My \wedge \exists z(Pz \wedge Gyzx)))(Txyz)$

This sentence requires of an assignment i in a state of evaluation s that most extensions j of i with a valuation for y such that $j(y)$ gives $i(x)$ a present, are also valuations such that $i(x)$ thanks $j(y)$ for a present $j(y)$ gives to her. So, interpreted in a state s, this example returns all those i in s that assign x an individual that renders thanks to most men that give her a present, irrespective of the number of presents given.

(23) If a MAN gives her a present, she usually thanks him for it
$\mathit{Usually}_{\{z\}}(\exists y(My \wedge \exists z(Pz \wedge Gyzx)))(Txyz)$

When interpreted in a state s, this example returns all those i in s that assign x an individual that renders thanks for most presents given by a man, irrespective of the number of men that give it.

Strong, weak and mixed implication

The preceding discussion shows that *EDPL* treats unselective and asymmetric adverbs of quantification in a uniform way. We find the following equivalences:

Fact 3.2
If defined,

- $\mathit{Sometimes}_X(\phi)(\psi) \Leftrightarrow \mathit{Sometimes}(\phi)(\psi)$
- $\mathit{Never}_X(\phi)(\psi) \Leftrightarrow \mathit{Never}(\phi)(\psi)$
- $\mathit{Always}_\emptyset(\phi)(\psi) \Leftrightarrow (\phi \hookrightarrow \psi)$

So, for the adverbs *sometimes* and *never* it makes no difference whether or not they quantify over a selection of the variables introduced in their restriction. This is as it should be, since there seems to be no evidence whatsoever that there are distinct asymmetric readings of these adverbs.

On the other hand, for the adverbs *usually* and *always*, it does make a difference whether or not they select variables for asymmetric quantification, and which variables they select. Furthermore, we see that the weak implication ($\hookrightarrow$) addressed in the digression of section 3.2, now appears to be a borderline case of asymmetric adverbial quantification, i.e., universal quantification over the values of an empty set of selection indices. So, this weak implication does not constitute a really different notion of implication of its own, but it fits into the more general scheme of adverbial quantification as one of the many forms of asymmetric quantification.

This claim about weak implication can be further substantiated by slightly varying the dime implication again:

(24) If I have a dime in my pocket, I throw it in the parking meter.
$(\exists y(Dy \wedge Piy) \hookrightarrow Tiy) \Leftrightarrow \mathit{Always}_{\emptyset}(\exists y(Dy \wedge Piy))(Tiy)$

(25) If a man has a dime in his pocket, he throws it in the parking meter.
$\mathit{Always}_{\{x\}}(\exists x(Mx \wedge \exists y(Dy \wedge Pxy)))(Txy)$

On its most natural reading, the first example was argued to state that if I have a dime in my pocket, then I throw one in the meter, and this reading is captured by interpreting the conditional sentence as one of weak implication (or of universal quantification over the values of an empty set of selection indices). However, the second example, which is a minor variation of the first, is most likely interpreted as stating that every man who has a dime, throws one in the meter.[9] Neither the weak, nor the strong, reading of the implication gives us this. In fact, this example has a mixed strong and weak reading: strong with respect to the men who are faced with a parking meter and who have a dime in their pocket; weak with respect to the number of dimes they throw in. Now, if the weak implication is supposed to constitute a form of implication of its own and if it is used to account for the first example, we still need an explanation of this second, related, example, which, on the preferred reading, does not exemplify a purely weak or strong implication. If, on the other hand, we fit both these sentences into the scheme of asymmetric adverbial quantification, both get assigned proper readings in a uniform way.

---

9. Of course, these sentences will have to be understood as restricted to men faced with a parking meter they are obliged to throw a dime in. Furthermore, the analysis will have to be supplemented with a proper interpretation of the definite noun phrase *the parking meter*. However, these two issues do not concern us here. Relevant in the present discussion is the quantificational force of the indefinites in the dime implications.

3.3 Adnominal quantifiers

EDPL is also easily extended with adnominal (binary) quantifiers. Basically, the treatment of these quantifiers offered below is that of Chierchia [1988, 1992], also proposed in van Eijck and de Vries [1992]. As is the case with the adverbial quantifiers, the adnominal quantifiers are interpreted as 'internally dynamic', i.e., their analysis accounts for anaphoric relationships between indefinites occurring in the restriction of such quantifiers and anaphoric pronouns in their nuclear scope. I will not present an analysis of the external dynamics of such quantified structures, that is, the anaphoric relationships that may obtain between the noun phrases in quantified structures and pronouns in successive sentences. As is argued in van de Berg [1990], for a treatment of the external dynamics of adnominal quantifiers we need to extend *(E)DPL* so as to deal with plural noun phrases, more in particular with plural pronouns, and such an enterprise falls beyond the scope of the present chapter.

Internally dynamic adnominal quantifiers neatly fit in the general scheme of quantification in *EDPL*. Let D be a binary quantifier which has $[D]$ as its usual set-theoretic interpretation, then:

Definition 3.3 (Binary quantifiers)
If $x \notin D(s)$
- $s[\![Dx(\phi)(\psi)]\!] = \{i \in s \mid [D](\{j \mid i \leq_{\{x\}} j \ \&\ j \lessdot s[x][\![\phi]\!]\})(\{j \mid j \lessdot s[x][\![\phi]\!][\![\psi]\!]\})\}$

On the present definition of adnominal quantification, a binary quantifier Dx quantifies over the possible valuations of a single variable x and, hence, over the individuals in D (that is, if we again assume that the quantifier satisfies the constraints of extension, quantity and conservativity, cf., the remarks in section 4.1).

Let us consider one example:

(26) Most men who gave her a present had packed it up
$\mathit{Most}\, y(\exists z(Pz \wedge Gyzx)(Uyz)$

The interpretation of this example in a state s yields a state consisting of assignments $i \in s$ such that most extensions of i to y under which the value of y is a man who gave a present to the value of x, are extensions under which the value of y has packed up a present he gave to the value of x. Put more simple, the interpretation of this example in a state s returns those assignments i in s which assign an individual z to x such that most men who gave a present to z gave her a present they had packed up.

We find the following correspondences between binary and unary quantifiers in *EDPL*:

Fact 3.3
- $\mathit{An}\, x(\phi)(\psi) \quad \Leftrightarrow \downarrow\exists x(\phi \wedge \psi)$
- $\mathit{No}\, x(\phi)(\psi) \quad \Leftrightarrow \neg\exists x(\phi \wedge \psi)$

- $Every\,x(\phi)(\psi) \Leftrightarrow \forall x(\phi \hookrightarrow \psi)$

We see that the binary determiners *a(n)* and *no* have the same truth-conditional content as their usual first order paraphrases. Moreover, observe that *EDPL* licenses a weak and a strong reading of the quantifier *every*, both of which are intuitively motivated. If we treat *every* as a binary quantifier, the weak reading results. This reading is appropriate for the sentence *Every man who has a dime puts in the parking meter*. On the other hand, if we translate *every* by means of the first order universal quantifier, as $\forall x(\phi \rightarrow \psi)$, then the strong reading results, and this gives the proper reading of the (strong) donkey sentence.

Of course, the ambivalence of *every* also shows in its negation *not_every*. In case *every* is read strongly, a quantified structure *Not every A B* translates as $\neg\forall x(A'x \rightarrow B'x) \Leftrightarrow \downarrow\exists x(A'x \wedge \neg B'x)$. This seems appropriate for the negation of the strong donkey sentence. The sentence *Not every man who owns a donkey beats it* has the truth conditions that there is a man who owns a donkey which he does not beat. On the other hand, if *every* is read weakly, as a binary determiner, then *Not every A B* translates as $Not\_every\,x(\phi)(\psi)$, which is equivalent with $\downarrow\exists x(\downarrow A'x \wedge (A'x \rightarrow \neg B'x))$. This seems appropriate for the negation of the weak dime sentence. The sentence *Not every man who has a dime throws it in the meter* then has the truth conditions that there is a man who has a dime and who does not throw any dime he has in the meter.

To conclude, we find two traditional but non-equivalent ways to translate natural language *every* in the language of *EDPL*, both with intuitively motivated readings. Notice that it is especially the determiner *every* which has been associated with both weak and strong readings in the literature. Now it need not be so much of a merit of *EDPL* that it accounts for both kinds of readings in a principled way, but it is a relative merit that it does do so without entailing ambiguity of all other quantifiers.

A last observation concerns the relation between determiners and quantifying adverbs. The uniform analysis of determiners and adverbs of quantification shows in the following equivalence. If adverb A and determiner D have the same set-theoretical interpretation, then:

Fact 3.4

- $A_{\{x\}}(\exists x\phi)(\psi) \Leftrightarrow\ D\,x(\phi)(\psi)$

Consider the sentence *If a man owns a donkey, he is usually rich* on its asymmetric construal in which *usually* selects donkey owning men for quantification. The translation of the sentence then is $(Usually_{\{x\}}(\exists x(Mx \wedge \exists y(Dy \wedge Oxy)))(Rx))$, and, under this translation, the sentence is equivalent with the sentence *Most men that own a donkey are rich*, translated as $Most\,x(Mx \wedge \exists y(Dy \wedge Oxy))(Rx)$.

Infelicitous indices

As was stated at the start of this section, the present analysis of adnominal quantification is basically that of Chierchia and van Eijck and de Vries. The most significant difference with these two approaches is that *EDPL* explicitly expels the reinstantiation of variables which are already in use. It can be argued that, in particular, the treatment of quantifiers in Chierchia and van Eijck and de Vries calls for such a novelty constraint on indices (which, by the way, is also Chierchia's conclusion).

Consider the following example, in which an indefinite in the restriction of a determiner is coindexed with the determiner:

(27) Every$_x$ man that has a$_y$ kid who owns a donkey$_x$ is happy.
$\mathit{Every}\, x(Mx \wedge \exists y(Ky \wedge \exists x(Dx \wedge Oyx) \wedge Hxy))(Hx)$

With respect to this translation notice, first, that the head determiner *every* has a restriction $Mx \wedge \exists y(Ky \wedge \exists x(Dx \wedge Oyx) \wedge Hxy)$ in which the quantifier $\exists x$ associated with the noun phrase *a donkey* binds the variable x in Hxy. So, under this translation the head determiner *every* quantifies, not over men that have a kid who own a donkey, but over men such that there is a kid who owns a donkey which has the kid, i.e., over men such that there is a donkey which has a kid who owns it.

Second, it appears that also the occurrence of x in the nuclear scope Hx gets bound by the quantifier $\exists x$ in the restriction of the head determiner *every*. The reading that results then is that, for every man, if there is a donkey which has a kid that owns it, then there is a *happy* donkey which has a kid that owns it. Clearly, this is not at all a sensible reading of the original example *Every man that has a kid who owns a donkey is happy*, but if we assume a standard translation procedure and a free indexing mechanism it is predicted to be one.

In Chierchia [1992] the problem just sketched is circumvented by the use of λ-recipes in the translation of quantified structures, this in a way that need not concern us here. Still, Chierchia as well finds reason to require indefinites always to introduce fresh variables. However, the motivation he gives for this is slightly misguided.

Chierchia [1992, pp. 141 ff] proposes a conservativity constraint on determiners D to the effect that $D(A)(B) \Leftrightarrow D(A)(A \,\&\, B)$. Next he argues that if in the restriction A of the determiner a free variable in the restriction is reinstantiated, then his theory would predict wrong readings. Consider, for instance, the sentence *Every man who knows her*$_x$ *and marries an*$_x$ *Italian is happy*. Chierchia argues that if conservativity holds then this sentence is equivalent with the sentence *Every man who knows her*$_x$ *and marries an*$_x$ *Italian is a man who knows her*$_x$ *and marries an*$_x$ *Italian and who is happy*, which, in its turn, is equivalent with *Every man who knows her*$_x$ *and marries an*$_x$ *Italian is a man who knows her*$_x$ *and marries an*$_x$ *Italian and is a man who knows her*$_x$ *and marries an*$_x$ *Italian and who is happy*. Observe that in the nuclear scope of the last sentence the first occurrence of *an*$_x$

Italian binds the second, coindexed, occurrence of *her*$_x$ and, hence, that sentence means that every man that knows her and marries an Italian is a man who marries an Italian *he knows* and who is happy.

Chierchia concludes from this that also the original sentence *Every man that knows her*$_x$ *and marries an*$_x$ *Italian is happy* has a reading that every man that knows her and marries an Italian is a happy man who marries an Italian he *knows*. However, this is not what the argument shows, since the original sentence, on Chierchia's proposal, simply does not have that reading. So, what the argument *does* show is that the original sentence is not equivalent with the sentence derived from it by conservativity, and, hence, that quantifiers in Chierchia's system are not in general conservative in the indicated sense.

Although the argument is somewhat misguided, its conclusion may, nevertheless, remain the same. Chierchia concludes that reinstantiation has to be expelled and this will also be the conclusion if, as Chierchia apparently wants, a dynamic variant of conservativity is to be retained as a general constraint on natural language determiners. The argument shows clearly that if a form of conservativity in Chierchia's sense is to be retained, it should at least be required that indefinites always introduce novel variables. To that same end, then, Chierchia [1992, appendix V] suggests the adoption of partial assignments and partial interpretation, and this chapter can be taken to show how such a suggestion may work out in detail.[10]

Final issues

To conclude this section, I will briefly reconsider some of the facts from section 3.2 and 3.4 in view of the extension of *EDPL* with quantifiers.

It is easily shown that update and distributivity (facts 2.6 and 2.7) remain valid when *EDPL* is extended with quantifiers. (To prove distributivity, use the conservativity of the quantifiers used.) To prove that also the facts 2.13, 2.14 and 2.15 remain valid it suffices to prove lemma 1 for the extended system. In order to prove lemma 1 in the extended system, the induction in section 3.4 only needs to be extended with a clause dealing with asymmetric adverbs, since unselective adverbial quantification and adnominal quantification is subsumed under asymmetric quantification.

To facilitate exposition, I use the following abbreviatons:

10. It must be noticed that also the notion of conservativity should be subject to further scrutiny. As things stand, Chierchia's notion of dynamic conservativity is incompatible with the novelty conditon he proposes. If the restriction A of a quantifier contains an existential quantifier $\exists x$, then the conservative copy of A into the nuclear scope B always yields an undefined formula, that is, on the novelty condition that Chierchia proposes to adopt. A simple solution to this problem consists in restricting dynamic conservativity to the unindexed quantified structures in natural language. The novelty condition then may be taken to apply to the indexing of their conservative paraphrases.

$s_i = \{k \mid i \leq_X k \ \&\ k \lessdot \{i\}[\![\phi]\!]\}$,
$t_i = \{k \mid i \leq_X k \ \&\ k \lessdot \{i\}[\![\phi]\!][\![\psi]\!]\}$,
$s_j = \{l \mid j \leq_X l \ \&\ l \lessdot \{j\}[\![\phi]\!]\}$ and
$t_j = \{l \mid j \leq_X l \ \&\ l \lessdot \{j\}[\![\phi]\!][\![\psi]\!]\}$.

The inductive proof of lemma 1 for quantified structures employs the corollary of lemma 1 that if $i \approx j$ and $X \cap D(i) = X \cap D(j) = \emptyset$, then for all k such that $i \leq_X k$ and $k \lessdot \{i\}[\![\phi]\!]$ there is one l such that $j \leq_X l$, $l \lessdot \{j\}[\![\phi]\!]$ and $k \approx l$, if defined. So, if, by the induction hypothesis lemma 1 holds for $[\![\phi]\!]$ and $[\![\psi]\!]$, then there is a bijection from s_i to s_j, the restriction of which to t_i, moreover, is a bijection from t_i to t_j.

Suppose that $i \approx j$, that $\{i\}[\![A_X(\phi)(\psi)]\!]$, $\{j\}[\![A_X(\phi)(\psi)]\!]$ are defined and that $f \in \{i\}[\![A_X(\phi)(\psi)]\!]$, i.e., $f = i$ and $i \in \{i\}[\![A_X(\phi)(\psi)]\!]$. Since $[A]$ is conservative $[A](s_i)(t_i)$ is true, and since $[A]$ satisfies extension $[A]_{s_i}(s_i)(t_i)$ is true. By induction, the corollary of lemma 1 entails the existence of a bijection from s_i (and t_i) to s_j (and t_j). Hence, since $[A]$ satisfies quantity, $[A]_{s_j}(s_j)(t_j)$ is true. By extension again we have that $[A](s_j)(t_j)$ is true, and, using conservativity, $j \in \{j\}[\![A_X(\phi)(\psi)]\!]$. Since $f = i$, $f \approx j$. Furthermore, for any $j' \in \{j\}[\![A_X(\phi)(\psi)]\!]$, by definition, $j' = j$. So, if $i \approx j$ and $\{i\}[\![A_X(\phi)(\psi)]\!]$, $\{j\}[\![A_X(\phi)(\psi)]\!]$ are defined, then for any $f \in \{i\}[\![A_X(\phi)(\psi)]\!]$ there is one $f' \in \{j\}[\![A_X(\phi)(\psi)]\!]$ such that $f \approx f'$.

3.4 On situations

Instead of Lewisian cases, 'situations' (or 'occasions', or 'events') have been argued to be needed in order to deal with symmetric and asymmetric readings of donkey sentences (Berman [1987], Kadmon [1987, 1990], Chierchia [1988], Heim [1990], van Eijck and de Vries [1992], among others). Like Chierchia [1992], I think that, although the notion of a situation will probably be useful for the semantics of natural language, such entities do not basically contribute to the understanding of symmetric and asymmetric adverbial quantification. In this section I point out some limitations of the approaches proposed, which I will dub 'situation-based approaches', and argue that they neither refute, nor improve upon the Lewisian analysis adopted above.

I first discuss the two most recent situation-based accounts of adverbial quantification and next discuss some general objections that have been raised against such proposals.

Situations and asymmetric quantification

The most recent situation-based account of adverbial quantification, and the one that most clearly expresses its intuitive appeal, can be found in van Eijck and de Vries [1992]. The basic idea in this approach, which is essentially similar to that of Chierchia [1988], is twofold. In the first place predicates are assigned an argument slot for situations ('occassions' in Chierchia and van Eijck and de Vries).[11] In the second place adverbial quantifiers are interpreted as quantifying over such situations.[12]

To get an idea, the predicate *put_in*, for instance, is conceived of as a quaternary relation which holds of three individuals z, z', z'' and a situation s iff in s z puts z' in z''. The sentence *If a man has a dime, he puts it in the parking meter* is conceived of as asserting that for all situations s, if a man has a dime in s, then in s a man puts a dime in the parking meter. Van Eijck and de Vries: "We can paraphrase the semantic condition imposed by (56) [the interpretation of the dime sentence, PD] as follows: every occasion where there is a man with a dime is an occasion where there is a man who puts a dime in the parking meter. This is intuitively acceptable." Indeed, this paraphrase is very intuitive, and quite correctly it may be taken to predict truth conditions that do not require all men to throw *all* the dimes they have in the parking meter. That is to say, the sentence is predicted to have these weak truth conditions if we assume that if in a certain situation a man has any two dimes d and d' in his pocket, then there are no 'sub-situations' in which he has d in his pocket, but not d'. This assumption, as well, seems quite acceptable.

It is also suggested that this situation-based approach is able to give an account of an asymmetric reading of the following sentence:

(28) If a girl has a boyfriend, she usually teases him.

Van Eijck and de Vries point out that this sentence at least has a reading in which the adverb *usually* quantifies over girls who have a boyfriend and not over girl–boyfriend pairs. The sentence is taken to state that the quantifier relation *Most* holds of the set of situations in which a girl has a boyfriend and the set of situations in which a girl has a boyfriend which she teases. Van Eijck and de Vries add: "As soon as we have a model where occasions are fully individuated, our quantificational analysis gives the right meanings." (p. 14)

The last claim is highly disputable. Let us just assume that we have a model where situations are individuated in a way that makes the adverb in the sentence at issue in effect quantify over girls who have a boyfriend, and not over girl–boyfriend pairs nor over boyfriends of a girl. Then, on van Eijck and de Vries' analysis two things immediately follow. In the first place, the sentence then *lacks* a symmetric reading. Adverbial quantification remains to be analyzed in terms of quantification over situations, and since it is assumed that quantification over situations in which a girl has a boyfriend amounts to quantification over girls who have a boyfriend, the above

11. See Davidson [1967], Parsons [1985, 1989, 1991], Dowty [1989], for independent motivation for assigning verbs a situation (or event) argument slot.

12. Stump [1981], Rooth [1987], Kadmon [1987, 1990], Berman [1987], Kratzer [1988], Heim [1990], de Swart [1991] also analyze quantifying adverbs as quantifiers that range over situations (events, occasions).

sentence can only be read as quantifying over girls who have a boyfriend. More in general, van Eijck and de Vries' proposal excludes the, undisputed, coexistence of various symmetric and asymmetric readings that adverbially quantified sentences may have.

In the second place, let us furthermore adopt the quite natural assumption that a boy is the boyfriend of a girl iff the girl is a girlfriend of the boy. Then assume again that *If a girl has a boyfriend she usually teases him* amounts to quantification over girls who have a boyfriend. The last assumption then precludes the possibility that the structurally completely parallel sentence *If a boy has a girlfriend, he usually teases her* amounts to quantification over boys who have a girlfriend. This is quite an unwanted result. Moreover, on the same assumptions the sentence *Always if a boy has a girlfriend, he is happy* would be predicted only to have a reading saying that every girl that has one or more boyfriends has at least one happy boyfriend. Clearly, this is one of the least plausible readings of this sentence, if it is a reading at all.

Several authors (Bäuerle and Egli [1985], Kadmon [1987, 1990], Kratzer [1988], Heim [1990], among others) have discussed some, perhaps not always tenable, but certainly valuable empirical observations about linguistic factors that favour certain symmetric or asymmetric readings of adverbially quantified sentences. These observations all presuppose that there is a variable individuation of the objects quantified over, whether these are situations, cases, assignments or whatever. If, as in van Eijck and de Vries' proposal, it remains entirely up to the individuation of situations in the *model* what adverbial quantification amounts to, the analysis is too rigid to properly account for the variability of symmetric and asymmetric quantification, let alone to even raise the issue what linguistic factors favour or disfavour certain readings.

Situations and E-type pronouns

An alternative and very elaborate treatment of donkey sentences that critically employs situations can be found in Heim [1990]. Heim builds on proposals in Berman [1987] and Kadmon [1987, 1990] and adopts Evans' [1977, 1980] E-type analysis of pronouns within a situation based model.

Basically, the idea is this. In the first place, predicates and nouns are interpreted relative to situations. Heim here employs the situations from Kratzer [1989], which are considered parts of the world partially ordered by a part-of relation. Second, as above, adverbial quantifiers are taken to quantify over situations. Finally, donkey pronouns are interpreted as definite descriptions which identify a referent by means of a description which is retrieved from the linguistic context by a transformational copying rule. These descriptions are interpreted relative to certain (minimal) situations, and, therefore, they do not, in general, raise overly objectionable uniqueness presuppositions. In order to illustrate Heim's theory, I sketch the analysis of two prototypical examples.

First, the sentence *If a man owns a donkey, he is happy* is given the following interpretation: *Every minimal situation s in which there is a man and a donkey owned by the man is part of a situation in which the (unique) man in s is happy*. Since quantification is over *minimal* situations in which a man owns a donkey, and under the assumption that there is only one man in any minimal situation in which a man owns a donkey, the definite *the man* is defined in each situation satisfying the restriction. Moreover, under the same assumption, the sentence turns out true iff every man who owns a donkey is happy.

The second example is *If a man owns a donkey, he is usually rich*, where the adverb *usually* is interpreted as quantifying over donkey owning men. This sentence is interpreted as follows: *Most minimal situations s in which there is a man z and which are part of a situation in which z owns a donkey, are part of a situation in which the (unique) man in s is rich.* So, under an assumption similar to the one in the former example, since quantification here is over *minimal* situations in which there is a man and which are part of a situation where he has a donkey, it comes down to quantification over men who own a donkey.

I will now argue that Heim's (and Kadmon's) analysis, as it stands, still makes too strong uniqueness predictions. The relevant examples here are adverbially quantified sentences in which two indefinites in the restrictive clause are both referred back to by pronouns in the nuclear scope.[13]

Consider the following example:

(29) If a man has a pet, he usually treats it well.

Let us suppose that *usually* quantifies, asymmetrically, over men which have a pet. Heim's analysis yields an interpretation which can be phrased as follows:

(30) Most minimal situations s in which there is a man z and which are part of a situation in which z has a pet, are part of a situation s' in which the man in s treats the pet the man in s owns well.

Under an assumption similar to the ones in the former two examples, the pronoun *he* in this example is given a well-defined interpretation, but the interpretation of

13. Strictly speaking, even the pronouns in a donkey implication *If a farmer owns a donkey he beats it* are not fully analyzed. In Heims final proposal, a transformational copying rule associates such pronouns with a description that is employed to identify their referents. For the donkey implication this rule generates one of two possible logical forms which can be phrased as follows: *If* a_x *farmer owns* a_y *donkey, the farmer who owns a donkey beats the donkey* x *owns*, and *If* a_x *farmer owns* a_y *donkey, the farmer who owns* y *beats the donkey a farmer owns*. In both logical forms, the induced definite description contains a free variable (x and y, respectively) which cannot be dealt with by other applications of the transformation rule as it is stated. Clearly, in order to treat the donkey implication the analysis has to be extended. In what follows, I will simply assume some such extension and use the paraphrases of occurring pronouns which Heim herself uses.

the pronoun *it* is problematic. On Heim's analysis, for this example to be defined, it is required that every men who owns a pet owns only one pet.[14] For this reason, Heim concludes that a sentence like the above requires us either to 'accomodate' the uniqueness assumption that no man owns more than one pet, or, otherwise, to be forced to interpret the sentence symmetrically (Heim [1990, p. 156]).[15]

I think that this is very unsatisfactory. I have not seen convincing, theory independent, motivation for assuming asymmetric quantifiers to involve such uniqueness effects. The dime implication discussed above in fact points to the contrary:

(31) If a man has a dime, he always throws it in the parking meter.

As is argued above, the symmetric interpretation of this sentence is highly implausible. But the alternative that Heim and Kadmon leave us is to accomodate the assumption that every man has at most one dime. This is just as implausible. Clearly, on its most intuitive reading this example is not symmetric nor does it involve such a uniqueness requirement. Sentences like these, then, fall beyond the scope of Heim's and Kadmon's analysis of adverbial quantification.

Adverbial quantification over situations

I will now turn to two general objections that have been raised against attempts to treat adverbial quantifiers, uniformly, as quantifiers over situations.

Already in Lewis [1975] it is acknowledged that situations, or, rather, events and moments or stretches of time, may be (involved in) the objects over which adverbial quantification takes place. However, the primary claim in that article is that events and times do not in general provide for the objects over which adverbs of quantification can be taken to quantify. Clear-cut exceptions are adverbially quantified structures in which the restriction and nuclear scope are interpreted independently of any situational context. Mathematical statements are a case in point, since mathematical predicates do not seem to express time or event bound properties in a sensible sense of times and events. Lewis discusses the following example:

---

14. In fact, this is the kind of uniqueness 'effect' Kadmon [1987, 1990] argues for. I think that Kadmon's observations concerning the uniqueness/maximality of plural pronouns are highly valuable, but the specific uniqueness effects of singular pronouns which she predicts are disputable and seem quite theory dependent. Unfortunately, it would take us too far if I were to revive the whole discussion here.

15. Notice, that if the assumption is accomodated, the asymmetric interpretation of the example is equivalent with the symmetric one. As soon as we assume that no man owns more than one pet, then it makes no difference whether we quantify over minimal situations in which a man has a pet or over minimal situations in which there is a man and which are part of a situation in which there is also the man's unique pet, that is, on the assumptions Heim and Kadmon apparently make about the individuation of situations and on the assumption that the quantifiers at issue satisfy the constraints of extension, quantity and conservativity. So, on these assumptions, Heim's asymmetric reading of the above sentence *only* differs from its symmetric reading in that it has the indicated uniqueness presuppositions.

(32) A quadratic equation never has more than two solutions.

Under no sensible notion of situations (events, times), it seems, can statements like these be properly conceived of as quantifying over situations (events, times).

Other examples can be made up from what Kratzer [1988] calls 'individual-level' (or 'stative') predicates. In contradistinction with so-called 'stage-level' (or 'event') predicates, individual-level predicates lack an argument position for events or spatiotemporal location. Still, sentences with only individual level predicates do occur in adverbially quantified structures:

(33) When a Moroccan knows French, she usually knows it well.

It does not seem to make good sense to interpret this example as stating that most situations in which a Moroccan knows French, are situations in which he or she knows it well.

Another argument raised against situation-based approaches to adverbial quantification has to do with the so-called cases of the indistinguishable participants.[16] Consider the following example:

(34) If a bishop meets another man he blesses him.

This sentence is generally read as expressing that every bishop blesses any man he meets. As several people have shown, a situation-based treatment of this example runs into troubles when we consider cases where two bishops meet. On Kadmon's and Heim's analysis the pronoun *he* is associated with the definite description *the bishop*, and would fail to identify a referent in situations in which two bishops meet. Therefore, if there are such bishop-bishop encounters, the sentence must be taken to quantify, asymmetrically, over minimal situations which contain a bishop and which are part of a situation where that bishop meets another man (possibly a bishop). Furthermore, since the second pronoun is associated with the description *the man the bishop meets*, for the pronoun to be defined it would have to be presupposed that every bishop meets at most one man. So, if there are bishops who meet other bishops, an example like the above is predicted to display uniqueness effects.

It is hard to make out whether these predictions are realistic in the present case, since it can be argued that the symmetric predicate *meet* in example 34 applies to two person encounters only, and, thus, renders the uniqueness presuppositions trivially satisfied, i.e., irrefutable. Still, Heim expresses her doubts whether the predictions are fulfilled in other, related, cases. ([1990, p. 157-8]) In my opinion, the following two examples show that they are not, at least not in general:

(35) If a Dutchman has quarrels with a neighbour, he always/usually detests his political taste.

16. This label is Heim's, the example is attributed to Kamp, and a basically similar example comes from van Eijck. Heim moreover credits Kamp for having stressed the significance of this example at several occasions and Chierchia credits Kamp and Rooth. Since what follows mainly reports existing observations, I will be on the safe side if I just credit them all.

(36) If a soccer player competes with another player for a place in the A-team, he usually thwarts him whenever he can.

Dutchmen normally have at least two neighbours all of whom they can have quarrels with. Also, several soccer players may be in the running for one and the same place in the A-team. Nevertheless, these examples seem fine and the quantifying adverbs do not seem to be restricted to quantify only over the cases in which a Dutchman quarrels with only one neighbour, or over the cases where only two soccer players fight for a place in the A-team, respectively. So, at least the prediction of a uniqueness effect is not corroborated.

Example 34 also poses a problem for the situation-based proposals of Chierchia [1988] and van Eijck and de Vries. On these proposals, the sentence would be true if every situation where a bishop meets another man is a situation where a bishop blesses another man he meets. Now consider again a situation in which two bishops meet. On the sketched analysis it would only be required that at least one bishop blesses the other. This is too weak a requirement, since on the generally accepted interpretation of example 34 it is required that every bishop blesses every man (including every bishop) he meets.[17]

The point here is that, intuitively, every meeting corresponds to one (minimal) situation. However, example 34 does not seem to quantify over such situations. On the generally accepted analysis example 34 says that that for any b and b', if b is a bishop, b' a man, possibly a bishop, and if b meets b', then b blesses b'. So, the adverbial quantifier *every* appears to discriminate the case in which a bishop b meets a bishop b' from the case in which bishop b' meets b, and of both b and b' example 34 says that they bless b' and b, respectively. Notice that this is precisely what the Lewisian analysis gives us.

Chierchia concludes from this that, if the adverb in example 34 requires us to distinguish the case where b meets b' from the case where b' meets b, then "what follows under the present account is that occasions have to be more finely structured than what we have been assuming. Perhaps as fine grained as assignments to sequences of variables, which is of course the idea that *DRT* is built on." ([1988, p. 72]). And, in fact, Chierchia [1992] drops the hypothesis that quantifying adverbs uniformly quantify over situations. In the latter paper such adverbs quantify over variable assignments, or, more precisely, over tuples of variable values. It may be

---

17. Notice that on the analysis of Chierchia [1988] and van Eijck and de Vries the following sentences can both be true:

(37) Always if a bishop meets another man he blesses him.
(38) Always if a bishop meets another man he does not bless him.

For instance, if only bishops meet bishops and if on every meeting of bishops at least one bishop blesses another bishop and at least one does not bless another one, then these sentences would come out both true. This seems mistaken. Intuitively, both sentences should come out *false* then.

noted here that, like in Lewis paper [1975], situations may be included among the objects quantified over.

***

The above discussion shows that, concerning the relatively intuitive notion of a situation that has been adopted for an analysis of adverbially quantified donkey sentences, situations are irrelevant in some cases (with respect to situation independent clauses), and too coarsely grained in others (unable to discriminate 'indistinguishable' participants).

The proposals to treat adverbs as quantifiers over situations also do not seem to be particularly useful for a general analysis of asymmetric quantification. As we saw above, in van Eijck and de Vries the notion of situation used is too rigid and excludes the coexistence of symmetric and asymmetric readings of sentences. Clearly, if adverbs may be taken to quantify symmetrically and asymmetrically, and if adverbs quantify over situations, then situations need to be quantified over flexibly. For asymmetric readings of an adverb, Heim then uses sub-minimal situations, situations which are part of a minimal situation in which the restriction of the adverb is true. However, this approach does not go together well with the adopted E-type approach, since the use of subminimal situations generates what I take to be unwanted uniqueness presuppositions.

Berman [1987] and Chierchia [1988] propose accounting for asymmetric quantification by appealing to contextual factors that are assumed to restrict the domain of situations quantified over. For instance, one can obtain an asymmetric reading of *If a man owns a donkey, he is usually rich* by restricting the quantifier *usually* to a domain of situations which do not distinguish between the several donkeys a man may have. As is the case with Heim's subminimal situations, each situation in such a domain then can be assumed to correspond uniquely to a man who owns a donkey, and quantification over situations in that domain then amounts to quantification over donkey owning men. However, this can hardly be called an analysis of asymmetric quantification as long as it is unclear what contextual factors are supposed to affect the delimitation of the domain and with what effect. Clearly, something must be added here, if only to exclude a restriction of the domain of situations, say, to that of situations in which no man beats a donkey.

As is argued in section 4.2, Root has proposed an effective way to deal with asymmetric readings by shifting from a domain of assignments to a domain of equivalence classes of assignments. For as far as the asymmetric reading of *If a farmer owns a donkey he is usually happy* is concerned, the quantifier is restricted to a domain of sets of assigments, each set in which uniquely determines a farmer who owns one or more donkeys together with all donkeys he owns. This approach gives a plausible and clear-cut treament of asymmetrically quantified donkey sentences. Moreover, in the update approach of *EDPL* it appears that asymmetric quantifiers do not even require us to resort to a different level of objects (classes of assignments instead of assignments), since the Rootian equivalence classes of assignments can be taken to be partial assignments, which are the kind of entities in terms of which the semantics of *EDPL* itself is defined.

To conclude this section, a situation based approach, in the present state of the art, does not refute, nor improve upon the Lewisian treatment of adverbial quantifiers as quantifiers over cases. Like Lewis and Chierchia, I think that situations (events, occasions, time intervals) may be properly included in the things, i.e., the tuples, quantifying adverbs quantify over, but the use of situations has not been shown to enhance a fully general analysis of symmetric and asymmetric quantification. On the contrary, it appears that if unselectivity is dissociated from the Lewisian approach, this approach fares best in dealing with both symmetric and asymmetric quantification. Adverbs of quantification need not quantify over the 'complete' tuples of individuals addressed in their restriction, it may be over parts of them. As is shown in the preceding section, both forms of quantification can be dealt with, uniformly, in terms of quantification over partial assignments.

4 The notion of information about variables

This last section treats four loose ends, which are tied together since they are all concerned with elucidating the notion of information about the values of variables. In section 4.1 the extensional system of *EDPL* is turned into an intensional system in which information states also contain information about the world and restrict the values of variables relative to the worlds which are still considered possible in such states. The resulting system of interpretation, *MDPL*, properly comprises *EDPL* and *US* interpretation.

The next three sections build on *MDPL*, but all the results stated carry over to *EDPL*. Section 4.2 states some further logical properties of the *MDPL* structure of information states and of the *MDPL* interpretation in that structure. It is shown that all *MDPL* updates can be characterized in terms of an (associative) product operation on the domain of information states and that it is only definedness conditions, that is, in fact, the felicity conditions on the introduction of discourse referents, that prevents *MDPL* from collapsing into a static semantics.

In section 4.3 it is argued that the use of Veltman's epistemic operators can be best characterized at a semantic/pragmatic level of information exchange, rather than on the purely semantic level of information update. The *MDPL* structure of information states is then used in an extension of *MDPL* to a rudimentary model of information exchange. The epistemic operators, with their original Veltman semantics, are shown to be analyzed reasonably well in this extended model.

Section 4.4, finally, studies the notion 'the value of a variable', which *MDPL*'s information states are claimed to contain information about. In that section I give a precise definition of this notion and I show that, on this definition, the notion of the value of a variable has the properties of a, maybe more intuitive, notion of a partial object. *MDPL*'s information states, thus, can be properly conceived of as partial worlds inhabited by partial objects.

Unfortunately, lack of space prohibits the presentation of a comparison with related frameworks such as data semantics (Veltman [1981, 1985], Landman [1986]) and versions of situation semantics (Barwise and Perry [1983], Kratzer [1988] and Muskens [1989]). This, then, must be left for another occasion.

4.1 Intensional MDPL

The *EDPL* notion of information about the values of variables is an extensional one. The models for *EDPL* are extensional, i.e., the predicates and relation expressions are interpreted as sets of (tuples of) individuals, and, similarly, information about the values of (sequences of) variables is cast in terms of the sets of (tuples of) individuals that are their possible values. This notion of information is somewhat unsatisfactory.

In the first place, an *EDPL* formula without free variables still behaves classically in the sense that it is either true or false in an information state s, that is, if it is defined for s of course. Apart from the introduction of new objects to its domain, such a formula does not give a genuine update of the information contained in s. The interpretation of such a formula in s either preserves all the assignments to the variables in s, if the formula is true in s, or rejects them all, if the formula is false in s. However, from the epistemic perspective upon interpretation adopted here, one would expect, at least, that contingent formulas *do* give rise to genuine information updates.

Related to the above is the fact that the extensionality of *EDPL* offers no room for a sensible interpretation of epistemic operators such as, for instance, Veltman epistemic operator *might*.

Consider the following example:

(39) A man is raking the leaves over there in the park. Maybe he found your bracelet. .... He did not find your bracelet.

Assuming that $\Diamond\phi$ in s expresses consistency of s with ϕ, as in Veltman's update semantics, this sentence is consistent, which is as it should be. As long as one does not have the information that if a man is raking the leaves in the park he never finds your bracelet, it is consistent to say that a man who is raking the leaves in the park might have found your bracelet, also if you later learn he did not find it. However, in every extensional model of *EDPL* either every information state contains the information that if a man is raking the leaves in the park he never finds your bracelet, or every information state contains information to the contrary. The models themselves decide whether a man who is raking the leaves may have found your bracelet.

As a consequence, for any state s, if s is updated with *A man is raking the leaves over there in the park* then it is either inconsistent to add *Maybe he found your bracelet*, or it is consistent, and in that case you know that there is a man who is raking the leaves who has found your bracelet. Hence, *A man who is raking the leaves in the park might have found your bracelet* entails *A man who is raking the leaves in the park has found your bracelet*. This should not be the case.

Notice that it is precisely the extensionality of *EDPL* that troubles us here. Since predicates are assigned a fixed extension, and since information states only contain information about the extensions of variables, it is the model that decides, if the value of a variable x is required to be in the set of men who rake the leaves in the park, whether there is a value of a variable x which is also in the set of individuals who have found your bracelet. The conclusion then is that, in order to improve upon this situation, we have to adopt a more intensional notion of information.

The subject of this section is to show how *EDPL* can be intensionalized. The intensional version of *EDPL* will be referred to as *MDPL*.

The world as a subject

Basically, *EDPL* can be turned into a modal dynamic predicate logic, *MDPL*, by means of the following two adjustments. The first adjustment consists in the adoption of intensional models. An intensional model M for *MDPL* is a modal predicate logic model, i.e., a triple $\langle D, W, F\rangle$, consisting of a non-empty set D of individuals, a distinct non-empty set W of possible worlds, and an intensional interpretation function F for the non-logical constants. The function F assigns the predicate and relation expressions an extension relative to possible worlds, that is, F maps each n-ary relation expression to a function from worlds to sets of n-tuples of objects.

The second adjustment intensionalizes the *EDPL* notion of information. Like the interpretation of the non-logical constants of the language, the possible valuations of variables are relativized to possible worlds. Information about the values of variables in *MDPL* is cast in terms of the possible assignments to variables relative to the worlds which are considered possible. Thus, information about the values of a sequence of variables can be conceived of as a function which assigns to any possible world the possible values of the variables in that world and, hence, it encodes the requirement that the variables stand in a certain relation.

I will elaborate upon the second adjustment first, and then show how intensional model are employed in the semantics of *MDPL*.

MDPL relates information about the values of variables to possible worlds by introducing the world as a subject, labeled by a distinguished variable v. The possible valuations of v characterize the possible ways the world is considered to be like. The indices in *MDPL*, i.e., the elements of information states, are assignments which have v in their domain. An assignment i to the variables $x_1, \ldots, x_n$ and v must be understood to register that the variables have possible values $i(x_1), \ldots i(x_n)$ in world $i(v)$.[18]

So, for any domain of variables $X \subseteq V$, we will be concerned with the assignments in the set $\{i \cup j \mid i \in W^{\{v\}} \ \& \ j \in D^X\}$ which I refer to as D^X_W. D^X_W contains all the assignments which assign individuals to the variables in X and a world to v. S^X_W, the set of intensional information states about the values of X is the set of subsets of D^X_W:

Definition 4.1 (MDPL information states)

- $S^X_W = \mathcal{P}(D^X_W)$
- $S_W = \bigcup_{X \subseteq V} S^X_W$

An *MDPL* information state about the values of a set of variables is a set of assignments assigning possible values of X relative to possible worlds, i.e., possible values of v. These information states integrate *EDPL* information about the values of variables and *US* information about the world.

If we fix a world w, which can be conceived of as an *EDPL* model, an *MDPL* information state $s \in S^X_W$ characterizes an *EDPL* information state. The possible assignments to X in w conforming to s are $\{i \in D^X \mid \exists j \in s\colon i \leq_{\{v\}} j \text{ and } j(v) = w\}$. So, *MDPL* information states in fact associate *EDPL* information with worlds.

On the other hand, if we fix a possible assignment $i \in D^X$, then an *MDPL* information state $s \in S^X_W$ characterizes a set of worlds, a *US* information state. Relative to i, s restricts the set of possible worlds to the set of worlds in which i is considered to be a possible assignment to the variables in X: $\{j(v) \mid j \in s \text{ and } i \leq_{\{v\}} j\}$. So, *MDPL* information states also associate *US* information with variable assignments.

The set of assignments to no variables $D^\emptyset_W$ $(= W^{\{v\}})$ in *MDPL* uniquely corresponds with the set of possible worlds. (The function ι from $D^\emptyset_W$ to W such that $\iota(i) = i(v)$ is a bijection.) Therefore, the set of information states about no variables $S^\emptyset_W$ $(= \mathcal{P}(W^{\{v\}}))$ in *MDPL* is isomorphic to the set of sets of worlds ($\mathcal{P}(W)$). Recall that the set of information states about no variables in *EDPL* only consists of a true state and an absurd state. So, in contradistinction with *EDPL*, information states

---

18. Of course, we might as well use as indices pairs consisting of a world and an assignment, as in Heim [1982, 1983]. This comes down to the very same thing, since the set of pairs consisting of a world and an assignment of individuals to the variables in X is isomorphic to the set of assignments which assign individuals to the variables in X and a world to v. I have no principled reason for using assignments of worlds to the variable v instead of the worlds themselves, only an economic one. By introducing the world as a subject, we can retain all facts and definitions from the sections 3 and 4 in unaltered form.

about the values of no variables in *MDPL* may contain substantial information, that is, they may contain information about the world.

The notions of minimal, maximal and absurd information remain the same as in *EDPL*.

The information that *MDPL* information states contain about the values of variables is of an intensional nature. *MDPL* information requires not so much that the values of variables are in certain *sets*, but that they have certain *properties*. In *MDPL*, the information that a state s has about the values of a sequence of variables is that they stand in a certain relation, where relation is conceived to be, what Montague calls, a relation in intension, i.e., a function from possible worlds to relation extensions.

Let us call such a relation the relation ascribed to a tuple of variables $x_1, \ldots, x_n$ in a state s:

Definition 4.2
If $x_1, \ldots, x_n \in D(s)$,

- the relation ascribed to $x_1, \ldots, x_n$ in s, $\mathcal{R}^s_{x_1,\ldots,x_n}$, is the function $f\colon W \to 2^{D^n}$ such that $\forall w \in W\colon f(w) = \{\langle i(x_1), \ldots, i(x_n)\rangle \mid i \in s \;\&\; i(v) = w\}$

Information about the values of $x_1, \ldots, x_n$ is a function which assigns to a possible world the set of possible values of $x_1, \ldots, x_n$ in that world.

So, if P is the property of being a walking man, i.e., the function that assigns to every world the set of walking men in that world, then a state s can be said to contain the information that x has the property P, i.e., that x has the property of being a walking man, iff $\forall w \in W : \mathcal{R}^s_x(w) \subseteq F(P)(w)$.

Interpretation in MDPL

Since information states in *MDPL* are sets of assignments, like the states in *EDPL*, the definitions (and facts) of *EDPL* may be adopted without amendments, that is, except for the clause that defines the interpretation of atomic formulas. The interpretation of atomic formulas in *MDPL* is defined as follows:

Definition 4.3 (MDPL semantics (atomic formulas))

- $s[\![Rx_1 \ldots x_n]\!] = \{i \in s \mid \langle i(x_1), \ldots, i(x_n)\rangle \in F(R)(i(v))\}$ if $x_1, \ldots x_n \in D(s)$

Notice that the *MDPL* interpretation of an atomic formula is not defined relative to a possible world. The information state s itself is updated with the information that the values of $x_1, \ldots x_n$ stand in the relation R in the world assigned to v. The interpretation of an atomic formula $Rx_1 \ldots x_n$ in s in *MDPL* retains those $i \in s$ that assign the variables $x_1 \ldots x_n$ a value that is in the extension of R in $i(v)$.

So, update with $Rx_1 \ldots x_n$ rejects a possible assignment to the variables $x_1, \ldots, x_n$ if it is part of an assignment $i \in s$ such that the values of $x_1, \ldots x_n$ are not in the extension of R in $i(v)$, and the assignment to the variables is retained as part of an assignment $j \in s$ if the values of $x_1, \ldots x_n$ are in the extension of R in $j(v)$. So, in *MDPL*, a valuation of the variables $x_1, \ldots x_n$ may survive with respect to some associated worlds, and be rejected in relation to others.

In *EDPL*, formulas without free variables are either true or false in a state s, that is, if they are defined of course. This is different in *MDPL*. Like information states about the values of no variables, formulas without free variables may convey contingent information. Consider an atomic formula consisting of a 0-place predicate, i.e., a proposition letter, p. The valuation of p, $V(p)$, is a function from worlds to sets of 0-tuple of individuals. For any world w, $V(p)(w)$ is either $\{\emptyset\}$ (i.e., true) or $\emptyset$ (i.e., false). The update $s[\![p]\!]$ of a state s with p then consists of the assignments i in s such that p is true in $i(v)$. Clearly, this may involve a genuine update of s without turning it into the absurd state.

Let us briefly go through some examples. First, consider *A man walks*, translated as $\exists x(Mx \wedge Wx)$. Interpreted in a state s, this sentence gives us extensions j of assignments i in s with a valuation for x such that $j(x)$ is a man who walks in $j(v)$ $(= i(v))$. Like in extensional *(E)DPL*, the introduction of x in the domain of discourse makes x available as an antecedent for future anaphoric reference, and, clearly, the resulting state entails that x is a man who walks. (If s' is the state that results from the update, then s' ascribes x a property $\mathcal{R}_x^{s'}$ which is such that $\forall w \in W$: $\mathcal{R}_x^{s'}(w) \subseteq (F(M)(w) \cap F(W)(w))$.)

Next, consider the sentence *No man walks*, translated as $\neg\exists x(Mx \wedge Wx)$. If we interpret this sentence in a state s, we keep the assignments $i \in s$ that cannot be extended with a valuation j for x such that $j(x)$ is a man in $i(v)$ who walks in $i(v)$. In other words, we keep those $i \in s$ such that in $i(v)$ no men walk. Another example is the donkey sentence *If a farmer owns a donkey he beats it*, translated as $\exists x(Fx \wedge \exists y(Dy \wedge Oxy)) \rightarrow Bxy$. The interpretation of this sentence in a state s preserves the assignments $i \in s$ such that for every extension j of i with a valuation for x and y such that $j(x)$ is a farmer in $i(v)$ and $j(y)$ a donkey that $j(x)$ owns in $i(v)$, $j(x)$ beats $j(y)$ in $i(v)$. So, these are those $i \in s$ that assign a world $i(v)$ to v in which every farmer beats every donkey he owns.

As a last example, consider a sentence with a binary quantifier: *Most boys who have met her like her*, $Most\, y(By \wedge Myx)(Lyx)$. Interpreted in a state s this sentence gives in return all those $i \in s$ such that most extensions of i with a valuation for y that is a boy who has met $i(x)$ in $i(v)$, assign y an individual that also likes $i(x)$ in $i(v)$. In other words, the update preserves the assignments $i \in s$ such that in $i(v)$ most boys who have met $i(x)$ like her.

In order to conclude the exposition of *MDPL*, let us consider the relation between the information a state s has about the values of a sequence of variables $x_1, \ldots, x_n$ and the truth of atomic formulas with free variables $x_1, \ldots, x_n$ (the definition of truth remains that of section 3):

Fact 4.1
For all relation expressions R:
- $s \models Rx_1 \dots x_n$ iff $\forall w \in W$: $\mathcal{R}^s_{x_1,\dots,x_n}(w) \subseteq F(R)(w)$

A state s ascribes the tuple of variables $x_1, \dots, x_n$ an intensional relation $\mathcal{R}^s_{x_1,\dots,x_n}$ which is at least as specific as the interpretation of an n-ary relation expression R which is true of $x_1, \dots, x_n$ in s.

Epistemic modalities

Having turned (extensional) *EDPL* into intensional *MDPL*, we can properly introduce Veltman's epistemic operator $\Diamond$ ('might'). I add an epistemic operator $\Box$ ('must'), which is the dual of $\Diamond$, and an epistemic connective $\Rightarrow$ ('if ... then surely ...').[19] The interpretation of these operators is defined as follows:

Definition 4.4 (Epistemic operators)
- $s[\![\Diamond\phi]\!] \quad = \{i \in s \mid s[\![\phi]\!] \neq \emptyset\}$
- $s[\![\Box\phi]\!] \quad = \{i \in s \mid s \models \phi\}$
- $s[\![\phi \Rightarrow \psi]\!] = \{i \in s \mid s[\![\phi]\!] \models \psi\}$

A formula $\Diamond\phi$ tests whether update with ϕ is still possible in a state s. If update with ϕ does not produce the absurd state, then s accepts $\Diamond\phi$, that is, $s[\![\Diamond\phi]\!]$ is s then. If ϕ is false in s, then $\Diamond\phi$ is rejected, and the absurd state results. A formula $\Box\phi$ tests whether ϕ is already true in s. If ϕ is true in s, then $s[\![\Box\phi]\!]$ $(= s)$ is true in s, and if it is not, $s[\![\Box\phi]\!]$ is false in s. A formula $\phi \Rightarrow \psi$ tests whether ψ turns out true if we update with ϕ. If ψ is true in the update of s with ϕ then $\phi \Rightarrow \psi$ is true in s, false otherwise.

I will not expand upon these epistemic operators and their behaviour here.[20] I refer to Veltman [1981, 1985] and Landman [1986] for an extensive discussion of the epistemic operators, alternative definitions and potential deficiencies of the analyses. In this section I will only discuss some examples involving *might* and state some characteristic facts. In section 4.3 the use of these operators is reassessed.

On the present definition of *might*, which is Veltman's definition cast in the framework of *MDPL*, *might* typically serves to express the partiality of the information we may have about the values of variables and the ways in which our information may increase. Consider the following examples:

19. As we will see below, these natural language paraphrases are not very satisfactory, but they will do for the moment.
20. But is is appropriate to warn the reader that, for the time being, it is better to disregard embedded use of the epistemic operators. The embedded use presents typical problems which can only be properly addressed after we have more fully appreciated the use of these operators in section 4.3.

(40) A man is raking the leaves over there in the park. Maybe he found your bracelet. .... He did not find your bracelet.
$\exists x(Mx \wedge Rx) \wedge \Diamond Fxy \wedge \neg Fxy$
(41) Somebody is knocking at the door. ... It might be John. ... It's Mary.
$\exists x Kx \wedge \Diamond(x = j) \wedge x = m$
(42) Somebody is knocking at the door. ... It's Mary. ... *It might be John.
$\exists x Kx \wedge x = m \wedge \Diamond(x = j)$

The interpretation of the first sentence in the first example yields a state which ascribes x the property of being a man who is raking the leaves in the park. If we have no information to the contrary, this is consistent with the property of finding your bracelet, so *Maybe he found your bracelet* is acceptable (true) in that state. Of course, this does not preclude that one may find out later that he did not find your bracelet and the whole example, therefore, is consistent. Notice that this example does not allow us to conclude that a man who is raking the leaves in the park did find your bracelet, as it was shown to do in *EDPL*. What it implies at most is that we consider it a possibility that a man who rakes the leaves finds your bracelet, but we may also consider the possibility that no such man finds it.

The other two examples can now be more fully analyzed then in *US*, since *MDPL* also accounts for the anaphoric relationships involved.[21] The first sentence in both examples presents an object which has the property of knocking at the door. Without information to the contrary, we may record that, as far as we know, this object might be John, even though we later may get informed that it is Mary. On the other hand, as soon as we are informed that it is Mary, we can not, of course, say that it might be John (that is, assuming that we know that Mary is not John).

As the following fact shows, the three epistemic operators are closely related:

Fact 4.2

- $\Diamond\phi \Leftrightarrow \neg\Box\neg\phi$
  $\Box\phi \Leftrightarrow \neg\Diamond\neg\phi$
- $(\phi \Rightarrow \psi) \Leftrightarrow \Box(\phi \rightarrow \psi)$

As indicated above, $\Diamond$ (*might*) and $\Box$ (*must*) are duals. An epistemic implication $\phi \Rightarrow \psi$ (*If A then surely B*) is equivalent with $\Box(\phi \rightarrow \psi)$ (*It must be that if A then B*). However, it is *not* equivalent with $\phi \rightarrow \Box\psi$ (*If A then it must be that B*). In a state s in which $\phi \rightarrow \psi$ is not true, and, hence, $\phi \Rightarrow \psi$ is false, it is still possible to update with $\phi \rightarrow \Box\psi$. The update of such a state with $\phi \rightarrow \Box\psi$ equals its update with $\neg\neg\phi$.

21. The analysis is not complete of course, since proper names have not been introduced. Notice that it makes no difference for this example whether proper names are treated as rigid or non-rigid designators.

The following fact shows that iterations of epistemic operators are always redundant:

Fact 4.3

- $\Diamond\Diamond\phi \Leftrightarrow \Diamond\phi \Leftrightarrow \Box\Diamond\phi$
  $\Diamond\Box\phi \Leftrightarrow \Box\phi \Leftrightarrow \Box\Box\phi$

That concludes the elementary exposition of *MDPL*. We have seen that *EDPL* is easily made intensional. In the resulting system of *MDPL* the values of variables are ascribed properties by information states and information states contain the information that tuples of them stand in certain relations, intensionally conceived. Furthermore, Veltman's epistemic operators are interpreted in an intuitively satisfactory way. They are readdressed in section 4.3.

4.2 The lattice of information states

This section and the following ones may serve to substantiate the claim in section 2.3 that the notion of information employed in *MDPL* is of a more ambitious nature than that of a mere means to account for anaphoric relationships. This section pays closer attention to the *MDPL* structure of information states and to some fundamental properties of *MDPL* interpretation in that structure. We will see that this structure is a lattice and that information update corresponds to a meet operation on that lattice.

The results from this section are employed in section 4.3, in which it is shown that the structure of information states may serve to model the information that language users have and exchange, and in section 4.4, which shows that the kind of information employed in *MDPL* can be properly conceived of as information about partial objects.

I start reviewing the structure of information states. This structure is shown to be a non-distributive lattice. Next we will see that the interpretation function $[\![\ ]\!]$ does distribute over the lattice. We will see, moreover, that *MDPL* interpretation can be defined in terms of the lattice's meet operation (which is labeled the product operation) and separately defined denotations of *MDPL* formulas. *MDPL*'s dynamics then appears only to consist in its (asymmetric) definedness conditions.

The information lattice

In section 3.1, it was observed that the structure $\langle S, \leq \rangle$ is a partial order. In fact the structure is a lattice, that is, for every two states s and t there is a unique weakest state t' such that $s \leq t'$ and $t \leq t'$ and a unique strongest state s' such that $s' \leq s$ and $s' \leq t$. The weakest common update of s and t I will call their (information) product, and the strongest common 'downdate' of s and t I will call their common ground. These two states can be defined as follows (the definition of $\leq$ is repeated here for convenience):

Definition 2.3 (Update)

- $s \leq t$, iff $D(s) \subseteq D(t)$ and $\forall i \in t$: $i \gtrdot s$

Definition 4.5 (State product and common ground)

- $s \wedge t = \{i \in D^{D(s) \cup D(t)} \mid i \gtrdot s \text{ and } i \gtrdot t\}$
- $s \vee t = \{i \in D^{D(s) \cap D(t)} \mid i \lessdot s \text{ or } i \lessdot t\}$

The product of two states s and t contains the combined information contained in s and t. It contains information about the values of all the variables s or t contain information about, and it excludes valuations of these variables which are excluded by s or t. So, $s \wedge t$ contains the information that s or t has about the values of the variables in their combined domain. The common ground of s and t contains just the information s and t agree upon. It contains only information about the values of variables both s and t contain information about, and it only excludes assignments which are excluded by s and t. So, $s \vee t$ contains the information that both s and t have about the values of the variables in their shared domain.

The following fact shows that the product $s \wedge t$ is the weakest common update of s and t, and that the common ground of s and t is the strongest common downdate:

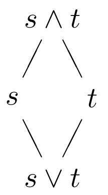
Fact 4.4

1. $(s \vee t) \leq s \leq (s \wedge t)$ and $(s \vee t) \leq t \leq (s \wedge t)$
2. $\forall s'$: if $s' \leq s$ and $s' \leq t$ then $s' \leq (s \vee t)$
   $\forall s'$: if $s \leq s'$ and $t \leq s'$ then $(s \wedge t) \leq s'$[22]

So, the information product of s and t can be properly conceived of as indicating the joint information of s and t, the weakest state they might both arrive at by information update. Similarly, the common ground of s and t contains the information s and t agree upon. It is the strongest state of which both are an update. The information product and common ground of two states is depicted by the following snapshot of the lattice of information states:

---

22. Proof of the first clause: By the definition of $\wedge$, $\forall i \in (s \wedge t)$: $i \gtrdot s$ and $i \gtrdot t$, so $s \leq (s \wedge t)$ and $t \leq (s \wedge t)$. By the definition of $\vee$, for all $j \in s$ or $j \in t$: $j \gtrdot (s \vee t)$, so $(s \vee t) \leq s$ and $(s \vee t) \leq t$. Proof of the second clause: (i) Let $s' \leq s$ and $s' \leq t$. Then $D(s') \subseteq D(s)$ and $D(s') \subseteq D(t)$, so $D(s') \subseteq D(s \vee t)$. Now consider an $i \in (s \vee t)$, i.e., $D(i) = D(s) \cap D(t)$ and $i \lessdot s$ or $i \lessdot t$. If $i \lessdot s$, then, since $s' \leq s$ and $D(s') \subseteq D(i)$, $i \gtrdot s'$. If $i \lessdot t$, then, since $s' \leq t$ and $D(s') \subseteq D(i)$, $i \gtrdot s'$. So, for any $i \in (s \vee t)$: $i \gtrdot s'$, and, hence, $s' \leq (s \vee t)$. (ii) Let $s \leq s'$ and $t \leq s'$. Then $D(s) \subseteq D(s')$ and $D(t) \subseteq D(s')$, so $D(s \wedge t) \subseteq D(s')$. Now consider a $j \in s'$ and the restriction i of j to $D(s \wedge t)$, i.e., $D(i) = D(s \wedge t)$ and $i \leq j$. Since $D(s) \subseteq D(i)$ and $s \leq s'$, $i \gtrdot s$. Similarly, since $D(t) \subseteq D(i)$ and $t \leq s'$, $i \gtrdot t$. So, for any $i \in s'$: $i \gtrdot (s \wedge t)$, and, hence, $(s \wedge t) \leq s'$.



Interpretation in the information lattice

The lattice $\langle S, \leq \rangle$ is not distributive, that is, $s \wedge (t \vee t')$ and $(s \wedge t) \vee (s \wedge t')$ are not in general the same, nor are $s \vee (t \wedge t')$ and $(s \vee t) \wedge (s \vee t')$.[23] However, the *MDPL* interpretation function does distribute over $\wedge$ and $\vee$ (I only assume the interpretation of the basic language of *EDPL* here; for simplicity's sake quantifiers are disregarded here and epistemic modals are reintroduced below):

Fact 4.5 (Distributivity in $\langle S, \leq \rangle$)
If defined,

- $s\llbracket\phi\rrbracket \wedge t\llbracket\phi\rrbracket = (s \wedge t)\llbracket\phi\rrbracket$
- $s\llbracket\phi\rrbracket \vee t\llbracket\phi\rrbracket = (s \vee t)\llbracket\phi\rrbracket$[24]

In a snapshot:

---

23. Let $s, t \in S^{\{x\}}$, $s = \{i\}$, $t = \{j\}$, $i(x) \neq j(x)$ and $t' = \bot^{\emptyset}$. Then $(s \wedge t) \vee (s \wedge t') = \bot^{\{x\}} \vee \bot^{\{x\}} = \bot^{\{x\}}$, while $s \wedge (t \vee t') = \{i\} \wedge \top^{\emptyset} = \{i\}$. Furthermore, in that case $(s \vee t) \wedge (s \vee t') = \{i, j\} \wedge \top^{\emptyset} = \{i, j\}$, while $s \vee (t \wedge t') = \{i\} \vee \bot^{\{x\}} = \{i\}$.

24. Proof. First observe that $s\llbracket\phi\rrbracket$ and $t\llbracket\phi\rrbracket$ are defined iff $D(s)$ and $D(t)$ are in $\mathcal{D}(\phi)$ iff (since $\mathcal{D}(\phi)$ is continuous and closed under $\cap$ and $\cup$) $D(s \wedge t)$ and $D(s \vee t)$ are in $\mathcal{D}(\phi)$ iff $(s \wedge t)\llbracket\phi\rrbracket$ and $(s \vee t)\llbracket\phi\rrbracket$ are defined. So, assuming that $s\llbracket\phi\rrbracket$ and $t\llbracket\phi\rrbracket$ are defined, we know that $(s \vee t)\llbracket\phi\rrbracket$ and $(s \wedge t)\llbracket\phi\rrbracket$ are defined. (This assumption is indicated by * in the proof below.)
That $s\llbracket\phi\rrbracket \vee t\llbracket\phi\rrbracket = (s \vee t)\llbracket\phi\rrbracket$ and $s\llbracket\phi\rrbracket \wedge t\llbracket\phi\rrbracket = (s \wedge t)\llbracket\phi\rrbracket$, if defined, is proved by induction on the complexity of ϕ. I give the outlines of the proof of the distribution over $\vee$. (Again, an existentially quantified formula $\exists x\phi$ is conceived of as the conjunction $\exists x \wedge \phi$, where $s\llbracket\exists x\rrbracket = s[x]$, if defined.)

$s\llbracket Rx_1 \ldots x_n\rrbracket \vee t\llbracket Rx_1 \ldots x_n\rrbracket = \{i \in D^{D(s) \cap D(t)} \mid i < s\llbracket Rx_1 \ldots x_n\rrbracket \text{ or } i < t\llbracket Rx_1 \ldots x_n\rrbracket\} =^*$
$\{i \in D^{D(s) \cap D(t)} \mid i < s \text{ and } \{i\} \models Rx_1 \ldots x_n \text{ or } i < t \text{ and } \{i\} \models Rx_1 \ldots x_n\} =$
$\{i \in D^{D(s) \cap D(t)} \mid i \in (s \vee t) \text{ and } \{i\} \models Rx_1 \ldots x_n\} = (s \vee t)\llbracket Rx_1 \ldots x_n\rrbracket$

$s\llbracket\exists x\rrbracket \vee t\llbracket\exists x\rrbracket = s[x] \vee t[x] = \{i \in D^{(D(s) \cap D(t)) \cup \{x\}} \mid i < s[x] \text{ or } i < t[x]\} =^*$
$\{i \in D^{(D(s) \cap D(t)) \cup \{x\}} \mid \exists j \leq_{\{x\}} i\text{: } j < s \text{ or } \exists j \leq_{\{x\}} i\text{: } j < t\} =$
$\{i \in D^{(D(s) \cap D(t)) \cup \{x\}} \mid \exists j \leq_{\{x\}} i\text{: } j \in (s \vee t)\} = (s \vee t)[x] = (s \vee t)\llbracket\exists x\rrbracket$

$s\llbracket\neg\phi\rrbracket \vee t\llbracket\neg\phi\rrbracket = \{i \in D^{D(s) \cap D(t)} \mid \exists j \geq i\text{: } j \in s \text{ and } \{j\} \dashv \phi \text{ or } \exists j \geq i\text{: } j \in t \text{ and } \{j\} \dashv \phi\} =^*$
$\{i \in D^{D(s) \cap D(t)} \mid \exists j \geq i\text{: } j \in s \text{ and } \{i\} \dashv \phi \text{ or } \exists j \geq i\text{: } j \in t \text{ and } \{i\} \dashv \phi\} =$
$\{i \in D^{D(s) \cap D(t)} \mid i < s \text{ or } i < t \text{ and } \{i\} \dashv \phi\} = (s \vee t)\llbracket\neg\phi\rrbracket$

$s\llbracket\phi \wedge \psi\rrbracket \vee t\llbracket\phi \wedge \psi\rrbracket = s\llbracket\phi\rrbracket\llbracket\psi\rrbracket \vee t\llbracket\phi\rrbracket\llbracket\psi\rrbracket =_{ih} (s\llbracket\phi\rrbracket \vee t\llbracket\phi\rrbracket)\llbracket\psi\rrbracket =_{ih} (s \vee t)\llbracket\phi\rrbracket\llbracket\psi\rrbracket = (s \vee t)\llbracket\phi \wedge \psi\rrbracket$

$$\begin{array}{ccc} & (s \wedge t)[\![\phi]\!] & \\ / & & \backslash \\ s[\![\phi]\!] & & t[\![\phi]\!] \text{ if defined} \\ \backslash & & / \\ & (s \vee t)[\![\phi]\!] & \end{array}$$

The distributivity of $[\![\ ]\!]$ in $\langle S, \leq \rangle$ shows that the contents of a formula ϕ in *MDPL* are, in a certain sense, state independent. The product (ground) of the update of two states with ϕ contains precisely the same information as the update of the product (ground) itself. As we will see presently, we can even define these contents separately.

Static MDPL denotations

As we have seen above, an existentially quantified formula $\exists x\phi$ can be conceived of as the conjunction of $\exists x$ with ϕ, where $s[\![\exists x]\!] = s[x]$ (if $x \notin D(s)$). Using the label (*MDPL*) atom for both atomic formulas and such 'wild' quantifiers, we can associate with the *MDPL* atoms states which can be conceived of as their denotations:

Definition 4.6 (Denotation of MDPL atoms)

- $[Rx_1 \ldots x_n] = \{i \in D_W^{\{x_1,\ldots,x_n\}} \mid \langle i(x_1), \ldots, i(x_n)\rangle \in F(R)(i(v))\}$
- $[\exists x] \qquad = \top^{\{x\}}$

(Remember that $D_W^X = \{i \cup j \mid i \in W^{\{v\}} \ \& \ j \in D^X\}$.) An atomic formula $Rx_1 \ldots x_n$ denotes a state with a domain consisting of $x_1, \ldots, x_n$, and which consists of assignments i such that $i(v)$ is a world where $i(x_1), \ldots, i(x_n)$ stand in the relation R. The quantifier $\exists x$ simply denotes the state of no information about x. In terms of these denotations, the *MDPL* interpretation of atomic and existentially quantified formulas can be (equivalently) stated as follows:

Fact 4.6

- $s[\![Rx_1 \ldots x_n]\!] = s \wedge [Rx_1 \ldots x_n]$ if $x_1, \ldots, x_n \in D(s)$
- $s[\![\exists x]\!] \qquad = s \wedge [\exists x] \qquad$ if $x \notin D(s)$

So, the *MDPL* interpretation of an atom a (an atomic *MDPL* act) is in fact a product operation, i.e., the operation that for any input information state s yields the product of s and the denotation of a.[25]

Also compound formulas can be associated with specific contents. First, let, for any $s \in S^X$, the complement $\overline{s}$ of s be $\{i \in D^X \mid i \notin s\}$. Next, let X_ϕ be the smallest

25. We can give similar definitions of *PL* and *DPL* existential quantification. Let s be the interpretation $[\![\phi]\!]$ of a formula ϕ in *PL*, i.e., the set of (total) assignments verifying ϕ in *PL*. Then $[\![\exists x\phi]\!]$ in *PL* is $(\bot^{V\setminus\{x\}} \vee s) \wedge \top^{\{x\}}$, which is also the *DPL* result $s[x]$ of reinstantiating x in s. This result is obtained by, first, removing x from the domain of s by means of $\bot^{V\setminus\{x\}} \vee s$, and, next, adding x again by taking the product with $\top^{\{x\}}$. Thus conceived, the *PL* and *DPL* quantifier $\exists x$ is associated with the instruction "forget about x, and take an x".

set of variables in the domain $\mathcal{D}(\phi)$ of ϕ, if there is such a set. (If $\mathcal{D}(\phi)$ is empty, we may as well use the empty set.). Then the denotation of compound formulas can be defined in terms of the denotations of their atoms in the following way (we may disregard $[\exists x\phi]$ now, which is $[\exists x \wedge \phi]$):

Definition 4.7 (Denotation of MDPL compounds)

- $[\neg\phi] = \overline{\bot^{X_\phi} \vee [\phi]}$
- $[\phi \wedge \psi] = [\phi] \wedge [\psi]$

We then get the following fact:

Fact 4.7

If defined,

- $s[\![\phi]\!] = s \wedge [\phi]$[26]

It appears that *MDPL* is almost a static semantics, since the *MDPL* update of a state with a formula ϕ can be cast in terms of the (static) denotation of ϕ and the (associative) product operation. All that prevents it from collapsing into a completely static semantics are the definedness conditions. These ensure that variables are introduced *before* they are referred back to.

Digression

Since the product operation is defined for all pairs of states, and since all *MDPL* atoms have a defined denotation, it is possible to speculate about adopting the definition of $[\![\ ]\!]$ in terms of $[\ ]$ and $\wedge$ and *dropping* the definedness conditions. This would have, among others, the following effects:

In the first place $\exists x_j Rx_1 \ldots x_j \ldots x_n$ and $Rx_1 \ldots x_j \ldots x_n$ would turn out to be fully equivalent then (if $x \in D(s)$, $s[x] = s \wedge \top^{\{x\}} = s$). So, in that case indefinites would really *be* free variables, which is reminiscent of "Lewis' philosophy". In the second place, update would be preserved (clearly, $s \leq (s \wedge [\phi])$), but $[\![\ ]\!]$ no longer distributes over $\vee$. (If $x \in D(s)$ and $x \notin D(t)$, then $s[x] \vee t[x] = s \vee t[x] \neq (s \vee t)[x]$. Still, $(s \vee t)[x] \leq s \vee t[x]$.)

However, I do not think that such a total (i.e., non-partial) semantics would be

---

26. Proof: by induction on the complexity of ϕ. The only non-trivial step in the proof concerns negation. The inductive proof that $s[\![\neg\phi]\!] = s \wedge [\neg\phi]$ uses the fact that X_ϕ, the smallest domain of ϕ, if defined, is a subset of the domain $D([\phi])$ of $[\phi]$. (Proof: by induction on the complexity of the normal binding form of ϕ. (i) $X_{Rx_1 \ldots x_n} = \{x_1, \ldots, x_n\} = D([Rx_1 \ldots x_n])$; (ii) $X_{\neg\phi} = X_\phi =_{ih} X_\phi \cap D([\phi]) = D([\neg\phi])$; (iii) $X_{\exists x\phi} = X_\phi \backslash \{x\} \subseteq_{ih} \{x\} \cup D([\phi]) = D([\exists x\phi])$; (iv) $X_{(\phi\wedge\psi)} =$ (normal binding form; continuity of $\mathcal{D}$) $X_\phi \cup X_\psi \subseteq_{ih} D([\phi]) \cup D([\psi]) = D([\phi \wedge \psi])$.) So, if $s[\![\neg\phi]\!]$ is defined, $X_\phi \subseteq D([\phi])$ and $\bot^{X_\phi} \vee [\phi] = \{i \in D^{X_\phi} \mid i \lessdot [\phi]\}$. Hence, $[\neg\phi] = \overline{\bot^{X_\phi} \vee [\phi]} = \{i \in D^{X_\phi} \mid i \not\lessdot [\phi]\} = \{i \in D^{X_\phi} \mid i \not\lessdot (\{i\} \wedge [\phi])\} =_{ih} \{i \in D^{X_\phi} \mid i \not\lessdot \{i\}[\![\phi]\!]\} = \{i \in D^{X_\phi} \mid \{i\} \dashv \phi\}$. Since, by definedness, $X_\phi \subseteq D(s)$, $s \wedge [\neg\phi] = \{i \in s \mid i \gtrdot [\neg\phi]\} = \{i \in s \mid \exists j \leq i\text{: } j \in D^{X_\phi}\ \&\ j \dashv \phi\}$, which, by lemma 1, is $\{i \in s \mid i \dashv \phi\}$, which, by fact 2.8, is $s[\![\neg\phi]\!]$.

a viable alternative. Notice that the denotation of $\neg\phi$ is not 'properly' defined in terms of that of ϕ, since this definition critically refers to syntactic properties of ϕ, i.e., to the (syntactically defined) domains of ϕ. (Notice, as well, that the domains of ϕ can not be determined semantically anymore when the definedness conditions are dropped.) I do not think that it is possible to improve upon this definition. Notice, first, that it is crucial for a proper analysis of negation to distinguish embedded free variables occurrences from existentially bound ones, just in order to properly distinguish the meaning of, for instance, $\neg\exists xFx$ and $\neg Fx$. Now, if the definedness conditions are dropped and, hence, $\exists xFx$ is fully equivalent with Fx, then there is no hope of giving a proper, purely semantic, interpretation of negation.

End of digression.

To sum up the results, *MDPL*'s structure of information states, and, hence, that of *EDPL*, is a non-distributive lattice. The meet of two states is labeled the information product of these states and the join their common ground. Next, we saw that the *MDPL* interpretation of a formula ϕ in a state s can be stated as the product of s and the separately defined denotation of ϕ. Still, *MDPL* is not a static semantics after all, because of the presence of the definedness conditions.

4.3 Aspects of a theory of information exchange

In this section I will argue that Veltman's epistemic operators can be understood better within a system of information exchange, a system which properly extends the recipient oriented update semantics by taking the speaker and the information exchange between speaker and recipient into account. I will sketch the rudiments of such an extended system and show how the epistemic operators may be analyzed in that system in a relatively satisfactory way.

Interactive aspects of epistemic operators

MDPL and *US* describe the information that people may have about the world and about the values of variables and give an account of how this information is updated by interpreting sentences. If we focus on this goal of information update, epistemic modals appear to play a very marginal role. They do not incite any genuine update of information about the world or about the values of variables. The operators serve to express global properties of information states and they can only be either true or false in such information states. As Veltman puts it: "If you learn a sentence ϕ of $L_0^{\mathcal{A}}$ [the language without *might*, PD], you learn that the real world is one of the worlds in which the proposition expressed by ϕ holds: the real world is a ϕ-world. But it would be nonsense to speak of '*might* ϕ-worlds'. If ϕ might be true, this is not a property of the world but of your knowledge of the world." ([1990, p. 12])

Veltman's observation raises the question of what intuitive sense it makes to hear or 'learn' that it might be the case that ϕ. On the present, and Veltman's, definition, if someone hears that it might be that ϕ, he is being told, so to speak, that his information is consistent with the information conveyed by ϕ. However, this can at most be part of the analysis of a statement that it might be the case that ϕ. If *might* ϕ says something about a state of information, then, it seems, it says something about the *speaker's* state of information in the first place. When Veltman describes the interpretation of *might* ϕ, he says: "..., all you can do when told that it might be the case that ϕ is to agree or to disagree." (Veltman, [1990, p. 8]) Clearly, if in view of the statement that it might be the case that ϕ there is anything to agree or disagree upon, then what has to be agreed or disagreed upon is whether the state of the hearer, like the state of the speaker, can be consistently updated with ϕ. Thus conceived, the statement that it might be the case that ϕ serves to establish agreement between speaker and hearer about the possibility of ϕ, that is, it aims to establish that both consider the proposition expressed by ϕ possible. Such a statement makes sense of course. However, such an interpretation of *might* goes beyond the scope of the purely recipient oriented update semantics which we have presented sofar, and it should be cast within some model of information exchange.

Something similar holds for the operator $\Box$. Restricting ourselves to the hearer's interpretation of a formula $\Box\phi$, it only tells him, one might say, that he has the information that ϕ, something which is either true or false and not particularly useful when update of information about the world is at issue. However, if a statement that $\Box\phi$, like $\Diamond\phi$, is understood within the context of information exchange, it can be conceived of as testing whether the hearer, like the speaker, agrees that ϕ is the case, which does make sense. (Thus conceived, the $\Box$ operator may be better paraphrased by means of the sentential operator 'of course', instead of 'must'.)

It appears, then, that we can make more sense of the epistemic operators $\Diamond$ and $\Box$ by conceiving of them as interactive operators which are used to establish agreement between speaker and hearer about the possibility or truth of certain propositions. However, with this conception of the meaning of these operators we have left the realm of interpretation simpliciter, which is that of pure information update, and we have got involved in the analysis of information exchange.

In the remainder of this section I will give an outline of a system of information exchange which is a proper extension of *MDPL*, and I will show how some of the interactive aspects of the epistemic modals can be accounted for in that system. Furthermore, I will present a solution to some problems, already alluded to above, which pertain to the embedded use of the modals. The aims, however, will remain relatively modest. For one thing, the system does not allow a speaker to convey information which conflicts with information the hearer has. In order to solve such disagreement, some kind of belief revision seems to be required and revision falls beyond the scope of the present undertaking. Furthermore I will not attempt to model ((ex-)change of) information about the information of others or about the exchange situation itself. So, the facts established about exchange situations by means of epistemic statements are not reflected in the states of the exchanging participants. For these reasons, then, the implications the ensueing discussion has may be best conceived of as implications for a theory of information exchange which is yet to be developed.

Proper information exchange

In what follows, the *MDPL* information states will be used to model the information of agents which are involved in situations of information exchange. I will restrict attention to two agent situations, typically that of a speaker and a hearer, but the models can be easily extended to cover situations with a larger number of agents. First, I describe the results of information exchange brought about by the utterance of *MDPL* formulas without epistemic operators.

What we will be dealing with are, basically, situations which are pairs of states, s and t, both non-absurd, which will be written as $\langle s\ t\rangle$. If not indicated otherwise, it is assumed that s is the state of the speaker, and t that of the hearer. The information the two participants have together is contained in the product $s \wedge t$ of their information states, but, of course, it is not required that one (or both) of them has this information. The idealization I make is that the agents in an exchange situation aim at sharing one another's information, and, thus, to profit both from the information both have together.

If $\langle s\ t\rangle$ is an (initial) exchange situation, then $\langle s\ t\rangle[\![\phi]\!]$ is the situation that results from accepting ϕ in that situation. This situation will be defined to be the situation consisting of the states that result from updating s and t with ϕ respectively, that is, the situation $\langle s[\![\phi]\!]\ t[\![\phi]\!]\rangle$. Clearly, not any update of an exchange situation will count as a genuine case of information exchange. For a proper exchange of information it is required, in the first place, that the speaker conveys information she actually has. This is, of course, Grice's maxim of quality. So, since it is assumed that the state s in $\langle s\ t\rangle$ is that of the speaker, for $\langle s\ t\rangle[\![\phi]\!]$ to constitute a case of information exchange it is required that $s \models \phi$.

In the second place, for an update of an exchange situation to constitute a proper case of information exchange it is furthermore required that the hearer does not accept information inconsistent with the information he has. This restriction, in fact, indicates a limit of the present first order exchange model. If the speaker attempts to exchange the information that ϕ, and the hearer has information to the contrary, then the exchange is simply taken to come to a halt. For the exchange to proceed in such a situation, a higher order discussion may be required about evidence for information and justification of evidence, as well as some method of belief revision. Since, as I indicated above, such issues of higher order information and belief revision fall beyond the scope of the present undertaking, we just have to settle for expelling the occurrence of inconsistency of information.

Putting things together, the notion of proper information exchange is defined as follows:

Definition 4.8 (Proper information exchange (1))
If ϕ contains no epistemic operators, then

- $\langle s\ t\rangle[\![\phi]\!] = \langle s[\![\phi]\!]\ t[\![\phi]\!]\rangle$ if $s \models \phi$ and $t \not\dashv \phi$

Like I said, the first side condition, that $s \models \phi$, captures Grice's maxim of quality within the present framework. The second side condition, that $t \not\dashv \phi$, requires ϕ to be consistent with the state of the hearer. If this condition is not satisfied, we will assume that the hearer objects and that, for instance, a discussion starts which falls beyond the present framework.

The above definition of proper information exchange validates the following fact.

Fact 4.8
If $\langle s\ t\rangle[\![\phi]\!]$ is defined, then:

- $s \wedge t \sqsubseteq s[\![\phi]\!] \wedge t[\![\phi]\!]$[27]

This fact says that after a proper information exchange, speaker and hearer together have information that is licensed by the information they had together before the exchange. In order words, proper information exchange does not involve addition of information 'out of the blue'. In my opinion, fact 4.8 expresses a requirement which any theory of information exchange should observe, that is, as long as it does not take second order information and information about the exchange situation into account. Disregarding these kinds of information, proper information exchange should not, and in the present framework does not, license jumping to conclusions.

Digression

If we assume that the side conditions on proper information exchange apply to each exchange act, that is, to each separate utterance, then the exchange of information brought about by means of $\exists x\phi$ and ψ is not fully equivalent with an exchange by means of $\exists x(\phi \wedge \psi)$. Let us write $\phi.\ \psi$ for the successive utterance of ϕ and ψ, and define $\langle s\ t\rangle[\![\phi.\ \psi]\!]$ to be $\langle s\ t\rangle[\![\phi]\!][\![\psi]\!]$. Then we find that:

Fact 4.9

- $\langle s\ t\rangle[\![\exists x\phi.\ \psi]\!] \not\Leftrightarrow \langle s\ t\rangle[\![\exists x(\phi \wedge \psi)]\!]$

The reason is that, for the first exchange $\langle s\ t\rangle[\![\exists x\phi.\ \psi]\!]$ to be proper, it is required that $s \models \exists x\phi$ and $s[\![\exists x\phi]\!] \models \psi$ (i.e., $s \models \exists x\phi \rightarrow \psi$). However, for the second exchange $\langle s\ t\rangle[\![\exists x(\phi \wedge \psi)]\!]$ to be proper it is only required that $s \models \exists x(\phi \wedge \psi)$.

27. Proof. By fact 4.5 $s[\![\phi]\!] \wedge t[\![\phi]\!]$ is $(s \wedge t)[\![\phi]\!]$. Since $s \models \phi$ (proper information exchange), and using fact 2.7, we find that $s \wedge t \models (s \wedge t)[\![\phi]\!]$.

The difference between $\exists x\phi.\ \psi$ and $\exists x(\phi \wedge \psi)$ in situations of information exchange may be used to explain a difference observed between the following examples (cf., Evans [1977, 1980], Kadmon [1987, 1990], among others):

(43) There is a doctor in London. He is Welsh.

(44) There is a doctor in London who is Welsh.

Within the present framework there is no difference between the situations that result from a proper use of these examples, but there is a difference in the side conditions on the speaker's state. For an utterance of example 43 to constitute a case of proper information exchange, the speaker is required to have the information that if somebody is a doctor in London, he is Welsh. There is no such requirement for a proper use of example 44.

Of course, under normal circumstances, it is quite unlikely that people have information that if somebody is a doctor in London then he is Welsh. In fact, if someone says that there is a doctor in London and next tells us that he is Welsh, we are inclined to assume that the speaker is not talking about some arbitrary doctor in London, nor that she has the information that any arbitrary doctor in London is Welsh, but that she has some specific doctor in London in mind. Whereas the indefinite *a doctor* introduces a new discourse referent to the hearer, like other indefinites do, in this example it is taken to refer to a discourse referent which is already familiar to the speaker. In other words, the speaker appears to inform the hearer about the value of a variable about which she has the information that it is a Welsh doctor in London, but which is new to the hearer.[28]

The *MDPL* exchange model can be extended with such specific uses of main clause indefinites simply by relaxing the definedness conditions. If an exchange situation $\langle s\ t\rangle$ is updated with a formula $\exists x\phi$, we may require that the variable x is not in the domain of the hearer's state t, but leave out a similar requirement on the speaker's state. It then depends on the speaker's state whether the quantifier is used specifically or non-specifically. In case x is not in the domain of her state s, then the quantifier $\exists x$ is used non-specifically. Otherwise, if x is in the domain of her state s, then the quantifier is used specifically.[29]

Clearly, the present proposal remains far from a full-blown treatment of the specific/non-specific contrast, and I will not elaborate on it here. Suffice it to note that the unlikeliness that the side condition associated with a non-specific reading

28. Notice that, in contradistinction with the proposals of Evans and Kadmon, on such a specific interpretation of the indefinite *a doctor*, the hearer need not associate any uniqueness implications with the anaphorically related pronoun. If there is any difference on the hearer's part between a specific or a non-specific interpretation of example 43, it must be taken to reside on a different level of information, viz., that of information about the information of the speaker.

29. Groenendijk and Stokhof [1981], cf., also [1984, Ch. 5, App. 1] presents an epistemic/pragmatic treatment of specificity. It would be very interesting to compare their notion of an epistemic model with the *MDPL* notion of an information state, but this must be left for another occasion.

of example 43 is satisfied correctly predicts the likeliness of a specific reading of the example.

That concludes the digression.

Epistemic modals in information exchange

I now turn to the use of the epistemic operators $\Diamond$ and $\Box$ in the exchange model. On the definition of proper information exchange, a use of $\Diamond\phi$ and $\Box\phi$ implies a symmetric test of the speaker's and hearer's state (it is still assumed that ϕ does not contain embedded occurences of epistemic operators):

Fact 4.10 (Epistemic operators in exchange)

- $\langle s\ t\rangle[\![\Diamond\phi]\!] = \langle s\ t\rangle$ if $s \models \Diamond\phi$ and $t \models \Diamond\phi$
- $\langle s\ t\rangle[\![\Box\phi]\!] = \langle s\ t\rangle$ if $s \models \phi$ and $t \models \phi$[30]

In an exchange situation, a proper use of $\Diamond\phi$ ($\Box\phi$) tests whether both speaker and hearer agree on the possibility (truth) of the proposition expressed by ϕ. We might say that the operators $\Diamond$ and $\Box$ test the common ground.[31] In case of disagreement, we can again expect the hearer to object. For instance, if on the present definition of information exchange an utterance of $\Diamond\phi$ is unacceptable to the hearer, this is because he has information inconsistent with ϕ and he can convey this information to the speaker. Similarly, if the hearer does not agree with $\Box\phi$ this is because he does not have the information that ϕ. The speaker then may react to the hearer's objection simply by informing him, yet, that ϕ.

So, if on hearing somebody knocking on the door the speaker says *It is John* ($x = j$), the hearer is expected to update his information accordingly if he has no information to the contrary. On the other hand, if the speaker says *Maybe it is John* ($\Diamond(x = j)$), the hearer is expected to agree, or, if his information excludes that it is John, to disagree and object that it can't be John. And, if in the same situation the speaker says *This is John of course* ($\Box(x = j)$), the hearer is expected to agree if he also thinks that it is John, or, if he is not convinced that it is John, to disagree, and ask, for instance, what makes the speaker think so.

We see that the epistemic modals, at least when they are not embedded, behave fine in information exchange. Let us now consider what happens when we allow embedded uses of the epistemic operators.[32] We saw above that proper information

30. By the definition of proper information exchange, $\langle s\ t\rangle[\![\Diamond\phi]\!] = \langle s[\![\Diamond\phi]\!]\ t[\![\Diamond\phi]\!]\rangle$ if $s \models \Diamond\phi$ and $t \not\models \neg\Diamond\phi$, that is, if $s \models \Diamond\phi$ and $t \models \Diamond\phi$. Furthermore, if $s \models \Diamond\phi$ and $t \models \Diamond\phi$ then $s[\![\Diamond\phi]\!] = s$ and $t[\![\Diamond\phi]\!] = t$, so $\langle s\ t\rangle[\![\Diamond\phi]\!] = \langle s\ t\rangle$ then. Similarly for $\Box\phi$.

31. If $s \models \phi$ and $t \models \phi$, then $(s \vee t) \models \phi$ and if $s \models \Diamond\phi$ and $t \models \Diamond\phi$, then $(s \vee t) \models \Diamond\phi$. On the other hand, although the fact that $s \models \phi$ (or $t \models \phi$) implies that $(s \wedge t) \models \phi$, from $s \models \Diamond\phi$ and $t \models \Diamond\phi$ it does not follow, and, of course, should not follow, that $(s \wedge t) \models \Diamond\phi$.

32. Veltman explicitly excludes embedded use of modals because, he says, they threaten idempotence. The inference $\phi \models \phi$ would no longer be valid and is not easily restored. Notice, however,

exchange does not license jumping to conclusions, that is, that exchange of information does not generate information which speaker and hearer did not already have, together. Now, if we use embedded modals it does. Here are some examples:

(45) Let $s \models \neg p, t \models \Diamond p$ and $(s \wedge t) \models \Diamond\neg q$;
then $s[\![\Diamond p \rightarrow q]\!] \wedge t[\![\Diamond p \rightarrow q]\!] \models q$

(46) Let $s \models \Diamond\neg p, t \models p$ and $(s \wedge t) \models \Diamond\neg q$;
then $s[\![\Box p \rightarrow q]\!] \wedge t[\![\Box p \rightarrow q]\!] \models q$

(47) Let $s \models \Diamond(p \wedge q), t \models \neg(p \wedge q)$ and $(s \wedge t) \models \Diamond p$;
then $s[\![p \rightarrow \Diamond q]\!] \wedge t[\![p \rightarrow \Diamond q]\!] \models \neg p$

(48) Let $s \models (p \rightarrow q), t \models \Diamond(p \wedge \neg q)$ and $(s \wedge t) \models \Diamond p$;
then $s[\![p \rightarrow \Box q]\!] \wedge t[\![p \rightarrow \Box q]\!] \models \neg p$

In the first example, the statement that if it might be that p, then q is true in the speaker's state, simply because she has information that p is false, and, hence, $\Diamond p$ is false in her state. However, the hearer, who still considers it possible that p, concludes from $\Diamond p \rightarrow q$ that q is the case. Hence, the product of information that results from the exchange contains the information that q, which, however, is not information which is entailed by the original product of s and t. Similar observations show that in the other examples, which also satisfy the conditions on proper information exchange, the exchange also generates information that is not licensed by the product of information the participants start with. This is problematic.

Notice that there is every reason to expect that $\Diamond$ and $\Box$ ill-behave in the present framework. Basically, the semantic framework models update of information about the world and about the values of variables induced by the interpretation of sentences. As Veltman observes, the modals do not express information about the world, but about states of information about the world. As we have seen, such operators can be properly used, given motivated side-conditions on information exchange, if they stand on their own, so to speak, that is, if they are not embedded. However, as soon as the operators are embedded, they get related to propositions expressing information about the world, and in that case the information they convey about the world is made dependent upon properties of the state in which interpretation takes place. So, what such a sentence says about the world may differ according to the state it is interpreted in.

Still there is a rigid and adequate way to expel jumping to conclusions within the *MDPL* exchange framework. We argued above that we should not conceive of epistemic statements $\Diamond\phi$ and $\Box\phi$ as statements which are used to exchange information, that is, factual information, but as statements that serve to establish agreement

that, as Veltman also observes, idempotence does not hold in *DPL*, and neither does it hold in *M/EDPL*. So, the loss of idempotence is not a reason to exclude embedded modals from *MDPL*. The point to be made here is that there are other reasons to be at least cautious with embedded modals.

about the possibility or truth of certain propositions in the first place. Now, if we do allow sentences which contain embedded occurrences of these operators, we may conceive of these sentences as well as sentences which serve to establish such kinds of agreement. So, if a sentence ϕ contains an embedded occurrence of an epistemic operator, then, the idea is that on hearing that ϕ the hearer does not update his state with ϕ, but tests whether he agrees with the speaker about the truth of ϕ.

We can elaborate this by distinguishing, like Veltman, two levels of sentence construction. At the first level the language $\mathcal{L}_0$ of predicate logic is generated without epistemic operators and at the second level the language $\mathcal{L}_1$ is generated which is $\mathcal{L}_0$ with epistemic operators. The interpretation of $\mathcal{L}_1$ (and $\mathcal{L}_0$) remains defined as above. The distinction between $\mathcal{L}_0$ and $\mathcal{L}_1$ sentences becomes relevant only in information exchange. In an exchange situation it is required that any sentence $\phi \notin \mathcal{L}_0$ is true in the hearer's state.

Definition 4.9 (Proper information exchange (2))
If ϕ contains epistemic operators, i.e., if $\phi \notin \mathcal{L}_0$, then

- $\langle s\ t\rangle[\![\phi]\!] = \langle s\ t\rangle$ if $s \models \phi$ and $t \models \phi$

So, a sentence ϕ with epistemic operators is now used, solely, to test whether speaker and hearer agree about the truth of ϕ. Clearly, this treatment of such sentences *effectively* excludes jumping to conclusions, since these sentences are disabled to contribute any factual information. So we may ask ourselves now whether it also does so in an *adequate* way. The following discussion aims to show that it does.

First notice that in this refined exchange model, which has separate side conditions for $\mathcal{L}_0$ sentences and $\mathcal{L}_1$ sentences which are not in $\mathcal{L}_0$, all sentences without embedded modals behave as they did in the original model. Furthermore, if ϕ is an $\mathcal{L}_0$ sentence, also the side conditions associated with an utterance of $\Diamond\phi$ or $\Box\phi$ remain to be that $s \models \Diamond\phi$ and $t \models \Diamond\phi$ or that $s \models \phi$ and $t \models \phi$, respectively. Moreover, formulas with embedded modals are predicted to be rejected precisely in cases where one might expect a hearer to object.

Let us consider the four examples which did license jumping to conclusions above. Suppose the speaker says *If might p then q*. The speaker is licensed to this statement iff her state s is such that $s \models \neg p$ or $s \models q$. Since this is an $\mathcal{L}_1$ statement, the hearer tests whether $\Diamond p \rightarrow q$ is *true* in his state t. Now, $\Diamond p \rightarrow q$ is not true in t iff $t \models \Diamond p \wedge \Diamond\neg q$. In other words, the hearer either accepts the speaker's statement, or he may be expected to reply with *No, it might be that p and it might be that not q*. The speaker then may be expected to reply, in her turn, that $\neg p$, if $s \models \neg p$, or that q, if $s \models q$.

The other three examples can be worked out in a similar way. If the speaker says *If it must be that p, then q* then her state s has to be such that either $s \models \Diamond\neg p$ or $s \models q$. If the statement is unacceptable for the hearer, then his state t must be such that $t \models p \wedge \Diamond\neg q$, and he may protest that *p, but maybe not q*. In that case, the speaker learns that p, if $s \models \Diamond\neg p$, or she objects to the hearer's objection by countering that q.

If the speaker is licensed to state *If p then it might be that q* then $s \models \neg p$ or $s \models \Diamond(p \wedge q)$. If the statement is unacceptable for the hearer, then $t \models \Diamond p \wedge (p \rightarrow \neg q)$. So he may be expected to protest that, as far as he knows, it might be that p, but if p, then not q. The speaker may answer that objection by informing the hearer that $\neg p$ or she herself learns that $p \rightarrow \neg q$.

Finally, if the speaker justifiedly utters *If p then it must be that q*, then $s \models \neg p$ or $s \models p \rightarrow q$. The hearer rejects this if $t \models \Diamond(p \wedge \neg q)$. His objection that it might be that p and not q then can be properly responded to by the speaker by stating that $\neg p$ or that $p \rightarrow q$.

We see that the refined exchange model not only expels jumping to conclusions, but, moreover, associates plausible side conditions with sentences containing embedded modals.[33] Of course, the analysis presented here is incomplete. The effects of violating side conditions, prosaically described above, are not formally predicted. The present framework only predicts in which specific cases a hearer may be expected to object, not what his objection will be. But notice that these predictions do seem correct. The above sketched cases in which the side conditions are violated, I think, correspond to cases where one, intuitively, would feel forced to object.

Notice, furthermore, that we have not paid attention to the interaction between epistemic operators and quantifiers, and I will not do so here. Let me only note that the quantifiers and the epistemic operators, as they are defined now, do not seem to interact as intuition might want to have it. For instance, $\forall x \Diamond \phi$ turns out equivalent with $\Diamond \exists x \phi$, which seems quite undesirable indeed. Sensible adjustments can be made, however. But, as yet it is not fully clear to me what information is to be expressed, or what agreement is aimed to be established, when someone utters a quantified statement containing epistemic operators and, therefore, it would be premature to start improving upon the analysis presented here. For this reason, this issue too will be left for future research.

4.4 Partial objects in partial worlds

This final section aims to give more substance to the notion 'the value of a variable', i.e., the kind of things which *MDPL*'s information states contain information about. I will give this notion a precise definition and show that the defined objects have properties which are traditionally associated with partial objects.

So, what then is a partial object? Landman [1986] gives the following characterizations. A partial object is something we assume in conversation and which

---

33. The question may remain, of course, whether the epistemic operators should be allowed to occur embedded in the first place.

we ascribe properties to and which can be followed through information growth, something that can be, or can be not identical to another partial object, something that can grow into a less partial object and that can turn out to be or to be not identical with another partial object and something that can be shared by different information states. In what follows I will show that such a notion of partial object is already implicit in *MDPL*'s notion of information. A partial object in *MDPL* is the value of a variable in a state of partial information.

In the following exposition I will not try to fully explain and motivate all definitions. They merely serve to show that the notion of the value of a variable can be conceived of as that of a partial object.

Partial objects

An information state contains information about the world, labeled by a variable v, and about individuals, labeled by ordinary variables. Such an information state determines what are the possible values of the variables in its domain. Therefore, a *subject* of an information state s we may think of as a variable x in its domain, and we can take an *object* of s to be the value $[x]_s$ of that variable in a state s, a notion which will be defined shortly. The values in a state s of s' subjects may be thought of as the objects of s since s contains information *about* the values of its subjects. Moreover, these values are *partial* objects since states, generally, contain only partial information about the values of subjects.

So, the question now is what the value of a variable is. Normally the question what the value (or denotation) of a variable is is given a conditional answer, since it depends on an assignment. The value of x is $i(x)$ under assignment i. The answer can be made unconditional by phrasing it as a function from assignments to assignment values. Hence, the meaning of a variable x can be equated with the function h from variable assignments to variable values such that for any assignment g: $h(g) = g(x)$. This is relatively standard. In *MDPL* this notion of the meaning of a variable is relativized again to information states. The value of a variable x in a state s, which I will label the *denotation* of x in s, is just the restriction of the meaning of x to the assignments considered possible in s[34]:

Definition 4.10 (The denotation of variables)

If x is a subject of s, i.e., if $x \in D(s)$,

- the denotation of x in s, $[x]_s$, is the function f: $s \to D$ such that for all $i \in s$: $f(i) = i(x)$

34. I am not so happy about the adopted term 'denotation', but I couldn't come up with something better. As appears from the definition of the denotation of x in s, it would be misleading to refer to it as 'the meaning of x' or 'x's intension', since these terms may be reserved for the function that assigns each state the denotation of x in that state. Also, using the term 'value' would be confusing, since that would render this term ambiguous.

The value of x in s may depend on the values of other variables. So, its denotation in s is a function that assigns the value $i(x)$ of x to each assignment i in s. Notice that the denotation of a variable in a state s, as it were, carries the state s of which it is an object with it: s is its function domain.

The objects of a state s are the denotations of the subjects of s:

Definition 4.11 (Partial objects)

- The set of partial objects of a state s, $\mathcal{E}(s)$, is $\{[x]_s \mid x \in D(s)\}$
- The set of partial objects, $\mathcal{E}$, is $\bigcup_{s \in S} \mathcal{E}(s)$

An object $\mathbf{d} \in \mathcal{E}(s)$ will be called a total object iff it is a constant function: $\forall i, j \in s$: $\mathbf{d}(i) = \mathbf{d}(j)$. So, if $[x]_s$ is a total object, then according to s the identity of the value of x is fixed. An object $\mathbf{d} \in \mathcal{E}(s)$ is called impossible iff s is absurd. If a state contains contradictory information about the value of a variable, i.e., if it is an absurd state, then, for any subject x, the denotation of x in s is impossible.

If a state s is updated with an atomic formula $Rx_1 \ldots x_n$, the variables $x_1, \ldots, x_n$ can be taken to refer to their denotations in s:

Fact 4.11

- $s[\![Rx_1 \ldots x_n]\!] = \{i \in s \mid \langle [x_1]_s(i), \ldots [x_n]_s(i) \rangle \in F(R)(i(v))\}$

Of course, given the definition of the denotation of a variable in s this fact is quite trivial indeed. The remainder of this section then is intended to show that the notion of the denotation of a variable itself is not that trivial after all. These denotations can be ascribed identity conditions which are typical of partial objects, they can be taken to grow into less partial objects, that is, they can be followed under information growth, and they can be shared by different information states, that is, agents can be taken to exchange information about shared objects.

Identity

On the present notion of partial objects they have standard identity conditions, but non-standard non-identity conditions:

Fact 4.12

- $[x]_s = [y]_s$ iff $s \models x = y$
- $[x]_s \neq [y]_s \not\Rightarrow s \models x \neq y$

An identity statement $x = y$ is true in a state s iff the denotations of x and y in s are identical. On the other hand, the fact that two partial objects $[x]_s$ and $[y]_s$ are not identical in a state s should not, and does not, imply that $x \neq y$ is true in s, it only says that $x = y$ is not true in s. However, if $[x]_s \neq [y]_s$, what does follow is that $s \models \Diamond(x \neq y)$.

For two objects in a state s to be distinct something more is required, viz., that their values cannot coincide. Let us say that $\mathbf{d}$ and $\mathbf{d}'$ in $\mathcal{E}(s)$ are distinct iff there is no i in s such that $\mathbf{d}(i) = \mathbf{d}'(i)$. Then $[x]_s$ and $[y]_s$ are distinct iff $s \models x \neq y$.

So, if we consider two variables x and y in $D(s)$, they may be assigned the same denotation and this is the case iff $s \models x = y$. However, they may also be assigned different denotations. In that case they are either distinct, if $s \models x \neq y$, or their relative identity is unresolved in s. In the latter case they might be identical but they might as well be non-identical, i.e., in that case $s \models \Diamond(x = y)$ and $s \models \Diamond(x \neq y)$.

Approximation

It should have been noticed that a partial object can not be an object of two different information states, simply because partial objects are tied to their states. They 'live' in their partial world, so to speak. Still, partial objects can be followed under information update. If a state s' is an update of state s, then for any partial object $\mathbf{d}$ in $\mathcal{E}(s)$ there is one, less or equally partial, object $\mathbf{d}'$ in $\mathcal{E}(s')$ such that, as I will call it, $\mathbf{d}$ approximates $\mathbf{d}'$. Approximation is defined as follows:

Definition 4.12 (Approximation)
If $\mathbf{d} \in \mathcal{E}(s)$ and $\mathbf{d}' \in \mathcal{E}(s')$, then

- $\mathbf{d}$ approximates $\mathbf{d}'$, $\mathbf{d} \leq \mathbf{d}'$ iff $D(s) \subseteq D(s')$ and $\forall j \in s'\ \exists i \in s : i \leq j$ and $\mathbf{d}(i) = \mathbf{d}'(j)$

Notice the following fact.

Fact 4.13
If $x \in D(s), y \in D(s')$, then

- $[x]_s \leq [y]_{s'}$ iff $s \leq s'$ and $[x]_{s'} = [y]_{s'}$[35]

Approximation is intimately connected with update and identity. An object $\mathbf{d}'$ is approximated by an object $\mathbf{d}$ if $\mathbf{d}'$ lives in an update of the state which $\mathbf{d}$ lives in and $\mathbf{d}'$ is the denotation of a variable which is identified in s' with a variable which has a denotation $\mathbf{d}$ in s. Hence, clearly, if $s \leq s'$ then $[x]_s \leq [x]_{s'}$, that is, the denotation of a variable in s approximates its denotation in an update of s. Furthermore, if s' is an update of s, then the denotation of x in s approximates the denotation of y in s' if $x = y$ is true in s'.

The following fact shows that a partial object can be followed under information growth:

Fact 4.14

- If $\mathbf{d} \in \mathcal{E}(s)$ and $s \leq s'$, then there is one $\mathbf{d}' \in \mathcal{E}(s')$: $\mathbf{d} \leq \mathbf{d}'$[36]

35. Proof: by the definition of the denotation of variables, update and approximation.
36. Proof: Let $\mathbf{d}$ be the denotation of some variable x in s. Then $\mathbf{d}$ approximates the denotation of x

In the process of information update partial objects do not remain the same all the time, because they get more and more properties ascribed to them. Still, they always have a unique follow-up which they grow into.

Notice that two partial objects in a state s are distinct iff they can not both grow into one and the same possible object: for any $\mathbf{d}$ and $\mathbf{d}'$ in $\mathcal{E}(s)$, $\mathbf{d}$ and $\mathbf{d}'$ are distinct iff for all $\mathbf{d}''$ if $\mathbf{d} \leq \mathbf{d}''$ and $\mathbf{d}' \leq \mathbf{d}''$ then $\mathbf{d}''$ is impossible.

Approximation defines a partial order on the universe of partial objects:

Fact 4.15

- $\langle \mathcal{E}, \leq \rangle$ is a partial order

This corresponds to intuition. Of course a partial object approximates itself (reflexivity of $\leq$). Furthermore, if $\mathbf{d}$ can grow into $\mathbf{d}'$ and $\mathbf{d}'$ can grow into $\mathbf{d}''$, then $\mathbf{d}$ can grow into $\mathbf{d}''$ (transitivity). Finally, if $\mathbf{d}$ and $\mathbf{d}'$ can grow into each other, they are identical ($\leq$ is antisymmetric).

In fact $\langle \mathcal{E}, \leq \rangle$ is a meet semilattice. The meet of two objects is the mere identification of the two.[37] However, there is not in general a join of two partial objects $\mathbf{d}$ and $\mathbf{d}'$ in $\langle \mathcal{E}, \leq \rangle$. For instance, if $\mathbf{d} \in \mathcal{E}(s)$, $\mathbf{d}' \in \mathcal{E}(s')$, $s \in S^{\{x\}}$ and $s' \in S^{\{y\}}$ then there is no $\mathbf{d}''$ such that $\mathbf{d}'' \leq \mathbf{d}$ and $\mathbf{d}'' \leq \mathbf{d}'$. However, if two objects do have a common approximator, they are what I call a shared object.

Shared objects

Using the notion of approximation, we can also define what it means for two different states to contain information about the 'same' object. Two objects may be called a shared object if there is an object that approximates both:

Definition 4.13 (Sharing)

- $\mathbf{d}$ and $\mathbf{d}'$ are a shared object iff $\exists \mathbf{d}''$: $\mathbf{d}'' \leq \mathbf{d}$ and $\mathbf{d}'' \leq \mathbf{d}'$

The strongest approximating object can be found in the common ground:

Fact 4.16

- If $\mathbf{d} \in \mathcal{E}(s)$ and $\mathbf{d}' \in \mathcal{E}(s')$ are a shared object then $\exists \mathbf{d}'' \in \mathcal{E}(s \vee s')$: $\mathbf{d}'' \leq \mathbf{d}$ and $\mathbf{d}'' \leq \mathbf{d}'$[38]

Shared subjects always denote shared objects:

---

in s'. Next assume that $\mathbf{d}$ approximates $\mathbf{d}'$. Then $\forall j \in s'\ \exists i \in s : i \leq j$ and $\mathbf{d}'(j) = \mathbf{d}(i) = i(x) = j(x)$ and, hence, $\mathbf{d}'$ is the denotation of x in s'.

37. The meet $[x]_s \circ [y]_{s'}$ of $[x]_s$ and $[y]_{s'}$ can be defined as $[x]_{(s \wedge s')[\![x=y]\!]} = [y]_{(s \wedge s')[\![x=y]\!]}$.

38. Let $\mathbf{d}$ be the denotation of a variable x in s, and $\mathbf{d}'$ the value of a variable y in s'. If there is an object $\mathbf{d}''$ such that $\mathbf{d}'' \leq \mathbf{d}$ and $\mathbf{d}'' \leq \mathbf{d}'$, then $\mathbf{d}''$ is the value of a variable z in a state s'' such that $s'' \leq s$ and $s'' \leq s'$ and $\mathbf{d} = [z]_s$ and $\mathbf{d}' = [z]_{s'}$. But then $z \in D(s)$ and $z \in D(t)$. Hence, $z \in D(s \vee t)$ and $[z]_{s \vee s'} \leq [z]_s = \mathbf{d}$ and $[z]_{s \vee s'} \leq [z]_{s'} = \mathbf{d}'$.

Fact 4.17

- $\forall x \in D(s) \cap D(s')$: $[x]_s$ and $[x]_{s'}$ are a shared object

Shared objects can be followed under information growth:

Fact 4.18

- If $[x]_{s \vee s'} = [y]_{s \vee s'}$ then $[x]_s$ and $[y]_{s'}$ are a shared object
- If $[x]_s$ and $[y]_{s'}$ are a shared object then $[x]_{s \wedge s'} = [y]_{s \wedge s'}$

So, for any object $\mathbf{d}$ in the common ground of s and s', the follow-ups of $\mathbf{d}$ in s and s' are shared objects. Furthermore, if x in s and y in s' denote a shared object, then they denote one and the same object in the product of s and s'.

Let us finally see in what sense exchange of information about the values of variables is related to partial objects. It must be clear that, normally, the object which a variable x refers to is not literally the same object in the state of the speaker as the object referred to in the state of the hearer. Still, such a variable always refers to a shared object, if it is defined in both states. So, agents exchange information about shared objects, that is, about the follow-ups in their respective states of objects in their common ground.

To conclude, the notion of information in *MDPL* implicitly contains a notion of partial objects which can be introduced in an information state, which can grow into another, less partial, object, and which may turn out to be identical or distinct in the process of information update. These objects live in their state and they are completely dressed with the properties ascribed to them by the information state which they live in. Still, different information states may share objects and exchange information about them.

MDPL's partial objects share some characteristic properties with Landman's pegs [1986, pp. 124 ff]. But there are also differences. The fundamental difference is that the notion of a partial object in *MDPL* is a derived notion, whereas Landman's pegs are primitive objects. In *MDPL* a partial object only 'exists' in its own information state and has no status besides that. In Landman's theory pegs are some kind of objects which are assumed to be there from the start and whose main role is that of getting properties ascribed to them by information states.

In fact, Landman's pegs are more like *MDPL*'s variables, about the values of which *MDPL*'s information states contain information, and, hence, ascribe properties to. For instance, in Landman's theory two *different* pegs can be identical on the basis of a state s. Something similar may hold of two different variables in *MDPL* whose denotations in a state s can be identical. However, in *MDPL* no two different partial objects in a state s can be identical in s (although, of course, it is possible that they grow into one object after information update). Notice that in *MDPL* it is completely decided whether the values of two variables in the domain of s are identical in s. In Landman's theory the question of whether two pegs are identical on the basis of a state s depends on the possible extensions of s in a given information structure.

Furthermore, in Landman's theory two different pegs may be identical on the basis of a total information state, as, similarly, the values of two different variables can be identical in a total, or maximal, information state in *MDPL*. For this reason, in Landman's theory the 'real objects' in such a total state s are equated with equivalence classes of pegs which are indiscernable in s. In *MDPL*, on the other hand, the set of partial objects of a total information state constitutes a classical domain of objects by itself. The partial objects of a total information state in *MDPL* are total objects, and two non-identical objects of such a state s are distinct in s.

In Landman's theory, information states contain 'facts about pegs'. These pegs can be conceived of as some kind of constants to which properties are attributed by different information states and which can be followed through information growth. In *MDPL*, one might say, the constants in information update and exchange are variables, whereas the information that gets exchanged is information about the values of these variables. It may be worthwhile to note, finally, that Landman only offers a static semantics. Landman restricts himself to defining the eventual truth or falsity of statements in states which an agent may arrive at after information growth. *MDPL*, on the other hand, offers a dynamic update semantics, and defines the growth of information itself that results from accepting such statements.

The role of variables

The reader will have noticed that, when compared to ordinary predicate logic, the role of variables has increased to a large extent. Variables not only serve to indicate which arguments are bound by which quantifiers, in *MDPL* they are also the one and only means to refer to the partial objects information is exchanged about. Still, they remain artefacts, of course. So, the desire to *eliminate* the variables may have increased accordingly.

It is compelling to speculate about ways to eliminate variables. Sometimes I have the feeling that there is a small switch which has to be turned in order to collapse the *MDPL* domain of partial objects into something much more simple and, by means of some adequate postulates, preserve all the results. Maybe one might start from a primitive lattice of information states appropriately related to a primitive meet semilattice of partial objects, together with a device that enables agents in an exchange situation to determine which shared objects are spoken about. It seems compelling to try out such an approach, because, besides enabling a simplification, in the end it might enable us to free ourselves from the definedness conditions, which, as argued, solely derive from the need to expel unfortunate indexing.

However, I will not speculate any further about such a possible development here. Before TEX's capacity exceeds, I only want to note that every chapter must come to an end and that, if a chapter has an end, only once it is here.

Bibliography

Asher, N. & Wada, H., 1988, 'A computational account of syntactic, semantic and discourse principles for anaphora resolution', *Journal of Semantics* 6, pp. 309–344

Bartsch, R., 1972, *Adverbialsemantik*, Frankfurt am Main: Athenäum Verlag [translation in English: Bartsch, R., 1976, *The grammar of adverbials*, Amsterdam: North-Holland]

Bartsch, R., 1987, 'Frame representations and discourse representations', *Theoretical Linguistics* 14, pp. 65–117

Barwise, J., 1987, 'Noun phrases, generalized quantifiers and anaphora', in: Gärdenfors, P. (ed.), *Generalized quantifiers*, Dordrecht: Reidel, pp. 1–29

Barwise, J. & Perry, J., 1983, *Situations and attitudes*, Cambridge: MIT Press

Bäuerle, R., 1979, *Temporale Deixis, temporale Frage*, Tübingen: Gunter Narr Verlag

Bäuerle, R. & Egli, U., 1985, *Anapher, Nominalphrase und Eselsätze*, Sonderforschungsbericht 99, 105, Konstanz: Universität Konstanz

Beaver, D., 1992, 'The kinematics of presupposition', in: Dekker, P. & Stokhof, M. (eds.), *Proceedings of the eighth Amsterdam colloquium, December 17–20, 1991*, Amsterdam: ILLC, pp. 17–36

Benthem, J. van, 1984, 'The logic of semantics', in: Landman, F. & Veltman, F. (eds.), *Varieties of formal semantics*, Dordrecht: Foris, pp. 55–80

Benthem, J. van, 1986, *Essays in logical semantics*, Dordrecht: Reidel

Benthem, J. van, 1991, *Language in action*, Amsterdam: North-Holland

Berg, M. van den, 1990, 'A dynamic predicate logic for plurals', in: Stokhof, M. & Torenvliet, L. (eds.), *Proceedings of the seventh Amsterdam colloquium, December 19–22, 1989*, Amsterdam: ITLI, pp. 29–52

Berg, M. van den, 1991, 'Dynamic generalized quantifiers', in: Does, J. van der & Eijck, J. van (eds.), *Generalized quantifier theory and applications*, Amsterdam: Dutch Network for Language, Logic and Information, pp. 223–244

Berman, S., 1987, 'Situation-based semantics for adverbs of quantification', in: Blevins, J. & Vainikka, A. (eds.), *University of Massachusetts Occasional Papers* 12, University of Massachusetts, Amherst, pp. 45–68

Bruyn, J. de & Scha R., 1988, 'The interpretation of relational nouns', *Proceedings of the 26-th annual meeting of the Association for Computational Linguistics*, State University of New York at Buffalo

Chierchia, G., 1988, 'Dynamic generalized quantifiers and donkey anaphora', in: Krifka, M. (ed.), *Genericity in natural language*, Tübingen: SNS, pp. 53–84

Chierchia, G., 1992, 'Anaphora and dynamic binding', *Linguistics and Philosophy* 15, pp. 111–183

Cooper, R., 1979, 'The interpretation of pronouns', in: Heny, F. & Schnelle, H. (eds.), *Syntax and semantics 10. Selections from the Third Groningen Round Table*, New York: AP, pp. 61–92

Davidson, D., 1967, 'The logical form of action sentences', in: Rescher, N. (ed.), *The logic of decision and action*, University of Pittsburgh Press, pp. 81–95

Deemter, K. van, 1991, *On the composition of meaning*, dissertation, University of Amsterdam

Dekker, P., 1990, 'An update semantics for dynamic predicate logic', in: Dekker, P. & Stokhof, M. (eds.), *Proceedings of the eighth Amsterdam colloquium, December 17–20, 1991*, Amsterdam: ILLC, pp. 113–132

Dowty, D., 1979, *Word meaning and Montague grammar*, Dordrecht: Reidel

Dowty, D., 1982, 'Tenses, time adverbs, and compositional semantic theory', *Linguistics and Philosophy* 5, pp. 23–57

Dowty, D., 1986, 'The effects of aspectual class on the temporal structure of discourse: semantics or pragmatics?', *Linguistics and Philosophy* 9, pp. 37–61

Dowty, D., 1989, 'On the semantic content of the notion "thematic role" ', in: Chierchia, G., Partee, B. & Turner, R. (eds.), *Properties, types and meaning VOL II*, Dordrecht: Kluwer, pp. 69–129

Dowty, D., Wall, R. & Peters, S., 1981, *Introduction to Montague semantics*, Dordrecht: Reidel

Eijck, J. van & de Vries, F.-J., 1992, 'Dynamic interpretation and Hoare deduction', *Journal of Logic, Language and Information* 1, pp. 1–44

Egli, U., 1979, 'The Stoic concept of anaphora', in: Bäuerle, R., Egli, U. & Stechow, A. von (eds.), *Semantics from different points of view*, Berlin, Springer Verlag, pp. 266–285

Evans, G., 1977, 'Pronouns, quantifiers and relative clauses (1)', *The Canadian Journal of Philosophy* 7, pp. 467–536 [reprinted in: Evans, G., 1985, *Collected papers*, Dordrecht: Foris, pp. 76–152]

Evans, G., 1980, 'Pronouns', *Linguistic Inquiry* 11, pp. 337–362 [reprinted in: Evans, G., 1985, *Collected papers*, Dordrecht: Foris, pp. 214–248]

Frege, G., 1892, 'Über Sinn und Bedeutung', *Zeitschrift für Philosophie und philosophische Kritik* 100, pp. 25–50

Gamut, L.T.F., 1991, *Logic, language and meaning,* Vol. 2: *Intensional logic and logical grammar*, Chicago: The University of Chicago Press

Geach, P.T., 1972, 'A program for syntax', in: Davidson, D. &, Harman, G. (eds.), *Semantics of natural language*, Dordrecht: Reidel, pp. 483–497

Groenendijk, J. & Stokhof, M., 1981, 'A pragmatic analysis of specifity', in: Heny, F. (ed.), *Ambiguities in intensional contexts*, Dordrecht: Reidel, pp. 153–190

Groenendijk, J. & Stokhof, M., 1984, 'Studies on the semantics of questions and the pragmatics of answers', dissertation, University of Amsterdam

Groenendijk, J. & Stokhof, M., 1988, 'Context and information in dynamic semantics', in: Elsendoorn, B. & Bouma, H. (eds.), *Working models of human perception*, New York: Academic Press, pp. 457–488

Groenendijk, J. & Stokhof, M., 1990a, 'Dynamic Montague grammar', in: Kálmán, L. & Pólos, L. (eds.), *Papers from the second symposium on logic and language*, Budapest: Akadémiai Kiadó, pp. 3–48 [also in: Groenendijk, J., Stokhof, M. & Beaver, D. (eds.), 1991, *Quantification and anaphora I*, DYANA report R2.2.A, Edinburgh: Centre for Cognitive Science, University of Edinburgh, pp. 79–134]

Groenendijk, J. & Stokhof, M., 1990b, 'Two theories of dynamic semantics', in: Eijck, J. van (ed.), *Logics in AI, European Workshop JELIA '90*, Berlin: Springer Verlag, pp. 55–64

Groenendijk, J. & Stokhof, M., 1991, 'Dynamic predicate logic', *Linguistics and Philosophy* 14, pp. 39–100

Heim, I., 1982, *The semantics of definite and indefinite noun phrases*, dissertation, University of Massachusetts, Amherst

Heim, I., 1983, 'File change semantics and the familiarity theory of definiteness', in: Bäuerle, R., Schwarze, Ch. & Stechow, A. von (eds.), *Meaning, use and interpretation of language*, Berlin: De Gruyter, pp. 164–189

Heim, I., 1990, 'E-type pronouns and donkey anaphora', *Linguistics and Philosophy* 13, pp. 137–178

Heim, I., 1992, 'Presupposition projection and the semantics of attitude verbs', *Journal of Semantics* 9, pp. 183–221

Hendriks, H., 1988, 'Type change in semantics: the scope of quantification and coordination', in: Klein, E. & Benthem, J. van (eds.), *Categories, polymorphism and unification*, Edinburgh/Amsterdam: CCS/ITLI, pp. 95–119

Hendriks, H., 1992, *Studied flexibility*, dissertation, University of Amsterdam

Hinrichs, E., 1986, 'Temporal anaphora in discourses in English', *Linguistics and Philosophy* 9, pp. 63–82

Janssen, T., 1986, *Foundations and applications of Montague grammar*, Amsterdam: CWI

Janssen, T., 1990, 'Models for discourse markers', in: Stokhof, M. & Torenvliet, L. (eds.), *Proceedings of the seventh Amsterdam colloquium, December 19–22, 1989*, Amsterdam: ITLI, pp. 213–226

Kadmon, N., 1987, *On unique and non-unique reference and asymmetric quantification*, dissertation, University of Massachusetts, Amherst

Kadmon, N., 1990, 'Uniqueness', *Linguistics and Philosophy* 13, pp. 273–324

Kamp, H., 1981, 'A theory of truth and semantic representation', in: Groenendijk, J., Janssen, T. & Stokhof, M. (eds.), *Formal methods in the study of language*, Amsterdam: Mathematical Centre, pp. 277–322 [reprinted in: Groenendijk, J., Janssen, T. & Stokhof, M. (eds.), 1984, *Truth, interpretation and information*, Dordrecht: Foris, pp. 2–41]

Kamp, H. & Rohrer, C., 1983, 'Tense in texts', in: Bäuerle, R., Schwarze, C. & Stechow, A. von (eds.), *Meaning, use and interpretation of language*, Berlin: de Gruyter, pp. 250–269

Kamp, H. & Reyle, U., 1993, *From discourse to logic*, Dordrecht, Kluwer

Karttunen, L., 1968a, *What do referential indices refer to?*, paper P-3854, Santa Monica, California: The RAND Corporation

Karttunen, L., 1968b, *What makes definite noun phrases definite?*, paper P-3871, Santa Monica, California: The RAND Corporation

Kratzer, A., 1988, 'Stage-level and individual-level predicates', manuscript, University of Massachusetts, Amherst

Kratzer, A., 1989, 'An investigation of the lumps of thought', *Linguistics and Philosophy* 12, pp. 607–653

Lambek, J., 1958, 'The mathematics of sentence structure', *American Mathematical Monthly* 65, pp. 154–169 [reprinted in: Buszkowsky, W., Marciszewski, W. & Benthem, J. van (eds.), 1989, *Categorial grammar*, Amsterdam: Benjamins, pp. 153–172]

Landman, F., 1986, *Towards a theory of information. The status of partial objects in semantics*, Dordrecht: Foris

Larson, R., 1988, 'Implicit arguments in situation semantics', *Linguistics and Philosophy* 11, pp. 169–202

Lewis, D., 1975, 'Adverbs of quantification', in: Keenan, E. (ed.), *Formal semantics of natural language*, Cambridge: UP, pp. 3–15

McConnell-Ginet, S., 1982, 'Adverbs and logical form: a linguistically realistic theory', *Language* 58, pp. 144–184

Montague, R., 1970a, 'English as a formal language', in: Visentini B. et al. (eds.), *Linguaggi nella societa e nella tecnica*, Milan: Edizioni di Comunita, pp. 189–224 [reprinted in: Thomason, R. (ed.), 1974, *Formal Philosophy, selected papers of Richard Montague*, New Haven: Yale University Press, pp. 188–221]

Montague, R., 1970b, 'Universal grammar', *Theoria* 36, pp. 373–398 [reprinted in: Thomason, R. (ed.), 1974, *Formal Philosophy, selected papers of Richard Montague*, New Haven: Yale University Press, pp. 222–246]

Montague, R., 1973, 'The proper treatment of quantification in ordinary English', in: Hintikka, J., Moravcsik, J. & Suppes, P. (eds.), *Approaches to natural language: proceedings of the 1970 Stanford workshop on grammar and semantics*, Dordrecht: Reidel, pp. 221–242 [reprinted in: Thomason, R. (ed.), 1974, *Formal Philosophy, selected papers of Richard Montague*, New Haven: Yale University Press, pp. 247–270]

Muskens, R., 1989, *Meaning and partiality*, dissertation, University of Amsterdam

Muskens, R., 1990, 'Anaphora and the logic of change', in: Eijck, J. van (ed.), *Logics in AI*, Berlin: Springer-Verlag, pp 412–427

Parsons, T., 1985, 'Underlying events in the logical analysis of English', in: Lepore, E. & McLaughlin, B. (eds.), *Actions and events: perspectives on the philosophy of Donald Davidson*, Oxford: Basil Blackwell, pp. 235–267

Parsons, T., 1989, 'The progressive in English: events, states and processes', *Linguistics and Philosphy* 12, pp. 213–241

Parsons, T., 1990, *Events in the semantics of English*, Cambridge, Massachusetts: MIT press

Partee, B., 1984, 'Nominal and temporal anaphora', *Linguistics and Philosphy* 7, pp. 243–286

Partee, B., 1991, 'Topic focus and quantification', in: Moore, S. & Wyner, A. (eds.), *Proceedings from SALT I* Cornell Working Papers in Linguistics 10

Partee, B. & Rooth, M., 1983, 'Generalized conjunction and type ambiguity', in: Bäuerle, R., Schwarze, Ch. & Stechow, A. von (eds.), *Meaning, use and interpretation of language*, Berlin: De Gruyter, pp. 361–383

Pelletier, F.J. & Schubert, L., 1988, 'Generically speaking', in: Chierchia, G., Partee, B. & Turner, R. (eds.), *Properties, types and meaning II*, Dordrecht: Reidel, pp. 193–268

Peters, S., 1977, 'A truth-conditional formulation of Karttunen's account of presupposition', *Texas Linguistic Forum* 6, pp. 137–149 [revised version in: *Synthese* 40, pp. 301–316]

Prior, A., 1967, *Past, present and future*, Oxford: Clarendon

Reichenbach, H., 1947, *Elements of symbolic logic*, New York/London: Macmillan

Roberts, C., 1987, *Modal subordination, anaphora and distributivity*, dissertation, University of Massachusetts, Amherst

Roberts, C., 1989, 'Modal subordination and pronominal anaphora in discourse', *Linguistics and Philosophy* 12, pp. 683–722

Roberts, C., 1992, 'Domain restriction in dynamic semantics', in: Goldberg, J., Kálmán, L. & Szabó, Z. (eds.), *Papers from the third symposium on logic and language, Revfülöp, 1990* Dordrecht: Reidel

Root, R., 1986, *The semantics of anaphora in discourse*, dissertation, University of Texas, Austin

Rooth, M., 1987, 'Noun phrase interpretation in Montague grammar, file change semantics, and situation semantics', in: Gärdenfors, P. (ed.), *Generalized quantifiers*, Dordrecht: Reidel, pp. 237–269

Rooth, M. & Partee, B., 1982, 'Conjunction, type ambiguity and wide scope "or" ', in: Flickinger, D., Macken M. & Wiegand N. (eds.), *Proceedings of the 1982*

West Coast Conference on Formal Linguistics, Stanford: Stanford University

Sells, P., 1985, *Restrictive and non-restrictive modification*, report CSLI-85-28, Stanford, Center for the Study of Language and Information

Seuren, P.A.M., 1985, *Discourse semantics*, Oxford: Blackwell

Stalnaker, R., 1974, 'Pragmatic presuppositions', in: Munitz, M. & Unger, P. (eds.), *Semantics and philosophy*, New York: University Press, pp. 197–213

Strawson, P., 1950, 'On referring', *Mind* 59, pp. 320–344

Stump, G., 1981, *Formal semantics and pragmatics of free adjuncts and absolutes*, dissertation, Ohio State University

Swart, H. de, 1991, *Adverbs of quantification: a generalized quantifier approach*, dissertation, Rijksuniversiteit Groningen

Veltman, F., 1981, 'Data semantics', in: Groenendijk, J., Janssen, T. & Stokhof, M. (eds.), *Formal methods in the study of language*, Amsterdam: Mathematical Centre [reprinted in: Groenendijk, J., Janssen, T. & Stokhof, M. (eds.), 1984, *Truth, interpretation and information*, Dordrecht: Foris, pp. 43–63]

Veltman, F., 1985, *Logics for conditionals*, dissertation, University of Amsterdam

Veltman, F., 1990, 'Defaults in update semantics', in: Kamp, H. (ed.), *Conditionals, defaults and belief revision*, DYANA report R2.5.A, Edinburgh: Centre for Cognitive Science, University of Edinburgh, pp. 28–63

Zeevat, H., 1989, 'A compositional approach to discourse representation theory', *Linguistics and Philosophy* 12, pp. 95–131

Zeevat, H., 1991, *Aspects of discourse semantics and unification grammar*, dissertation, University of Amsterdam

Samenvatting

Dit proefschrift is gewijd aan enige onderwerpen in de dynamische semantiek. In tegenstelling tot in de klassieke, statische, semantiek, wordt in de dynamische semantiek de betekenis van zinnen niet primair gesteld in termen van wat zinnen in isolatie betekenen, maar in termen van een vermogen om zekere toestanden te veranderen. Daarbij wordt uitgegaan van de observatie dat de betekenis van een indicatieve zin doorgaans meer behelst dan alleen de assertie dat een zekere stand van zaken het geval is en dat zulke andere aspecten van de interpretatie van zinnen in de semantiek verantwoord dienen te worden. Men denke hierbij aan semantische relaties die er tussen kunnen bestaan, zoals, bijvoorbeeld, anaforische relaties tussen uitdrukkingen in verschillende zinnen.

Het onderzoek waarvan dit proefschrift verslag doet bouwt voor een goed deel voort op Groenendijk en Stokhof's dynamische Montague grammatica en het eerste hoofdstuk bevat daarom een uitgebreide presentatie van dit systeem, die dient als een inleiding op de volgende drie hoofdstukken.

De dynamische Montague grammatica (*DMG*) geeft, onder meer, een verantwoording van anaforische relaties tussen onbepaalde termen (zoals *iemand* en *een man die in het park loopt*) en anaforische voornaamwoorden (zoals *hij*, *zij* en *het*). Zoals te doen gebruikelijk is, worden semantische verschijnselen hierbij verantwoord in een logisch systeem en is zo'n verantwoording met terugwerkende kracht van toepassing op (een fragment van) de natuurlijke taal, de uitdrukkingen waarvan op een min of meer standaard wijze vertaald worden in uitdrukkingen van het logische systeem.

Het in *DMG* gehanteerde systeem bevat drie primitieve zinsvormende operaties, die van existentiële kwantificatie (gebruikt in de vertaling van onbepaalde termen), negatie en conjunctie. Andere operaties, zoals universele kwantificatie, conditionalisering en disjunctie worden op een standaard wijze in termen van de eerstgenoemde drie gedefinieerd. Kenmerkend voor *DMG* nu is dat het alleen anaforische relaties verantwoordt tussen onbepaalde termen en anaforische voornaamwoorden als niet tegelijkertijd de onbepaalde term in het bereik staat van een negatie en het voornaamwoord daarbuiten.

Nu is het waar dat, naast namen, onbepaalde termen die niet in het bereik van een negatie staan de enige termen lijken te zijn die zich ervoor lenen om als antecedent in een anaforische relatie te staan. Dit is echter lang niet altijd het geval. Reeds in Groenendijk en Stokhof's artikel worden andere voorbeelden gegeven en deze lijken verantwoord te kunnen worden met een notie van 'dynamische negatie'.

In hoofdstuk twee van dit proefschrift worden deze en soortgelijke voorbeelden nader onderzocht.

Het tweede hoofdstuk opent met een discussie over de voorbeelden de analyse waarvan een notie van dynamische negatie lijkt te behoeven. Vervolgens wordt beargumenteerd dat de notie van dynamische negatie die is voorgesteld door Groenendijk en Stokhof wel geschikt lijkt voor een goed deel van deze voorbeelden, maar dat deze toch ook structureel te kort schiet. Voorts wordt aangetoond dat een notie van dynamische negatie die aan drie tamelijk plausibele voorwaarden voldoet binnen het kader van Groenendijk en Stokhof's framework niet gevonden kan worden.

In de tweede helft van dit hoofdstuk wordt vervolgens een alternatieve dynamische Montague grammatica ontwikkeld, en wordt bewezen dat deze in eerste opzet equivalent is met de dynamische Montague grammatica van Groenendijk en Stokhof. Echter, de alternatieve grammatica biedt juist de 'logische ruimte' die nodig is om een adequate notie van dynamische negatie te formuleren. De werking van de alternieve notie van dynamische negatie wordt tenslotte geïllusteerd aan de hand van de voorbeelden die in dit hoofdstuk aan de orde zijn gekomen.

De ontwikkeling van Groenendijk en Stokhof's dynamische Montague grammatica, en vervolgens van het alternatief daarvoor in hoofdstuk twee, kan gekenschetst als een proces dat een vorm van 'type-ophoging' behelst. In klassieke theoriën worden zinsbetekenissen geassociëerd met proposities, in *DMG* zijn dat eigenschappen van proposities, en in het in hoofdstuk twee ontwikkelde systeem zijn dat eigenschappen van eigenschappen van proposities. Een interessante observatie is nu dat de stap van zekere soorten van objecten (proposities) naar eigenschappen van eigenschappen van die objecten (of, gebruikelijker, verzamelingen van verzamelingen van die objecten) een ophoging is die veelvuldig wordt gebruikt in zekere flexibele syntactische en semantische systemen. In hoofdstuk drie wordt onder meer om die reden onderzocht of, en in hoeverre, met behulp van principes uit zulke flexibele calculi dezelfde resultaten bereikt kunnen worden als in de alternatieve Montague grammatica uit hoofdstuk twee.

In dit hoofdstuk wordt een flexibele Montague grammatica voorgesteld, waarin, uitgaande van een heel simpel, statisch interpretatiemodel, dynamische fenomenen verantwoorden worden middels algemene en welgedefiniëerde type-veranderende regels. Daarbij wordt in beginsel gebruik gemaakt van Hendriks' stelsel van typeverandering. Met een relatief simpele wijziging blijkt dit stelsel in staat een verantwoording te geven van intersentiële anaforische verbanden, en van andere vormen van tekst-structurering, voor zover die zogeheten monotoon stijgende uitdrukkingen behelzen (grof gesteld, uitdrukkingen die niet direct of indirect een vorm van (dynamische) negatie veronderstellen).

Om ook de dynamiek van monotoon dalende uitdrukkingen te verantwoorden, wordt de notie van een gegeneraliseerde duale geïntroduceerd. Deze inductief gedefiniëerde operatie omvat zowel de klassieke, statische negatie, als de dynamische negatie uit hoofdstuk twee, als noties van een duale zoals die gebruikt worden in de propositie logica en in de theorie van gegeneraliseerde kwantoren. De duale wordt vervolgens gebruikt in de interpretatie van type-veranderingen van monotoon dalende uitdrukkingen, en daarmee blijken de overige resultaten van hoofdstuk twee afleidbaar te zijn geworden.

Zoals dit hoofdstuk voorts aantoont, is het flexibele systeem, met de duale, een bijzonder krachtig instrument. Het systeem kan met succes toegepast worden in de analyse van nog veel complexere voorbeelden. Echter, zoals wel vaker het geval is met flexibele stelsels, het systeem leidt ook tot ontoelaatbare overgeneratie. Het hoofdstuk besluit met de conclusie dat het krachtige systeem van type-veranderende regels wellicht beter tot zijn recht zou komen als het met de nodige restricties zou worden toegepast op een dynamisch systeem.

Hoofdstuk vier grijpt weer terug op de originele dynamische Montague grammatica. In dit hoofdstuk wordt aangetoond dat de door Groenendijk en Stokhof ontwikkelde techniek voor het correct analyseren van intersententiële anaforische relaties met succes kan worden toegepast bij de verantwoording van op het eerste gezicht tamelijk andersoortige verschijnselen. Enige bestaande analyses van relationele naamwoorden (zoals *zuster*), van bijwoordelijke bepalingen en van temporele relaties in teksten blijken op een eenvoudige en uniforme manier geherformuleerd te kunnen worden in een dynamische Montague grammatica.

Het idee achter dit hoofdstuk is simpel. Zoals in de dynamische Montague grammatica als het ware onthouden wordt welke 'objecten' door een zin geintroduceerd worden, objecten waarnaar door voornaamwoorden terugverwezen kan worden, evenzo kunnen bepaalde lexicale uitdrukkingen geacht worden bepaalde impliciete argumenten aan te dragen, die door andere uitdrukkingen nader gespecificeerd kunnen worden. Bijvoorbeeld, een naamwoord als *zuster* kan opgevat worden als een (dynamisch) predicaat dat van toepassing is op een individu x als x een zuster is van een individu y, waarbij 'onthouden' wordt van welke individuen y zo'n x een zuster is. Zo'n analyse stelt ons in staat het naamwoord op uniforme wijze zonder verdere specificaties te gebruiken, als in *Elke zuster gaat aan het werk*, en met nadere specificaties, als in *De zuster van Jan rijdt naar huis*.

Op vergelijkbare wijze kan de semantische bijdrage van zekere bijwoordelijke bepalingen opgevat worden als een nadere specificatie van door werkwoorden aangedragen impliciete argumenten (bijvoorbeeld Davisoniaanse 'events') en kunnen temporele relaties in eenvoudige narratieve teksten verantwoord worden als relaties tussen de mogelijk events waarnaar de werkwoorden in een tekst kunnen verwijzen.

In het vijfde en laatste hoofdstuk wordt Groenendijk en Stokhof's dynamische predicaat logica bezien vanuit een 'update' perspectief. In dit hoofdstuk worden de in de dynamische predicaat logica verantwoorde anaforische relaties bestudeerd vanuit het perspectief van groei van informatie over partiële objecten. Daartoe wordt eerst een herformulering gegeven van de dynamische predicaat logica die gebruik maakt van partiële, in plaats van totale, interpretatiefuncties. Een karakteristiek verschil met dynamische predicaat logica is dat het resulterende systeem, op straffe van ongedefiniëerdheid, niet toestaat dat objecten die door indefiniete termen geïntroduceerd zijn weer verdwijnen door ongelukkige indicering.

Het systeem wordt vervolgens uitgebreid met kwantificerende uitdrukkingen, adnominale zowel als adverbiale (symmetrisch en asymmetrisch), die op eenvoudige wijze behandeld blijken te kunnen worden. In het laatste deel wordt de gehanteerde notie van ‘informatie over de waarden van variabelen’ aan een nadere studie onderworpen. De gebruikte structuur van informatie (-toestanden) wordt onderzocht op zijn formele eigenschappen, er volgt een uitweiding over informatie-uitwisseling, en tenslotte toon ik enige overeenkomsten aan tussen de gehanteerde notie van ‘waarde van een variabele’ en dat van een partiëel object. Informatie-toestanden blijken te kunnen worden opgevat als door partiële objecten bewoonde partiële werelden.