Logical Considerations in the Interpretation of Presuppositional Sentences

Pablo Gervás Gómez–Navarro

Department of Computing
Imperial College of Science, Technology and Medicine
University of London

A thesis submitted for the degree of
Doctor of Philosophy of the University of London
and for the Diploma of the Imperial College

September 1995

¿Y al fin de tal jornada
presumen espantarme?
Sepan que ya no puedo
morir sino sin miedo; (...)

Garcilaso de la Vega

And at the end of such journey
they expect to scare me?
Let them know that by now
if I die it will be without fear; (...)

A mis padres.

To my parents.

Acknowledgements

I would like to thank Krisia Broda, Cris Hankin, and Diego Eguidazu for reading early drafts of this thesis; Tony Hunter, Atocha Aliseda, Marcello D'Agostino, David Beaver, Andreas Schöter, Nissim Francez, Daniel Marcu, Rob van der Sandt, Alessandra Russo, and Senanu Etrey for comments and suggestions; all members of the Logic and Automated Reasoning Group for interesting seminars and discussions throughout the years; and my supervisor Dov Gabbay for his support.

I would also like to thank Silvia Onesti, Norman Hui, Cris Hankin, Don Addlington, Paul Preston and Anthony Finkelstein for upholding my faith in academic endeavour at times when every prospect seemed dark; Iñigo Egusquiza, Sara Román, Estela González, Mónica Nombela, Jane Evans, Hilfred Chau, Cecilia García, the Anderiz family, Zara Skipwith, Diego Eguidazu, Ola Al-Deeb, Silvia Onesti, Norman Hui, Irene Azcárraga, and the Hopkins family, for their friendship and support throughout the years; and all members past and present of the Arturo Duperier Spanish Society of Imperial College, for countless hours of merriment.

Very special thanks to Beatriz Gómez-Argüello.

And the most special thanks of all to my parents, Maria José and Agustín, and my brother Carlos, for their unwavering faith in my ability.

Most of the work in this thesis was carried out while I was supported by grant PG33503277 from the Spanish Ministerio de Educación y Ciencia (Becas F.P.I.).

Abstract

Tableaux originate as a decision method for classical logic. They can also be extended to obtain a structure that spells out all the information in a set of sentences in terms of truth value assignments to atomic formulas that appear in them. The aim of the present work is to study whether such a tableau representation for a linguistic context can provide a unifying framework for the treatment of presupposition in which presupposition behaviour can be described simply and justified in terms of the logical properties of tableaux.

The behaviour to be explained includes: how presupposition contributes to a given context in terms of information; how this contribution can sometimes be defeasible or redundant under specific circumstances; how this presuppositional contribution is worked out for compounds built from presuppositional sentences and natural language particles *if ... then*, *and* and *or*; and why certain constructions that involve presuppositions sound counterintuitive.

The traditional view on compositionality of presupposition is reformulated within a framework of classical logic tableaux for propositional logic. Descriptive rules are given that capture the behaviour of presupposition with respect to the structure of an enhanced version of tableaux. A new approach to the projection problem results, in which the presuppositions can be said to behave compositionally with respect to the semantics if not to the syntax of the sentences. This new approach is then expressed as a set of additional tableau expansion rules for presupposition. The resulting tableaux define a consequence relation that includes presupposition as well as traditional entailment. In terms of this consequence relation an integrated solution to the problems of contribution of presupposition to a context, defeasibility of presupposition and acceptability of sentences involving presuppositions is proposed. The extension of this approach to first order logic exposes interactions between existence predicates, presuppositions of existence, the extension of domains, and quantification.

Contents

Chapter 1

Introduction

1.1 An Overview of the Thesis

The present work is an attempt to study how presupposition (as a pragmatic operation with defined behaviour patterns) interacts with the classical logic semantics that have been used in many frameworks to represent natural language knowledge.

Presupposition as we know it exists only in natural language, and not in logic. The present work is concerned with studying how the consideration of a phenomenon equivalent to presupposition would affect the properties of traditional classical logic. There have been attempts before to build a logic with presupposition (for instance, van Fraasen [48, 49]). The present attempt is motivated by the recent developments that have taken place in the description of presupposition behaviour. The work of Karttunen[19, 20], Stalnaker [42, 43], Heim [12, 13, 14] and Beaver [1, 2] has clarified the behaviour of presupposition with respect to conditional and conjunctive constructions. The work of Gazdar [8, 9] and Mercer [27, 28, 30, 31, 32] has provided insights on the treatment of presuppositions that arise from negative constructions when they are faced with inconsistency. The work of van der Sandt [47] has shown how presupposition has an anaphoric ingredient. All these new insights were unavailable to the earlier attempts. The present work constructs a logic for presupposition that covers all these different aspects. In addition, the logic covers the issue of acceptability of sentences [1] and the problem of

[1] The work of Marcu [26] on infelicitously and felicitously defeasible presuppositions

quantification in contexts where objects can be said to exist or not to exist.

The study is carried out taking as reference a fragment of natural language built from declarative sentences (some of which may have presuppositions) and the connectives *if ... then*, *and*, *or*, and negation (in its different syntactical forms as determined by the grammar).

The study of presupposition can be described, as a first approximation, in terms of propositional logic. The declarative sentences are represented as atomic propositions. The connectives are represented as classical $\rightarrow$, $\wedge$, $\vee$ and $\neg$. In this case, presupposition involves a relationship between different sentences of a language. One sentence A *presupposes* another sentence B. A sentence may presuppose several sentences. A sentence that is presupposed may itself presuppose a further sentence. In this first approximation, representation of presupposition can be restricted to defining the ordered pairs of sentences for which this relationship holds. In terms of the description above this corresponds to defining (B, A) as belonging to the set of presuppositionally related sentences of the language. B is a *presupposition.* A is a *presuppositional sentence* or *presuppositional trigger*. To simplify the notation, this presupposition relation is represented as A^B.

Little (almost nothing) is said in this thesis concerning the translation from natural language sentences to logical propositions. This is not meant to imply that the issue is trivial. There are multiple problems involved in this translation (anaphora, scope, ellipsis, ambiguity ...). It is not immediately apparent that the problems treated here are independent from this process of translation. Other frameworks address the issues of presupposition including the process of translation from sentences to propositions (see Beaver's ABLE or van der Sandt's DRS treatment). They all resort to a logical ingredient to represent the structure of the information that is obtained as a result of the translation. The present work is concerned with how such a logical ingredient should behave. Although the process of obtaining the logical representation from the linguistic representation is a crucial part of the process of interpretation of sentences, it is not addressed here.

The present thesis is based on a working hypothesis that classical logic can be used to represent this underlying logical structure. In that sense, it differs from other logical and computational studies of presupposition in-

came to my notice when the present thesis was mostly finished. This work is based on a bilattice approach to logic and addresses issues similar to the acceptability of sentences.

terpretation. Research in the field has shown important results by using three valued logics, default logic, update semantics, bilattices ... All or any of these frameworks have at their disposal enormous expressive power and can therefore account for detailed behaviour with great accuracy. No claim of conceptual significance should be attached to the choice of classical logic. The reasoning behind this choice was that it is generally better known how families of logics are related to classical logic than how they are related to each other, so classical logic seemed a good starting point. The purpose of the work is to study the minimal alterations that must be made to a classical proof theory (Smullyan's semantic tableaux) in order for it to be able to model the observed behaviour of presupposition. This framework is not an attempt to provide yet another knowledge representation framework with claims to modelling presupposition behaviour in a manageable way. The emphasis throughout has been to obtain a simple solution that would fit as much as possible of the observed data. This ability to model the observed behaviour is evaluated by considering whether the resulting proof theory provides predictions about the behaviour of its propositions that match the intuitions that apply to the sentences that have been mapped into them. There are two types of prediction to be considered. On one hand, to predict which combinations of sentences and their presuppositions are unacceptable. On the other hand, to predict when a compound sentence presupposes some of the presuppositions of its components. The thesis shows how most of the required behaviour can be represented in classical logic, with slight extensions for the problematic issues.

The commitment to this very simple underlying framework restricts the classes of examples that may be treated.

Concerning the incremental way in which information is built up from a sequence of propositions, the particular methodology of the tableaux framework is shown to have implicit modelling of this feature. This requires the tableaux to be used as a representational device as well as a decision method. Although this is not an orthodox approach, it is shown to give good results.

The same interest in simplicity led to the decision not to consider repair operations of any sort as part of the interpretation process. This is not due to any belief that such operations have no role to play in interpretation. Rather, it is because they do play such a significant role that I believe it important to understand fully the process of interpretation without them as far as it will go before they are brought to bear on the problem. Although the framework does not allow belief revision, some examples that require it can be considered in the following way: when an example results in an inconsistency, it is assumed that this triggers belief revision mechanisms that are beyond the scope of the present work.

The first part of the thesis (chapters 3,4,5) studies presupposition as outlined above for the propositional case. Chapter 3 provides a formal description of the behaviour of presupposition. The behaviour is shown to be compositional only if the meaning of a compound proposition is understood as the set of truth value assignments to the atomic propositions in it that make the compound true. In that case each of these truth value assignments will require the truth of certain presuppositions, and the presuppositions required by the whole proposition can be worked out from the presuppositions required by each of the truth value assignments that are parts of its meaning. Chapter 4 presents a proof theory that accounts for this behaviour in the case of single sentences and discourses, and shows how it applies to the different examples. Chapter 5 studies the conceptual implications of the given proof theory. In the first part, the proof theory sketched as an auxiliary tool in chapter 4 is related to semantic models for classical logic, and the basic properties of the consequence relation that results are considered. The second part relates the insights obtained on presupposition behaviour with an existing formalization of logical abduction in a similar tableaux framework.

The second part of the thesis (chapters 6 and 7) addresses some of the questions that arise when presupposition is addressed in a finer grained framework based on predicate logic. Because different types of presupposition originate from different linguistic elements within a sentence, it has been found advisable to restrict the analysis of presupposition in the predicate case to certain forms of presupposition. Presuppositions of definite descriptions were chosen because they originate from very basic elements that can be represented in the logic while retaining a structure for the logical representation that is close to the structure of the sentence that is explicit in its syntax. Translations of natural language sentences that introduce connectives in the logical form that were not explicitly present in the original sentence have been avoided. This restricts the expressive power of the framework, but provides a very solid base from which to relate the projection behaviour of presupposition and the logical semantics of the corresponding sentence. Any observations about its behaviour will be framework independent and should translate easily to other applications.

In order to achieve this, an elementary mapping is provided between the basic structure of language sentences and the structure of propositions of predicate logic.

Presuppositions of definite descriptions are the type of presuppositions that has been studied more closely in the past. Underlying its problematic behaviour lies the concept of reference. This problem requires not only a predicate logic framework but an extension to a representation where varying domains are possible.

The final part of the thesis (chapter 8) compares the given formalisms to existing frameworks for the interpretation of presupposition.

Given all these constraints on the choice of representation, it may be surprising that the resulting framework captures reasonably well a wide range of phenomena: the projection of presuppositions as described by Stalnaker [42, 43] and Karttunen [19, 20], their defeasibility as described by Gazdar [8, 9] and Mercer [27, 28, 30, 31, 32] (including a very natural account of the reasoning by cases that is under-explained in Mercer's work), incremental construction of the context as defended by Heim [12, 13, 14] and Beaver [1, 2] (including an account of the global and local accommodation distinction with strong positive guidelines for deciding when each one should be applied), infelicity of constructions involving presupposition as described by Marcu [2] [26] (and even accounts for odd constructions not mentioned by Marcu), the anaphoric behaviour of presupposition satisfaction as defended by van der Sandt [47] (modelled in terms of logical abduction, which allows a very elegant and concise description of the whole range of phenomena, with presupposition presented as a form of adding information to a context), and a sketch of a set of models with differing domains for a language with existence predicates and presupposition of existence.

[2] Of course, some of the finer grain involved in these phenomena has been lost. The difference between infelicity and outright inconsistency is lost. The property of paraconsistency is unavailable in the framework.

1.2 Why Presuppositions

1.2.1 Presupposition

Researchers on the topic of presupposition have tried to explain the elusive relation between propositions like:

(1) *John has stopped beating his wife.*

and:

(2) *John used to beat his wife.*

What is it that makes one accept (2) on hearing (1)? Sentence (2) is not asserted as part of (1). When this happens, (1) is said to *presuppose* (2), and (2) is called a *presupposition* of (1).

This relationship gives rise to interesting problems. Does (1) make sense if (2) is not a true statement? Is (1) acceptable when (2) is not known to be true? Does (2) automatically become true whenever (1) is considered? In that case, what happens if (2) is later discovered to be false? What happens to (1) if (2) is false to start with? Imagine you put together two sentences using a logical connective. What happens to their presuppositions? Does it matter if one presupposes the other? Does it depend on which connective is used? The answer to these questions determines how one should react when trying to interpret sentences that carry presuppositions. The present work sets out to address the issue of whether all these questions can be formalised within one framework.

1.2.2 Types of Presupposition

Many linguistic phenomena have been identified that seem to share the properties of constructions originally termed as presuppositions. They have traditionally been referred to as presuppositions as well.

The following are examples of different types of presupposition. Where possible I have retained the same content for the example sentence so that differences in construction and presuppositional behaviour between different examples stand out. In each case I give the presuppositional sentence followed by the presupposition (or presuppositions) in brackets.

(3.a) *The typewriter is broken.*

(There is a typewriter)

(3.b) *Sam broke his typewriter.*

(Sam has a typewriter)

(3.c) *Sam's typewriter is broken.*

(Sam has a typewriter)

(3.d) *It was Sam who broke the typewriter.*

(Someone broke the typewriter)

(3.e) *The man who broke the typewriter is called Sam.*

(Some man broke the typewriter)

(3.f) *Sam broke the typewriter too.*

(Someone other than Sam broke the typewriter)

(3.g) *Even Sam broke the typewriter.*

(Someone other than Sam broke the typewriter *and* Sam was not the most likely candidate for breaking the typewriter)

(3.h) *Sam broke the typewriter again.*

(Sam has broken the typewriter before)

(3.i) *Bill criticised Sam for breaking the typewriter.*

(Bill believes Sam is responsible for breaking the typewriter *and* Bill believes breaking the typewriter is wrong)

(3.j) *Sam managed to break the typewriter.*

(Sam was trying to break the typewriter *and* breaking the typewriter was not easy)

(3.k) *Sam has stopped breaking the typewriter.*

(Sam used to break the typewriter)

(3.l) *Sam has given up breaking the typewriter.*

(Sam used to break the typewriter *and* Sam did it on purpose)

(3.m) *Sam regrets breaking the typewriter.*

(Sam has broken the typewriter)

(3.n) *Bill realized Sam has broken the typewriter.*

(Sam has broken the typewriter)

This list is not meant to be exhaustive.

The question of whether all these examples correspond to different manifestations of a common phenomenon or whether they are manifestations of different phenomena that present similarities in behaviour is still open. There is a general agreement that the latter is most probably the case (Karttunen and Peters[21]), but they do share a characteristic pattern of behaviour that is not easily captured in conventional formalisms. It is in virtue of these common properties that I consider them as a single class of phenomena, without making any claims about their ultimate conceptual similarity. The properties that require modelling are described below.

The present work concentrates mostly on types (3.a), (3.b), (3.c), (3.k) and (3.m).

1.3 What Needs Formalising

The interesting problems of presupposition arise when a language is used in communication. The typical set up involves describing a certain state of affairs by giving a sequence of sentences of the language. Each consecutive sentence of the sequence adds some new information to a description of the required state of affairs that is constructed progressively.

Presupposition is seen to play apparently different roles in the communication process.

1.3.1 Satisfaction: When Presupposition is True

A common behaviour of presupposition is to find that sentences that have presuppositions are used in situations where the presuppositions can be taken for granted. This happens in a context that already contains the presuppositions. In example (4):

(4) *There is a typewriter on my desk. The typewriter on my desk is broken.*

the first sentence is also a presupposition of the second. When a sentence is used in a context where its presuppositions are already present, it is said that the presuppositions of the sentence are *satisfied* in the context, or that the context *satisfies* the presuppositions of the sentence.

Use of sentences that have presuppositions in contexts where the presuppositions are satisfied is not the only one. Discourse based on this use has an elementary feel about it. It spells out more than is strictly necessary to understand the communication.

Story telling still retains the convention of introducing each presupposition before any presuppositional sentence that requires it. This is what gives fairy tale beginnings their particular flavour:

> (5) *Once upon a time, in a far away land, there lived a princess.*

In this example, not only the far away land and the princess are introduced before they are presupposed anywhere, even the time at which the story takes place is introduced in a similar way at the beginning of the sentence.

1.3.2 Accommodation: When a Presupposition is not Known

Presuppositional sentences are also used in contexts in which their presuppositions are not satisfied. In fact, this has become a very common use in narrative discourse. Nowadays it is more common to find stories beginning more in the manner of:

> (6) *Like a beast, the net came streaming up the ramp and into the sodium lamps of the trawl deck.*
>
> M.Cruz Smith, *Polar Star*

In this case, none of the presuppositions of these sentences (there is a net, there is a ramp, there are sodium lamps, there is a trawl deck, ...) were satisfied by their context of appearance. Yet the sentences are understood without any problems. And the presuppositions get included in the context as if they had been asserted.

This mechanism is used often by writers to give the reader the feeling that he is suddenly in the middle of a story (see Clark and Havilland [5]).

So it seems that a very common reaction if the presuppositions of a sentence are not already in the context when the sentence is interpreted, is to introduce them at that point. This mechanism is referred to as *accommodation* of the presuppositions.

Because there is no simple reply that constitutes a way of objecting to presuppositions, accommodation can be a way of introducing information through a back door.

This is the case of problematic questions like:

(7) *Have you stopped beating your wife yet?*

Phenomena of this kind appear already in the literature on fallacies as the Fallacy of Many Questions. For remarks on their use in argument the reader is referred to Hamblin [11].

1.3.3 When Sentences with Presuppositions are Used to Build Larger Sentences

There are additional phenomena involving presuppositions that require study. They relate to the way in which presuppositions of sentences react when the sentences are combined with connectives like *if ... then*, *and* and *or* to form longer sentences.

The connectives chosen represent a basic logical structure, so the role of reasoning and logical consequence in these phenomena can be studied.

For each one, the observed behaviour that has to be modelled is described below. These descriptions constitute a review of the cases that over the years have been found problematic to model[3].

The reader is invited to test the intuitions presented here by making up examples of his own along similar lines. To my knowledge, no systematic study of speaker's intuitions has been carried out.

Negation

A sentence and its negation have the same presuppositions.

Sentence

[3] For each example, the author who first mentioned it is given in brackets where possible.

(8.a) *The food is not tasty*

presupposes

(8.b) *There is food*

just as much as

(8.c) *The food is tasty* .

This is shown below to present problems when attempting to model presupposition as an entailment.

Conditionals that do not presuppose

One type of conditional that present problems of modelling in terms of behaviour of presuppositions is conditionals where the consequent presupposes the antecedent.

(9.a) *If John was beating the rug then he has stopped. (Mercer)*

Although built up using a sentence that presupposes another, this compound sentence does not presuppose anything. These cases can get very complex when the relationship between the presuppositions of the consequent are hidden amidst problems of reference. A famous example in the literature (first introduced by Gazdar [9]) has the same essential form but relies heavily on identifying a description given in the consequent with a person mentioned in the antecedent:

(9.b) *If Carter invites Angela Davis to the White House, then the president will regret having invited a black militant to his residence. (Gazdar)*

The inference that causes problems is identifying (or not) 'a black militant' with 'Angela Davis'.

Different studies have pointed out still further dimensions to the problem. The example below represents a similar structure where the relationship between the presupposition of the consequent and the antecedent is that the presupposition of the consequent states a logical consequence of the conjuncts that make up the antecedent.

(9.c) *If the dress Mary bought is powder blue and the dress Susan bought is, too, then Mary will regret having bought a dress that is the same colour as Susan's. (Mercer)*

There is another set of examples where the relationship seems to work in the opposite direction, so the antecedent of the conditional presupposes the consequent.

(10.a) *If all Bill's friends have encouraged him, he must have friends. (Gazdar)*

The same relation also works when the presuppositions of the antecedent arise from compound sentences.

(10.b) *If my cousin is a bachelor or my teacher is a spinster, someone at the party is unmarried. (Mercer)*

Conditionals that do presuppose

It has always been accepted that the presuppositions of the antecedent of a conditional become presuppositions of the whole conditional.

(11.a) *If all Bill's friends have encouraged him, he will go ahead with the plan.*

In attempting to formalize this behaviour it is important to keep in mind that there are other types of conditionals where the presuppositions of the consequent do become presuppositions of the whole compound.

(11.b) *If the problem was difficult then Morton isn't the one who solved it. (Soames)*

This sentence presupposes that someone solved the problem [4].

[4]Karttunen and Peters [21] consider this case to be an exception to the general rules, which attribute to conditionals $\alpha \rightarrow \beta$ presuppositions of the form $\alpha \rightarrow \delta$ (where δ is a presupposition of β), see Soames[40, 41], Mercer[27] for arguments against their stand and in favour of the one adopted here.

Ambiguous conditionals

To make the problem even more difficult, there are cases of conditional where it is not clear whether there is a relationship between the presuppositions of the consequent and the antecedent (whereby the conditional would not presuppose them) or not (in which case it would).

> (12.a) *If John came to the party, then the hostess must have been really glad that there was at least one policeman present. (Soames)*

Is John a policeman? Or is he a well known troublemaker and therefore likely to require police presence? In the first case, the conditional does not presuppose anything. In the second case, it presupposes that there is at least one policeman present at the party.

Or, similarly.

> (12.b) *If John has an oriental girlfriend, his girlfriend won't be happy. (van der Sandt)*

Does John's girlfriend resent his flirting with another, oriental, female? Or is she oriental and unhappy because of some other factors that the speaker is aware of (but has not included in his sentence)?

Even sentences that can be classed as one or another type when standing alone can become ambiguous when they are extended with additional information. Sentence

> (12.c) *If someone at the conference solved the problem, it was Julius who solved it.*

on its own seems to be a case of conditional where the consequent presupposes the antecedent. However in some contexts, like

> (12.d) *If someone at the conference solved the problem, it was Julius who solved it, but if it was solved at Nijmegen University it certainly wasn't Julius. (van der Sandt)*

it presupposes 'Someone solved the problem' just as if it were a case of conditional that presupposes the presuppositions of its consequent.

Unacceptable Conditionals

The relationship between presuppositions and connectives can even make certain sentences intuitively unacceptable [5]. This is the case with:

> (13) **If there is no King of France, then the King of France plays golf.*

Disjunctions that Presuppose

In some cases the presuppositions of both disjuncts can become presuppositions of the disjunction. Sentence:

> (14) *Mary stopped beating the rug or John stopped beating the egg. (Mercer)*

presupposes both that Mary was beating the rug and that John was beating the egg.

Disjunctions that do not Presuppose

For disjunction there are also examples where a relationship of presupposition between the disjuncts stops the presuppositions of one of the disjuncts from becoming presuppositions of the disjunct.

> (15.a) *Either John has stopped beating his wife or he hasn't begun yet. (Gazdar)*

In these examples one disjunct is the negation of the presupposition of the other. The symmetrical version also stops the presuppositions.

> (15.b) *Either John hasn't begun beating his wife yet or he has stopped. (Gazdar)*

There is another type of disjunction that does not acquire the presuppositions of its disjuncts.

[5] I follow the convention of writing unacceptable sentences with a preceding asterisk.

(15.c) *Bill has met either the king or the president of Slobovia. (Karttunen)*

In this case the presuppositions seem to be prevented from becoming presuppositions of the disjunct by their mutual incompatibility.

Similar behaviour has been attributed to:

(15.d) *Either Bill has just started smoking or he has just stopped smoking. (Soames)*

(15.e) *Your teacher is a bachelor or a spinster. (Mercer1988)*

Unacceptable Disjunctions

As with conditionals, there are certain combinations of a sentence and its presuppositions by means of disjunction that are unacceptable.

(16.a) **Either there is a King of France or the King of France plays golf.*

(16.b) **Either the King of France plays golf or there is a King of France.*

Conjunctions that Presuppose

Conjunctions that presuppose behave just like disjunctions that presuppose. Sentence:

(17) *Mary stopped beating the rug and John stopped beating the egg.*

presupposes both that Mary was beating the rug and that John was beating the egg.

Conjunctions that do not Presuppose

Whenever the second conjunct presupposes the first, the conjunction does not presuppose it.

(18) *Bill has friends and all his friends have encouraged him.*

Unacceptable Conjunctions

In the case of conjunctions, there are also unacceptable constructions:

> (19.a) **There is no King of France and the King of France plays golf.*
>
> (19.b) **Bill has no friends and his friends have encouraged him.*

and the symmetrical versions:

> (19.c) **The King of France plays golf and there is no King of France.*
>
> (19.d) **Bill's friends have encouraged him and he has no friends.*

There is another combination that results in a slightly different unacceptability. This construction is the symmetrical counterpart of that of conjunctions that do not presuppose.

> (20.a) **The King of France is bald and there is a King of France.*
>
> (20.b) **Bill's friends have encouraged him and he has friends.*

These sentences seem acceptable if understood as being used to emphasize strongly the first conjunct by adding that the necessary conditions for it to be true are met.

1.3.4 Defeasibility: When a Presupposition is False

When a presupposition is false in the context in which the sentence that presupposes it is asserted, there are two different reactions.

If presupposition is considered as an entailment of the sentence, inconsistency should result. This is clearly what is required in the case of positive sentences. Let the context be taken to contain the information that

> (21.a) (*John has no children*).

Assume the following sentence is uttered:

> (21.b) *John's children came to the party.*

This sentence is difficult to accept in the set context. Furthermore, the kind of alteration of the information state that is required to make it acceptable would be elimination or retraction either of sentence (21.b) or the given contextual information. This means that the treatment is similar to that required for resolving logical inconsistency. This is an important intuition on which part of the work presented here will hinge. I am going to assume that for interpretation purposes this sentence must be treated as if it were inconsistent with the context.

The other possible reaction is that observed for the similar example:

> (22) *John's children didn't come to the party.*

In this case, the sentence (22) is used in a context with which its presupposition (that John has children) is inconsistent.

Satisfaction is not possible. Nor is accommodation. (In the resulting context after interpreting the example it does not hold that John has children.) Yet the example can be interpreted. This behaviour is traditionally explained by saying that the presupposition has suffered *cancellation*.

Which one of these reactions is adopted seems to depend on whether the presuppositional sentence is negated or not. However, there is more to the question.

The interaction described so far between the negation of a presupposition ($\neg A$) and a negative sentence that presupposes it ($\neg B^A$) involves several conflicting intuitions.

The behaviour is different depending on whether they are close together in a discourse or not. The example given above is acceptable, but a straightforward discourse of two sentences bearing that same relation to each other, such as

> (23) *John has no children. John's children didn't come to the party.*

is not so easy to accept.

The relative order between the two propositions is also important (the symmetric counterpart of the example (23) above,

> (24) *John's children didn't come to the party. John has no children.*

is much easier to accept.

However, the distance between the sentences also plays a role here. Whereas (24) is acceptable, in a context where it has been accepted earlier as relevant information that

(25) *John's children didn't come to the party.*

it would be at least surprising to hear it said that John has no children. This contrasts with the discourse (24) where both sentences appear close together.

The use of conjunction to join the two sentences also interferes with the interpretation. The example

(26) *John's children didn't come to the party and John has no children*

is not as easily acceptable as (24), its counterpart in two sentences.

(27) *John has no children and John's children didn't come to the party*

seems to remain as unacceptable after joining the sentences as (23) was.

From these examples it seems that the interaction between the negation of a presupposition ($\neg A$) and a negative sentence that presupposes it ($\neg B^A$) is affected by three factors: the distance between $\neg A$ and $\neg B^A$, the relative ordering between $\neg A$ and $\neg B^A$, and the syntactic form of coordination used to relate them.

The first factor seems to be related with whether or not the conflicting sentences originate from the same source. With respect to the $\neg A, \neg B^A$ ordering, people are willing to tolerate this type of conflict between a statement and an existing context, possibly between sources distant in the discourse, but not between sentences that appear close together in the discourse. With respect to the $\neg B^A, \neg A$ ordering, the idea that presuppositions of negative sentences are defeasible can be understood in two different ways. One possibility is to consider that the presuppositions of negative sentences are rejected if they are inconsistent with the information available at the time of processing them. The other possibility is to consider that, in addition to the above, they have to be abandoned when propositions that follow the sentence in the discourse contradict them. In what follows, I assume that interpretation takes place under the first of these possibilities but not the second. That is, presuppositions of negative sentences lose their option on defeasibility once they are interpreted[6]. This assumption follows from the commitment to a homogeneous ultimate representation for assertion and presupposition (people don't necessarily remember whether a certain proposition was presupposed or actually asserted) together with the incremental approach to the interpretation of discourses.

The second factor involves a slightly different turn in interpretation. The reason why

> (24) *John's children didn't come to the party. John has no children.*

is easier to accept is that it has a possible reading as

> (28) *John's children didn't come to the party because John has no children.*

The second sentence is being put forward as a justification or explanation for the first. In actual language use, words like 'because' or 'if ... then' are used to specify this type of connection between sentences. In the case of discourses, words like 'therefore' can play this role. The same role can be played by other indicators, such as the punctuation of the discourse or sentence. Similar behaviour can be observed in discourses even if the particle linking the two sentences is omitted. A table of examples follows:

because
Bill hasn't already forgotten that today is Friday, because today is Thursday. *Mary is not surprised that Fred left because he didn't leave.* *John did not stop beating the rug because he hadn't started.*
If ... then
If Mary's boss does not have children, then it wasn't his child who won the fellowship.
Therefore
There is no King of France. Therefore the King of France isn't in hiding.
Punctuation
Jack's children are not bald; he doesn't have any. *The present King of Buganda is not bald; Buganda is a republic.*
No link shown
The food is not enough. There is no food. *John didn't fail to arrive. He wasn't supposed to come at all.* *Mary isn't sick too. Nobody else is sick besides her.*

[6]It is important to note that any sentences or discourses concerning requests for (or dialogue about) repair operations to the discourse itself are not considered in this work.

This also explains the effect of distance: distant propositions cannot be interpreted so easily as explanation of one another.

The third factor can be understood easily if it is assumed that the use of a conjunction–a connective with no indication of any particular relation between the conjuncts – instead of, for instance, the conditional, or the word 'because', rules out the possibility of interpreting one sentence as an explanation for the other. This could be argued on the grounds of Gricean maxims of conversation.

The behaviour described so far may happen under two different sets of circumstances: when the conflicting propositions originate from the same piece of coherent discourse (as above), or when they all originate (in one way or another) from a single utterance of a compound sentence. The first option is simpler to understand in intuitive terms but is more difficult to study formally because in many situations it is difficult to ascertain whether two consecutive sentences are intended as a continuous discourse or whether a change of context is intended in between. The second option involves cases like the examples:

> (29) *If Mary's boss does not have children, then it wasn't his child who won the fellowship. (Soames)*

1.3.5 Compound Presuppositions

Although in the majority of cases presuppositions seem to be simple sentences, there are examples of presuppositions that involve a certain logical structure.

Presuppositions that are Negated Propositions

There are certain instances where the presupposition of a sentence is a negative proposition.

> (30.a) *Bill started smoking*
>
> (Bill did *not* smoke before)
>
> (30.b) *Even X did Y*
>
> (X was *not* expected to do Y)

(30.c) *X did not either*

(Y did *not* , for some $Y \neq X$)

(30.d) *It was X who did not Y*

(Someone did *not* Y)

Presuppositions of this type are shown to play a role in chapters 3 and 4.

Presuppositions that are Conjunctions

There are also cases where the presupposition of a sentence could be understood as a conjunction.

(3.g) *Even Sam broke the typewriter.*

(Someone other than Sam broke the typewriter *and* Sam was not the most likely candidate for breaking the typewriter)

(3.i) *Bill criticised Sam for breaking the typewriter.*

(Bill believes Sam is responsible for breaking the typewriter *and* Bill believes breaking the typewriter is wrong)

(3.j) *Sam managed to break the typewriter.*

(Sam was trying to break the typewriter *and* breaking the typewriter was not easy)

(3.l) *Sam has given up breaking the typewriter.*

(Sam used to break the typewriter *and* Sam did it on purpose)

Presuppositions that are Disjunctions

There are sentences that can be understood to have a presupposition that has the form of a disjunction, but such presuppositions originate from compounds rather than simple sentences, and the connective involved seems to play a role in the construction of the presupposition. I refer to cases like:

(15.c) *Bill has met either the king or the president of Slobovia*

(Slobovia has *either* a king *or* a president)

(15.d) *Either Bill has started smoking or Bill has stopped smoking*

(*Either* Bill did not smoke *or* Bill did smoke before)

This question is discussed in chapter 4.

Presuppositions that are Conditionals

The possibility of a sentence presupposing a conditional proposition has been proposed[7] as an account of why some conditionals loose the presuppositions of their consequent.

I refer to cases like:

(9.a) *If John was beating the rug then he has stopped.*

The idea is that a sentence of this form would presuppose a conditional made up with its antecedent as antecedent and the presupposition of its consequent as consequent. This would be equivalent to considering that the sentence given presupposes:

(31) *If John was beating the rug then he was beating the rug.*

The fact that this sentence is a tautology would explain why the original sentence seems not to presuppose anything.

The problem with this account is that for a conditional like[8]

(11.b) *If the problem was difficult then Morton isn't the one who solved it*

it predicts a presupposition of the form

(32.a) *If the problem was difficult then someone has solved it*

whereas the sentence rather suggests a simple presupposition

(32.b) *Someone has solved the problem.*

[7] In Karttunen and Peters [21].

[8] This counterexample is due to Soames[40, 41].

Chapter 2

Previous Work

2.1 Presupposition

2.1.1 Origins

Frege

Frege [7] was the first to raise the issue of the kind of phenomena that have come to be referred as ‘presupposition’. He came upon it while studying the question of why different expressions can refer to the same object and yet remain different expressions. To explain this problem he postulated a distinction between the sense of an expression and the reference of an expression. Based on these concepts he summed up the situation by saying that, although each expression should ideally have one definite sense and one definite reference, in actual language use it was frequent to encounter both use of several different expressions for the same reference, and use of several different expressions for the same sense. He then proceeded systematically to explore the consequences of the distinction.

Every grammatical well-formed expression has a sense but not necessarily a reference. By means of a sign we express its sense and designate its reference. When a sign is used in communication it presupposes a reference. For sentences, when not used in quotation, the reference is a truth value, and the sense is a thought.

Quotation stops ordinary reference. When used in direct quotation the reference is a sentence. When used in indirect quotation the reference is a thought. Subordinate clauses may not always have a truth value because: 1) they are used in indirect reference, or 2) they are incomplete because they carry an indefinite indicator that refers to the main clause.

Either the thought or the truth value (the sense or the reference) on its own yields no knowledge.

Frege's observations about language in general (as given above) referred to natural language as used by humans. Many problems in the field of presupposition come from ignoring the distinction that Frege makes between languages as we know them (what is now termed natural language) and the logically perfect language that Frege aspires to define.

On human language, Frege says:

> Languages have the fault of containing expressions that fail to designate an object (although their grammatical form seems to qualify them for that purpose) because the truth of some sentence is a prerequisite.

This contrasts with the specification that he gives for a 'logical' language:

> A logically perfect language should satisfy the conditions that every expression grammatically well constructed as a proper name out of signs already introduced shall in fact designate an object, and that no new sign shall be introduced as a proper name without being secured a reference.

Russell

Russell's Theory of Descriptions provides a simple solution to the problem of interpreting definite references. According to Russell definite descriptions should be identified directly with the object that they describe. He objected to Frege's distinction between sense and reference on the grounds that no connection can be established between the sense of an expression and the object that it refers to. This leads to a general approach to the problem in which in all cases other than those were the existence of an object is explicitly asserted or denied, the use of a description carries the covert assertion that there exists an object which answers to it.

Russell does not use the term presupposition in his discussion of the matter. He tackles the problem of extracting a logical formulation of the information content of language expressions, and he is more concerned with what the actual result should be than with whether there are different ways in which expression can contribute. Russell's theory considers that definite descriptions assert the existence and uniqueness of the element described. Russell claimed that the meaning of

(33.a) *The typewriter is broken*

could be analysed as having the following components:

(33.b) *There is a typewriter*

(33.c) *There is a unique typewriter*

(33.d) *The typewriter is broken.*

This is the approach that tends to be followed in elementary transcriptions of such a sentence into predicate logic:

$$\exists x(T(x) \wedge B(x))$$

Russell defines the presupposition of uniqueness. These presuppositions have not been addressed as extensively as presuppositions of existence in the subsequent literature.

Strawson

Strawson [45] presents a contrasting account in which the existence of the objects referred to by definite descriptions is not considered part of the information asserted by a sentence. Strawson claimed that a sentence like

(33.b) *There is a typewriter*

was not strictly part of the meaning of

(33.a) *The typewriter is broken*

but rather linked to it by a vague relationship that he describes as follows:

- it is not asserted as part of the meaning of the sentence
- it is not entailed by the sentence

- it is a signal carried by the sentence that 'shows but does not state' that this additional information should be considered to be true

Following Frege he called this additional information a *presupposition* of the sentence.

He distinguishes between a sentence S, a use of S and an utterance of S. In a given sentence, such as for instance,

(35) *Our president is ill*

the same descriptive expression can be *used* to refer to different objects in each use. For instance, the sentence above may be used to convey the information that 'Felipe Gonzalez of Spain has flu' in one use and to convey the information that 'Bill Clinton of USA has measles' in another. Whatever the information that a sentence is being used to convey, it may occur more than once with that specific use, for instance when it is used successively to convey the same information to different people. Each one of such occurrences would constitute a different utterance of the sentence. Strawson claims that these distinctions are significant in the sense that one cannot say the same things about a sentence, its possible uses and its possible utterances. A sentence cannot be true or false, unless it is being *used* to make a statement. An expression cannot refer, it is only certain uses of an expression that can be said to refer to objects or persons or occurrences.

Meaning is a general direction for use, 'the set of rules, habits, conventions for its use in referring'. It is the meaning of a sentence that relates that sentence with the possible truth values that the different uses of the sentences may have. Similarly, the meaning of an expression relates that expression with the possible objects or persons that different uses of the expression may refer to.

By means of these distinctions, Strawson introduces the possibility of an uttered sentence being neither true nor false. A sentence must be either true or false only if it is being used to talk about someone or something.

Strawson then explains the presuppositions of a sentence as the necessary conditions for a use of the sentence to be either true or false.

Langendoen and Savin

The study of presuppositions takes a different turn when Langendoen and Savin [23] and Morgan [34] put forward the problem of how (whether) the presuppositions of a compound sentence can be determined from the presuppositions of its components. This is called the *presupposition projection problem.*

In truth, Langendoen and Savin attempted to address the question of how the presupposition and assertion of a complex sentence are related to the presuppositions and assertions of the clauses it contains. The quote below describes the main intention of Langendoen and Savin's work. It relates to how the two different ingredients of a sentence (the assertion and the presupposition) contribute to the logical form of compounds built using that sentence as a subordinate clause.

> If either an assertion or a presupposition contain a variable which stands for a subordinate clause (say, an object complement), then it follows that that variable is replaced only by the assertion of the subordinate clause.
>
> Langendoen and Savin [23]

2.1.2 Intuitions to Return to

Stalnaker

Robert Stalnaker [42] aims to give a general abstract account of the notion of presupposition. He defends that presupposition should be treated pragmatically, not semantically, that is, understood as presupposition of a person, not of a sentence. A person's presuppositions are the propositions whose truth he takes for granted. However, he does accept that certain sentences impose certain constraints on what the speaker may be presupposing. In this way his concept of presupposition includes the consideration of presupposition as a relation between sentences. It also allows description of the cases where sentences require not that a specific sentence be presupposed, but that a sentence of a certain kind be presupposed, without specifying which one (Stalnaker gives the example that a sentence like

(36) *He is a linguist*

presupposes the existence of a certain male, but there is no particular male that is required to exist by each use of the sentence).

Stalnaker [42] observes that as soon as there are established and mutually recognised rules relating what is said to the presumed common beliefs, it becomes possible to exploit those rules by acting as if the shared beliefs were different than they in fact are known to be.

In [44], Stalnaker considers an aspect of the theory of conversations that is worth bringing to mind. Having outlined a theory of presuppositions as those propositions whose truth a speaker takes for granted as part of the background of a conversation, he goes on to distinguish between non defective contexts, in which the presuppositions of the various participants are all the same, and defective contexts, in which they are not. Defective contexts are not stable, and they tend towards non defective contexts. Participants in a dialogue have the motivation (avoiding errors in communication) and the information ('clues' about 'what is presupposed') to notice any discrepancies. Stalnaker infers from this argument that in normal cases contexts are non defective. He does not mention that an intermediate step is required for participants to correct these discrepancies once they have noticed them.

Lewis

David Lewis [25] presents an account of the communication situation designed to include the concept of presupposition. The account is based on the following assumptions: (a) at any stage in a well-run conversation, a certain amount is presupposed, (b) presuppositions can be created or destroyed in the course of a conversation, and (c) presupposition evolves in a more or less rule-governed way during a conversation.

Lewis proposes a model for studying communication situations analogous to keeping score in a game. He postulates the existence of a certain *conversational score*. The components of a conversational score at a given stage are abstract entities. Sentences depend for their truth value, or for their acceptability in other respects, on the components of conversational score at the stage of conversation when they are uttered. Presuppositions can be considered as one type of component of this conversational scoreboard. The conversationalists may conform to directives regarding the development of the score. One such directive is that the presuppositions of any sentence uttered by a participant must already be a component of the conversational scoreboard.

Within his model of conversational scoreboards, Lewis observes that conversational score tends to evolve in such a way as is required in order to make whatever occurs count as correct play. If a speaker has uttered a sentence that has certain presuppositions and those presuppositions were not part of the conversational scoreboard, the scoreboard is altered so that they become part of it. This type of behaviour is called by Lewis *accommodation.* Because such a mechanism exists, participants may purposefully use utterances that do not conform to the conversational directives in order to achieve alterations of the conversational score beyond the scope of simple assertions.

2.1.3 Presupposition in Terms of Gricean Implicature

The work of Paul Grice [10] on the nature and importance of conditions governing conversation gave rise to two different competing approaches that tried to explain the behaviour of presupposition by relying on the new concepts.

Grice observed that during a conversation , information is imparted in ways other than by saying it (suggested, implied, meant ...). He refers to all these ways of imparting additional information as *implicating*, and to the additional information as *implicatures*.

Implicatures can be divided into conventional implicatures (those were the conventional meaning of the worlds used determines what is implicated as well as what is said) and others. Conversational implicatures are a certain subclass of nonconventional implicatures, essentially connected with certain general features of discourse.

Assuming the purpose of a conversation to be a maximally effective exchange of information, Grice specifies an idealised attitude for participants in a conversation in terms of a Cooperative Principle and a set of conversational maxims.

> The Cooperative Principle: Make your conversational contribution as is required, at the stage at which it occurs, by the accepted purpose or direction of the talk exchange in which you are engaged.

The set of maxims consists of the maxim of quantity (make your contribution no more and no less informative the is required), the maxim of quality (make your contribution one that is true), the maxim of relation (be relevant) and the maxim of manner (be perspicuous).

Karttunen

The first of these approaches to presupposition is presented in Karttunen and Peters [21]. Karttunen and Peters' system is an extension of the system for semantic interpretation developed by Montague [33]. They build onto the original system a method of handling presuppositional information based on non-cancelable presuppositions (presented as conventional implicatures) governed by the algorithms specified as filtering conditions.

Presupposition is seen as an *implicature* of sentences. Karttunen and Peters argue that the phenomena described as presuppositions can be explained in terms of several different implicatures in Grice's classification. They show particular interest in those that can be characterised as conventional implicatures. These have a particular characteristic that they cannot be cancelled conversationally. Because of these considerations, the system they propose only treats as proper presuppositions those arising from particles like *too*, *either*, *also*, *even*, *only*, the presuppositions of factive verbs like *forget*, *realize*, the presuppositions of implicative verbs like *manage* and *fail* and the presuppositions of cleft and pseudocleft constructions.

For these, Karttunen and Peters develop a system based on Montague style semantics that operates on two levels: the lexicon assigns to each expression ϕ an *extension* (its asserted content $A(\phi)$) and an *implicature* (its presuppositional content $P(\phi)$).

For example, for a sentence like

(37) *John drinks too*

assuming a transcription of (37) as $\phi =$ *John drinks too*,

$A(\phi) =$ *John drinks*

$P(\phi) =$*Someone other than John drinks*

The main difficulty for this method lies in determining what the presuppositions (as implicatures) of a compound sentence are in terms of the presuppositions of its components. For each expression derived by a syntactic rule, a semantic translation rule assigns extension and implicature to that expression as functions of the extensions and implicatures of the constituent phrases.

For the logical connectives the following rules are given:

$$A(\text{ If } \phi \text{ then } \psi) = A(\phi) \rightarrow A(\psi)$$
$$A(\phi \text{ and } \psi) = A(\phi) \wedge A(\psi)$$
$$A(\phi \text{ or } \psi) = A(\phi) \vee A(\psi)$$

$$P(\text{ If } \phi \text{ then } \psi) = P(\phi) \wedge [A(\phi) \rightarrow P(\psi)]$$
$$P(\phi \text{ and } \psi) = P(\phi) \wedge [A(\phi) \rightarrow P(\psi)]$$
$$P(\phi \text{ or } \psi) = [P(\phi) \vee A(\psi)] \wedge [A(\phi) \wedge P(\psi)]$$

These filtering conditions ensure that a sentence like

(38) *Bill drinks and John drinks too.*

presuppose

(39) *If Bill drinks then someone other than John drinks*

which was taken to be close enough to the observed behaviour (the sentence has no presupposition).

Gazdar

Gazdar [9] considers the addition of a sentence to a given context while taking into account all the additional information that is associated to it as implicatures. This approach leads to a characterization of presupposition according to the following specification.

An utterance of S in a consistent conversational context C presupposes P unless:

- P is incompatible with C, or
- the utterance conversationally implicates that the speaker is not taking P for granted

A speaker conversationally implicates that he is not taking a proposition P for granted whenever he chooses to assert a compound sentence involving P instead of asserting P on its own. This is based on the argument that the speaker would be breaching Grice's maxim of quantity if he chose the compound when to state the proposition would have been more informative. For instance, if someone knows that P is true and asserts $P \vee Q$, he is being less informative than he could be. The assumption that Grice's Cooperative Principle applies to the conversation suggests that whenever $P \vee Q$ is asserted, it is safe to assume that the speaker does not know P, $\neg P$, Q or $\neg Q$.

For instance, the sentence

(40) *If there is a king of France then the King of France is bald.*

has the implicatures that the speaker does not know any of the following sentences:

(41.a) *There is a King of France*

(41.b) *There isn't a King of France*

(41.c) *The King of France is bald*

(41.d) *The King of France is not bald*

Gazdar's system evolves around the concept of *satisfiable incrementation.* The satisfiable incrementation of a context X with a set of propositions Y is just the original context plus all those propositions in Y which cannot introduce inconsistency. For Gazdar, to update a context X with a given utterance P requires computing the satisfiable incrementation of X with the set of all the potential implicatures of P all the potential presuppositions of P and P. Given a proposition, Gazdar applies a set of functions to the proposition to obtain all the additional information on top of what it asserts. A sentence has a set of potential implicatures, referred to as *im-plicatures.* A sentence has a set of potential presuppositions, referred to as *pre-suppositions.* Only those im-plicatures some and pre-suppositions which are satisfiable in a context of utterance emerge as the implicatures and presuppositions of the sentence. For example,

(41.a) *There is a King of France*

is a pre-supposition of

(41.c) *The King of France is bald.*

This information, the implicatures of the proposition, is obtained from the sentence before it is added to the context, and it is all added to the context before the sentence itself. The mechanism that Gazdar proposes to formalize the behaviour of presupposition in complex sentences relies on a subtle division that he establishes between the implicatures of a sentence. According to Gazdar a hierarchy can be established among the implicatures of a sentence, so that they can be added to the context in a progressive manner according to their position in the hierarchy. Conversational implicatures are added to the context first. Then presuppositions are added.

The order of processing conversational implicatures and presuppositions implies that a presupposition can be cancelled if it clashes with any of its own conversational implicatures.

For the example above, once the conversational implicatures have been added to the context, the sentence given as example does not presuppose the potential presupposition (41.a) because it has been cancelled by one of the implicatures.

Soames

Soames [40] discusses both Karttunen and Peters' and Gazdar's approaches. He illustrates with a number of counterexamples how a theory of presupposition must take into account both the defeasibility of presupposition (as modelled by Gazdar) and the projection in compounds (as modelled by the filtering conditions of Karttunen and Peters.

He considers two alternative solutions to this problem, based on trying to combine the two approaches: one where the presuppositions that result from an application of Karttunen and Peters rules are considered as potential presuppositions that can be cancelled and one where any uncancelled potential presuppositions are governed by Karttunen and Peters rules.

First he presents a system in which the inheritance conditions are applied to obtain the presuppositions of compounds. The presuppositions obtained in this way are then considered as cancelable potential presuppositions. This approach presents problems, because the presupposition given by the inheritance conditions, such as, for instance, $\phi^i \wedge [\phi^e \rightarrow \psi^i]$ or $[\phi^i \vee \psi^e] \wedge [\phi^e \wedge \psi^i]$ are too complex to be cancelled by the conversational implicatures of the sentence (which remain simple because they are not subject to the same inheritance conditions).

Second he presents a system in which contextual-conversational cancellation is applied to the pre-suppositions of the sentences, and the remaining presuppositions are then filtered using the inheritance conditions. Soames shows that such a solution describes the behaviour well, and how it allows both a reformulation of the problem in terms of context satisfaction and an explanation of presupposition suspension. However, he acknowledges that this system does little to explain the behaviour that it describes.

2.1.4 Context Change Potentials

The interpretation of presupposition as a requirement has given rise to a different paradigm, based on considering the semantics of a sentence as how it changes the context. The treatment of presupposition involved in this paradigm has its roots in Stalnaker [43], and Karttunen [20].

The Context Change Potential paradigm relies on the following basic concepts.

A context satisfies a set of propositions.

An informative utterance augments the context of interpretation, increasing the set of propositions satisfied. A sentence is said to have a certain Context Change Potential, by which is meant the ability of particular utterances of the sentence to update or increment the context of utterance.

If a sentence presupposes something, then an utterance of the sentence can only be interpreted in contexts which satisfy the propositions that are presupposed.

When a complex sentence is uttered, some parts of the sentence may be interpreted in local contexts which differ from the global context in which the entire utterance is uttered.

Heim

Heim considers a representation of information in term of sets of possible worlds. Interpretation is described in terms of the effect of an utterance (represented by a proposition) on a context (represented as a set of possible worlds). Information is built incrementally by adding a proposition A to a context c. This results in a new context $c + A$ consisting of all the possible worlds that represented c except those where A was not true. The CCP of a single sentence such as A can be seen as a function from incoming context c

that does not satisfy A to outgoing context $c + A$ that differs minimally from c in that it satisfies A.

The semantics of a sentence is defined in terms of its potential to change the information in the context (Context Change Potential, or CCP for short). The CCP of a compound sentence is defined in terms of the CCPs for its individual components.

For sentences built with logical connectives the CCP is defined in terms of the CCPs of the atomic propositions involved:

$$c + \text{ Not } A = c \setminus (c + A)$$
$$c + \text{ If } A,\ B = c \setminus ((c + A) \setminus (c + A + B))$$

where \ stands for set substraction over the sets of possible worlds that represent each context.

A context admits a sentence S just in case each of the constituent sentences of S is admitted by the corresponding local context (as given by the CCP of the sentence).

A presuppositional sentence is admitted in a given context only if its presupposition is already satisfied by that context.

The projection problem then reduces to the fact that the context in which each component sentence appears when it is part of a compound sentence cannot be considered to be simply the context in which the compound sentence appears, but some local context as dictated by the CCP for the compound. For a sentence like:

(40) *If there is a king of France then the King of France is bald.*

the formulation of the CCP ensures that

(41.c) *The King of France is bald*

is only considered in a context to which

(41.a) *There is a King of France*

has already been added.

This approach must be extended in order to account for cases where presuppositional sentences have to be interpreted in contexts that do not satisfy their presuppositions. Such contexts do not admit the presuppositional sentences, so a repair operation on these contexts must be carried out. When the presupposition p of a sentence A is not satisfied in its context of utterance c, the context must be altered to a new context $c\&p$ so that the presupposition p is satisfied. Then A can be interpreted against this new context, by computing $c\&p + A$. This repair operation carried out on contexts is called *accommodation*.

Heim [13] addresses the question of accommodation and proposes a two-tiered account (local and global accommodation) to explain observed anomalies in cases of presuppositions of negative sentences that appear to be defeasible. Faced with accommodation of presuppositions originating under the scope of negation, Heim considers two possibilities, depending on whether the presupposition is accommodated within or without the scope of the negation.

When $\neg s$ has to be interpreted in a context c that does not satisfy p, and $\neg s$ presupposes p, the two alternatives operate as follows:

Global Accommodation	$(c\&p + \neg s) = (c\&p) \setminus (c\&p + s)$
Local Accommodation	$(c\&p + \neg s) = c \setminus (c\&p + s)$

Local accommodation allows the interpretation of a sentence like

(42) *The King of France didn't come to the party*

in a context where it is known that the King of France does not exist.

The two types of accommodation differ essentially in whether cases of accommodation occurring under the scope of negation (either natural or introduced as a result of the interdefinition of connectives) affect the whole context or only the context that is substracted from the whole. In global accommodation, the presuppositions of the negated sentence are accommodated into the previous context before the set substraction is carried out, so accommodation has an effect even after the substraction is carried out. In local accommodation, accommodation is only effected on the set of possible worlds that is substracted from the context, so that its effect on the final representation is greatly reduced.

In logical terms (assuming a classical logic with excluded middle) this distinction is equivalent to considering accommodation in the whole of the previous context (so it survives in the subsequent context where the presuppositional sentence is false) or only in the subpart of it where the presuppositional sentence is true (so the subsequent context is not affected by that accommodation).

Heim's account comes under criticism in Soames [41] and Beaver [1] for not providing clear guidelines as to when each type of accommodation is to be used. Even where tentative criteria can be proposed, the information needed to decide between one and the other is not always present when the decision has to be made.

Heim criticised Karttunen's approach arguing that it allowed the possibility of someone learning the correct rules for the assertive component of language and the wrong rules for the presuppositional component of language. Soames [41] argues that Heim's solution can be criticised on similar grounds: for the same truth conditions, several definitions of the CCPs for the connectives are possible, and only some of them give the right predictions for presupposition. Soames claims that because Heim's framework does not provide independent motivation for the choice of particular CCPs for the connectives, it has no explanatory power.

As well as providing a thorough review of the field, Soames [41] advances some novel ideas on the question of accommodation. Soames offers a more theoretical study of the matter in terms of an additional type of accommodation, in which, rather than accommodate the facts to fit the rules, waiving of the rules is allowed in certain cases (the rule that is waived in this case is the basic tenet of the cancellation approach that the presuppositions of a sentence be added to the context in which the sentence occurs). This results in a division between *de facto* accommodation (changing the recorded information to fit the rules) and *de jure* accommodation (allowing transgressions of the rules in certain cases). This can be seen as an attempt by Soames to reformulate the basic concepts behind the cancellation approach to presupposition in terms of the new concept of accommodation.

Soames [41] puts forward the question of what it is that people actually accommodate, taking a stand against the extended idea that the conditions imposed on context by a presupposition are always of the form of requiring the presence of a specific proposition. For instance a sentence like

(43) *The foreman was fired too.*

does not exactly require a proposition of the form

(44) *Someone other than the foreman was fired.*

but rather that the context contain some appropriate set of propositions such as

(45.a) *John was fired.*

(45.b) *John was not the foreman.*

or a similar set that provides the necessary information even if it does not match syntactically the predicted presupposition of the sentence. This observation by Soames draws attention to a general trend in the literature to interpret presupposition against the logical closure of the context rather than the context as a list of the propositions that have actually been mentioned before. Most subsequent formalisms take this into account (not van der Sandt's DRT approach).

The issue of presupposition with respect to definite and indefinite descriptions has been treated extensively in Heim [12]. In her analysis Heim argues convincingly in favour of the need for a representation of natural language that allows sentences of the representation language not to be quantified. Appropriate treatment of this issue requires a method of keeping track of the terms/referents that have been used so far in the discourse, and provision of means for distinguishing between introducing new terms and referring to terms already present. In Heim's framework this is achieved by replacing the representation of the context as possible worlds with one where context is represented by pairs $< g, w >$ of sequences of referents g and possible worlds w. For a context c, the proposition determined by c is $\{w|$ for some $g, < g, w > \in c\}$. This allows treatment of sentences with free variables. a sentence p with a free variable x_i is true in c if for every $< g, w > \in c$, p holds in w for the corresponding element of g. Similar solutions can be given for quantifiers of the form *every*.

$c+$ Every A, $B = \{< g, w > \in c|$ for every a, if $< g^i/a, w > \in (c + A)$, then $< g^i/a, w > \in (c + A + B)$

Because of the interpretation assigned to sentences with free variables and the fact that for $c+$ Every A, B to be defined $c + A$ and $c + A + B$ must be as well, a sentence like

(46.a) *Every nation cherishes its king*

is predicted to presuppose

(46.b) *Every nation has a king.*

Heim suggests that indefinites may be represented as new free variables. A sentence like

(30)*A fat man was pushing his bicycle*

is represented as

x_i (was a) fat man, x_i was pushing his bicycle

along the same lines as for the case of *every* this formulation gives an incorrect presupposition that

(48) *Every fat man has a bicycle*

unless local accommodation is invoked.

Beaver

Beaver [1] presents a solution based on update semantics. This account essentially extends the work of Heim [12, 13]). Beaver uses an Update Logic, designed to model the potential of a proposition to change an agent's information state. Both information states and propositions are represented in terms of possible worlds. Where Heim uses the definitions of a CCP of a compound sentence, Beaver uses the definition of the update semantics for the connective. The use of this logic, based on Veltman [50] allows Beaver to develop a strong formal treatment and allows him to claim independent motivation for his approach, thereby escaping some of the strongest criticisms presented against Heim. In most other aspects, his work follows Heim closely.

The meaning of an expression ϕ, written $[\phi]$ is defined as a relation between two information states: an input state and an output state. The update of a compound sentence is defined as a sequence of simpler updates that involve only their components. These sequence of simpler updates are defined individually for each connective. The pattern of update required for each connective constitutes a formulation of its update semantics. An information state is a set σ of possible worlds. The expression $\sigma[\phi]$ is the update of σ with an utterance ϕ.

$\sigma[\phi]$ = those worlds in σ where ϕ is true.

$\sigma[\text{ NOT } \phi] = \sigma \setminus \sigma[\phi]$

$\sigma[\phi \text{ AND } \psi] = \sigma[\phi][\psi]$

$\sigma[\phi \text{ IMPLIES } \psi] = \sigma[\text{ NOT } (\phi \text{ AND NOT } \psi)]$

$\sigma[\phi \text{ OR } \psi] = \sigma[\text{ NOT } (\text{ NOT } \phi \text{ AND NOT } \psi)]$

Atomic propositions operate in the same way as in Heim. The connectives are interdefined in the same way as in Heim (so they are also subject to most of Soames' criticisms).

Presupposition is represented as a test of the context: a presuppositional sentence is a valid utterance in a given context if its presupposition has no effect on that context. Given the definitions of logical consequence, this is equivalent to the presupposition being a logical consequence of that context.

Beaver counters Soames criticism of Heim's work by claiming independent motivation for his choice of update patterns. Beaver bases his update patterns in Veltman's [50] update semantics for the logics of epistemic modality.

The whole theory is carefully formulated to ensure that only global accommodation is required throughout. This leaves the issue of defeasibility of presupposition unaddressed.

However, several problems survive from Heim's approach: no satisfactory account for the behaviour of disjunction is provided; the defeasibility of presuppositions is not addressed; and presupposition as an informative operation is only allowed as a repairing modification to the original framework.

Beaver [2] extends the original update semantics account with features of dynamic predicate logic, in order to account for the problems with quantification. In order to tackle this problem he invokes dynamic generalized quantifiers.

2.1.5 Presuppositions as Defaults

Mercer [27, 32] is concerned with the defeasibility of presuppositions. This leads him to a framework based on default logic. This account originates from an approach to the projection problem that concentrates on how presupposition as an inference is defeated when inconsistent with more firmly established information (Gazdar [8, 9]). Mercer's work also reflects some of the problematic cases of defeasible behaviour that Heim analysed in terms of local accommodation. Heim put forward the intuition that defeasible behaviour in presuppositions is closely related with negation. Mercer goes a step further and defines as presuppositions only those that originate from negative sentences (and therefore have defeasible properties), where similar 'inferences' from positive sentences are simply entailments (because they are not defeasible).

A default is

$$\frac{\alpha : \beta}{\beta}$$

where α is the *prerequisite* of the default, the first β is the *justifications* of the default, and the second β is the *consequent* of the default.

A *default theory* Δ is a set of formulae W, and a set D of default rules.

An *extension* E of a default theory is a constructive fixed point having the following properties:

- it includes the set of formulae w of the default theory,
- it is logically closed,
- for all defaults in the theory $\dfrac{\alpha : \beta}{\beta} \in D$, if $\alpha \in E$ and $\neg\beta \notin E$ then $\beta \in E$.

This framework is used to represent the problem as follows. The set of formulae w is used to represent the context. To this is added the representation of the utterance with all its implicatures. The presuppositions of positive sentences come to be in the extension E by reason of being entailments of the utterance (E is logically closed). The presuppositions of negative sentences are represented as defaults, with the negated presuppositional sentence as prerequisite and the presupposition as both justification and conclusion.

If a presupposition is consistent with an extension of the theory, it can be considered as part of that extension. Propositions that appear in all possible extensions of the theory are taken as the actual presuppositions of the sentence. Whenever information in the theory is inconsistent with the presupposition (whether on addition of new information or when adding the presuppositional sentence) the default cannot be used so the presupposition is blocked. This mechanism accounts for the defeasibility of presuppositions.

For instance, the interpretation of

(49) *The King of France did not come*

in a context where there is no King of France might be represented as a theory $w = \{\neg exist(KoF)\}$, an utterance $\neg come(KoF)$, and a default $\frac{\neg come(KoF) : exist(KoF)}{exist(KoF)}$. Because the default is blocked in the context, the sentence is not considered to presuppose *There is a King of France.*

In order to avoid excessive production of presuppositions, Mercer needs to take into account that an assertion of $A \vee B$ or $A \rightarrow B$ carries with it the assumption that the speaker does not know A,$\neg A$, B, or $\neg B$, that is, that he considers an information state where all of them are open possibilities. The consistency checks required for defaults must consider these implicatures. These implicatures have the power to block the defaults that should not go through. However, in order to be able to account for this behaviour, Mercer's system requires long processing of the original proposition, and not every step of it is justified as fully as it could be desired.

Although the formal description is slightly different, the mechanism works along the same lines as Gazdar's.

The way they work can be seen in the example (40), (41.a) to (41.d) given for Gazdar's method.

Let (40) be represented as $exist(KoF) \rightarrow bald(KoF)$, the presuppositional relation as the default

$$\frac{\neg bald(KoF) : exist(KoF)}{exist(KoF)}$$

and (41.a) as $exist(KoF)$, (41.b) as $\neg exist(KoF)$, (41.c) as $bald(KoF)$ and (41.d) as $\neg bald(KoF)$. Mercer's proof theory would carry out the interpretation of (40) as follows.

Case analysis is required for disjunctive statements and any statements with similar semantics. Mercer explains that the default proof theory as given does not allow case analysis. He mentions that a transformed version does and that, although it 'does not have precisely the same properties as the original theory' the difference is insignificant for his purposes. From there on he applies case analysis as if it were valid in the given default proof theory.

Mercer selects the cases that he uses for case analysis by requiring that their possibility be provable from the sentence and its clausal implicatures (which are judgements about possibility). He emphasises that the choice of cases is crucial: selecting too few cases may result in excessive production of presuppositions and selecting too many may result in defective production of presuppositions. He gives two criteria for selecting the cases:

- each case must completely determine the truth value of each of the disjuncts
- the cases must reflect the linguistic situation

To start with the material implication has to be changed to a logical or. Then the selection of cases provides the two options

$$\neg exist(KoF) \wedge \neg bald(KoF)$$
$$exist(KoF) \wedge bald(KoF)$$

In the first case the default might have been applied ($\neg bald(KoF)$ is true, but it is blocked by $\neg exists(KoF)$). In the second case the default does not apply.

The predictions of the framework are to be read as follows. Since the proposition $exists(KoF)$ is simply not true in one case and actually entailed in the other, the sentence (40) is said not to presuppose (41.a).

The advantage of Mercer over Gazdar is that the order of addition of the different implicatures no longer plays a role, because the behaviour is described in terms of a fix point operator. Regarding this advantage, Beaver argues that it results from the fact that the system does not allow the establishment of priorities between defaults. Mercer has solved the problem of motivating a hierarchy of cancelable implicatures by representing only presuppositions as defaults (so they cannot clash with implicatures). But this forces all implicatures to become full entailments even where they should be as defeasible as presupposition.

On the other hand, the fact that complex processes are required to justify the necessary implicatures for anything other than very simple compounds weaken the adequacy of Mercer's method. This is addressed in more detail in chapter 8.

2.1.6 Presuppositions as Anaphora

Sandt [47] argues that presuppositions should be neither understood as referring expressions, nor explicated in terms of a non-standard logic, nor explained in terms of pragmatics, but rather read as anaphoric expression with extra descriptive content. This extra descriptive content allows presuppositions to establish a reference marker in case discourse does not provide one. In this case, the lexical material is accommodated.

The framework provides predictions on the behaviour of presuppositions as determined by the place along the discourse structure where the presupposition is accommodated.

The approach of van der Sandt is presented within the framework of DRT (Kamp and Reyle [18]). This framework provides a representation of discourse markers that can be easily used to model candidate discourse referents (including a binding operation between new markers and markers appeared before). Such a representation provides a structuring of the discourse in terms of nested DRSs over which the constraints can be easily defined as a search path.

In a DRS representation, discourses are divided into two different sets of ingredients: a set of discourse referents (the universe of discourse), and a set of conditions to be satisfied by these referents. Indefinite NPs introduce discourse markers into the universe of the DRS. These markers then serve as referents. Conditions assign properties to the members of the universe of discourse. Anaphoric elements are encoded separately in a DRS. They have to be incorporated into the structure by a process of resolving the anaphoric expressions. Anaphoric constructions constitute an instruction to look for the appropriate referent (a marker satisfying conditions equal or compatible with those of the anaphoric expression) somewhere earlier in the discourse.

This resolution can involve two different processes. If a referent is found earlier in the discourse, anaphoric binding takes place. The corresponding discourse markers are linked by putting the appropriate equations and the conditions associated with the anaphoric expression are transfered to the binding site. If no referent is found both the markers and the conditions are added to the structure at the accommodation site.

The presuppositions of

(50) *John‘s cat purrs*

would be represented as:

$$\begin{array}{|c|}\hline \text{x} \\ \hline \text{cat(x)} \\ \text{poss(y,x)} \\ \begin{array}{|c|}\hline \text{y} \\ \hline \text{John(y)} \\ \hline \end{array} \\ \hline \end{array}$$

This involves not only

(51.a) *John has a cat*

but also information analogous to

(51.b) *John exists.*

For sentence (5) the interpretation process would run as follows. A first representation is obtained of the form:

$$\begin{array}{|l|}\hline \\ \hline \text{purrs(x)} \\ \begin{array}{|c|}\hline \text{x} \\ \hline \text{cat(x)} \\ \text{poss(y,x)} \\ \begin{array}{|c|}\hline \text{y} \\ \hline \text{John(y)} \\ \hline \end{array} \\ \hline \end{array} \\ \\ \hline \end{array}$$

the interpretation of the presuppositions tries to find referents in the big box, and when it does not, it finds an appropriate location to place them. This leads to:

$$\begin{array}{|l|}\hline \text{y, x} \\ \hline \text{John(y)} \\ \text{cat(x)} \\ \text{poss(y,x)} \\ \text{purrs(x)} \\ \hline \end{array}$$

For compound sentences, the framework provides predictions determined by the constraints imposed on the search path for referents. These run as follows:

From	Allowed to search
antecedent of a conditional	outside the box
consequent of a conditional	in antecedent and outside the box
disjuncts	in the other disjunct or outside the box

For a sentence like

(52) *Either John has no donkey or his donkey is in the stable*

the method gives a final representation:

$$\begin{array}{|l|}\hline \textbf{x} \\ \hline \textbf{John(x)} \\ \neg \begin{array}{|l|}\hline \text{y} \\ \hline \text{donkey(y)} \\ \text{poss(x,y)} \\ \hline \end{array} \vee \begin{array}{|l|}\hline \textbf{z} \\ \hline \textbf{donkey(z)} \\ \textbf{poss(x,z)} \\ \text{in-the-stable(z)} \\ \hline \end{array} \\ \\ \hline \end{array}$$

The presupposition (in bold typeface) has had to be accommodated locally because otherwise the constraints would have been violated. This ensures that the sentence is predicted not to presuppose the existence of a donkey.

This is equivalent to example (15.b).

In many cases, more than one resolution is possible. Among the set of all logically possible interpretations, first the admissible interpretations and then the preferred interpretation must be selected. Sandt provides criteria for this.

This solution can account for many of the problems that faced the update semantic account. It captures an essential aspect of the nature of presupposition (its anaphoric nature), but it fails to address its direct relation with the semantics required for interpreting these connectives in a logical sense. This problem is addressed in chapter 8.

2.1.7 The Bilattice Approach

Marcu [26] and Schoter [37] address the issue of presupposition in the light of the theory of bilattices. Schoter presents a powerful logical formalism that can account for the properties of presupposition that escape classical logic (defeasibility, partiality, paraconsistency, relevance). Marcu addresses the issue of defeasibility in two different stages, according to whether the information obtained is felicitously or infelicitously defeasible.

2.2 Controversial Cases

Of all the examples given in chapter 1 there are several which are correctly handled by all the existing frameworks. On the remaining ones, the predictions of the different frameworks differ.

The controversial examples are the following ones.

2.2.1 Examples That Should Presuppose and Do Not

For statements of the form $X \rightarrow (\partial B \wedge A)$, Beaver's framework predicts a presupposition $X \rightarrow B$. In the case of

> (11.b) *If the problem was difficult then Morton isn't the one who solved it. (Soames)*

this would be equivalent to the sentence presupposing that if the problem was difficult then someone solved it. This is not intuitive.

2.2.2 Examples That Should Not Presuppose and Do

> (15.b) *Either John has stopped beating his wife or he hasn't begun yet. (Gazdar)*

Beaver's framework, although it gives the correct predictions for the symmetric counterpart of this sentence, incorrectly predicts a presupposition

> (56) *John used to beat his wife*

for this sentence. This is related to the fact that it defines disjunction in terms of conjunction, and conjunction is defined as asymmetric.

The update framework gives the correct predictions for the symmetric counterpart of this sentence, but incorrectly predicts a presupposition (56) for this one. This is related to the fact that it defines disjunction in terms of conjunction, ($\sigma[\phi \text{ OR } \psi] = \sigma[\text{ NOT (NOT } \phi \text{ AND NOT } \psi)])$ and conjunction is defined as asymmetric.

This means that by the time when the first disjunct is being processed the second disjunct is still being queued up behind a conjunction and so cannot play a role. Therefore the presupposition is accommodated. Considering that each negation that appears in the formulation of the CCP can act as a site for a local accommodation solution, a solution to this difficulty might be found by using some form of local accommodation. But even for that the order in which the information is forced to be available presents difficulties.

Another problematic case is:

> (57.a) *The King of France didn't come.*

when it is interpreted in the context of

> (57.b) *(There is no King of France.*

This corresponds to the types of example where presupposition is cancelled, or acts as a defeasible inference, or accommodated locally in other frameworks. Beaver and van der Sandt have no way of accounting for this behaviour. both their frameworks predict this discourse to lead to inconsistency. The presupposition ought to be defeasible, making the discourse acceptable.

2.2.3 Examples That Should be Predicted as Unacceptable

The cases

> (13) **If there is no King of France, then the King of France plays golf.*

> (16.a) **Either there is a King of France or the King of France plays golf.*

(16.b) **Either the King of France plays golf or there is a King of France.*

are unacceptable in virtue of the interaction between the asserted content of the sentences and the presuppositions of thee sentence. It seems fair to expect a theory of presupposition to have something to say about this interaction.

At a different level, not altogether unacceptable in the same sense but odd and requiring an explanation, stand the cases like:

(20.a) **The King of France is bald and there is a King of France.*

(20.b) **Bill's friends have encouraged him and he has friends.*

In these cases, there are strong intuitions that suggest that such constructions should not be acceptable. A theory of presuppositions should provide an explanation as to why this should be so.

Chapter 3

A Description of the Behaviour of Presuppositions

3.1 The Problem of Compositionality for Presuppositions

Although presupposition originates as a natural language phenomenon, for the purposes of the next few chapters sentences will be represented as propositions of a logical language. At this level of granularity, presupposition can simply be represented as a relation between sentences. For instance, the sentence

(58) *The typewriter is working*

can be said to presuppose

(59) *There is a typewriter.*

Where language is simplified in this way, the internal construction of each sentence does not play a role. The representation of presupposition can be restricted to defining the ordered pairs of sentences for which this relationship holds. I assume that such a relation of presupposing is given for the atomic formulas of the language. To make this information conspicuous without introducing too many definitions, I denote the fact that 'A presupposes B ' by writing each instance of A as A^B. It must be kept in mind that A^B

is only a notational device and $A^B \equiv A$ for logical purposes. In terms of this notation, the behaviour of presupposition concerning negation has the following implication: $\neg(A^B) \equiv (\neg A)^B$ (For simplicity, I leave out the parentheses in these cases from now on.)

For complex sentences the relation of presupposing has to be worked out, ideally in a compositional way.

A formal language has the property of *compositionality* if it is possible to describe the meaning of a complex expression of the language in terms of the meaning of its parts. It is considered a desired property for any formal language. When a logical statement is composed from propositions that presuppose other propositions, it should be possible to describe the presuppositions of the resulting complex expression in terms of the presuppositions of its parts. The actual problem is not so simple. The presuppositions of the parts interact with the parts themselves and this interaction may result in a reduction of the presuppositional information contributed by the part to the whole. Any attempt at a compositional description of the behaviour of presupposition must take into account: the presuppositions of the parts, the parts themselves, and the interaction between them.

If one takes the natural language connectives *if ... then*, *and*, and *or* to be related to material conditional, conjunction, and disjunction, natural language examples (like those given in chapter 1) provide some clues as to what the behaviour of presupposition should be. The behaviour of presuppositions of sentences of this form has traditionally been studied as part of the *projection problem for presuppositions*, which is concerned with describing the presuppositions of a sentence in terms of the presuppositions of its subordinate clauses. The constructions considered in the projection problem involve nested subordination (verbs of propositional attitude, factive verbs) beyond the natural language connectives treated here.

The description of the behaviour of presuppositions is undertaken in this chapter by means of rules that govern their compositionality. In order to obtain a simpler formulation of these rules, a specific representation of the semantics of the connectives is required. This representation is described in the next section.

3.2 The Language and Presupposition

3.2.1 The Logical Language to be used

The Language L

Information will be represented by a language L.

Definition 1 *The alphabet of L consists of non-descriptive symbols:*

- *sentential connectives* $\rightarrow, \vee, \wedge$
- *auxiliary signs: parentheses (,).*

Together with descriptive symbols:

- *a set of propositional constants* $p, q, r \ldots$

Definition 2 *A is an atomic formula of L if and only if A is a propositional constant of L.*

Definition 3 *Inductive definition of formula in L:*
1) an atomic formula in L is a formula in L
2) If A is a formula in L, then so is $\neg A$
3) If A and B are formulas in L, then so are $(A \wedge B)$, $(A \vee B)$, and $(A \rightarrow B)$.

Mapping Natural Language onto L

For the purpose of illustrating the relationship between the natural language examples and the logical language that is to be used to represent them, the following mapping from natural language sentences onto propositional letters is given.

Sentence	Propositional letter
The typewriter is blue	b
There is a typewriter	t
Sue is happy	h
Bill regrets there is no hot water left	r
There is no hot water left	w
Mary has had a bath	m
Bill has started smoking	e
Bill has stopped smoking	p
Bill smoked	s
John is married	j
John has children	c
John's children are at school	a
John's children have forgotten Bill	f
Bill regrets that John's children have forgotten him	g

The choice of propositional letter for each natural language sentence is arbitrary, but the assignment overall must be such that the relations of presupposition between sentences are preserved between the corresponding propositional letters.

The following relations of presupposition are relevant to this set of case sentences:

Proposition	presupposes	Proposition
b		t
r		w
e		$\neg s$
p		s
a		c
f		c
g		f

The Use of Metavariables

In some cases during the discussion that follows, metavariables $A, B, C \ldots$ are used to refer to propositions of L. A metavariable A stands for any proposition of L. It must be noted that in the case of A^B, A and B stand for any propositions p and q of L such that the sentence that is mapped into p presupposes the sentence that is mapped into q.

Discourses

When logical statements are strung into a sequence of assertions, a *discourse* is obtained. For propositions $P_1, P_2, \ldots P_n$, I denote the discourse obtained from them as $(P_1) \circ (P_2) \circ \ldots \circ (P_n)$. The symbol $\circ$ operates as a sentence concatenation operator. It is a well established problem in natural language interpretation that the position of a sentence within a discourse is significant to its interpretation. Therefore, it is interesting to consider a formalization in which the position of a formula P_i in a discourse is significant to its interpretation. An appropriate treatment of discourses is required to study the effect of context on presupposition interpretation.

3.2.2 Presuppositions of Atomic Propositions and Context

In order to obtain a model of the behaviour of presuppositions it is important to consider the role of propositions in describing a specific state of the world. When atomic propositions A, B, C are held to be true together, it can be said that they describe the generic state of the world in which they are all true. For instance, the propositions corresponding to the sentences:

(60.a) *There is a door to my left.*

(60.b) There is a window to my right.

(60.c) There is a computer in front of me.

(60.d) I am typing.

describe a generic state of the world that is true at the present moment (and at countless others in the recent past).

This concept of propositions being simultaneously true in a given situation plays an important role in describing the behaviour of presupposition so a formal system intended to deal with presupposition needs a way of representing it explicitly. Take the propositions that describe such a generic state of the world and pile them up. Call this a *pile*. For propositions A, B, C, the corresponding pile would be:

$$\begin{array}{c} A \\ B \\ C \end{array}$$

Such a pile represents the generic state of the world that the propositions characterize. In order to avoid confusion at later stages, it is important to note here that the positions of the different propositions in the pile are not significant. The following piles are all equivalent to one another:

$$\begin{array}{ccccc} A & B & C & A & \\ B & C & A & C & etc \\ C & A & B & B & \end{array}$$

The presuppositions of atomic sentences introduce unexpected complexity in this simple representation. In an attempt to avoid controversy at this early stage of the development of the system, it can be said that an atomic proposition A^B *signals* that not only A but also B is to be considered true in the state of the world under consideration. If A^B were

(61) *The computer is ancient.*

then the state of the world being described must be one where there is a computer, so it is actually a state of the world that can be described with two propositions, A^B and B, where B would stand for

(62) *There is a computer.*

The proposition A^B, when taken on its own, in fact describes a pile of the form

$$\begin{array}{c} A^B \\ B \end{array}$$

The definition of how this signalling operates, and, more specifically, how the presuppositions as signals ought to be handled during interpretation, will emerge progressively as the behaviour of presuppositions is discussed in different contexts. In the case above, it seems that the signal is to be interpreted as a positive indication to consider the pile extended with B rather than one containing only A^B.

This behaviour is also observed when the pile holds the negation of the presupposition, that is, when it represents the state of the world described by the sentences

(63) *There is no computer.*

(61) *The computer is ancient.*

The pile to represent such a state of the world would be

$$\begin{array}{c} \neg B \\ A^B \end{array}$$

This pile signals B and so is equivalent to

$$\begin{array}{c} \neg B \\ A^B \\ B \end{array}$$

The differences between this case and the previous one lie in the fact that the pile that results from considering the presupposition as well as the given information cannot represent any actual state of the world. In logical terms, the pile represents an *inconsistent* state of the world.

However, if the pile holds the presupposition itself, the observed behaviour presents a peculiarity. A pile

$$\begin{array}{c} B \\ A^B \end{array}$$

represents exactly the same generic state of the world as a pile containing only A^B. This suggests that in this case the signal that the presupposition provides is not relevant to the interpretation. The interpretation of the presupposition simply seems to produce redundant information. This apparent redundancy is studied more closely in chapter 5, where it is shown how it relates to the anaphoric properties of presupposition.

One of the intriguing properties of presupposition has always been that negating the sentences that it originates from seems not to affect its behaviour. This is not completely true. The behaviour of presupposition in an empty context is not affected (it also acts as a signal to introduce the presupposition as additional information). For the example above,

(64) *The computer is not ancient.*

intuition does suggest that the pile

$$\begin{array}{c} \neg A^B \\ B \end{array}$$

represents its information content faithfully. When the pile holds the presupposition as well as the presuppositional proposition,

$$\begin{array}{c} B \\ \neg A^B \end{array}$$

the same apparent redundancy as in

$$\begin{array}{c} B \\ A^B \end{array}$$

is observed.

The behaviour differs considerably when the pile holds the negation of the presupposition. A state of the world described by the sentences

(64) *The computer is not ancient.*

(63) *There is no computer.*

need not be inconsistent. However, it is awkward in some way. The signal is still there, but it seems to be overcome by the additional information in the pile. The appropriate pile to represent such a state of the world would be

$$\begin{array}{c} \neg A^B \\ \neg B \end{array}$$

Here again it is important to remember that the position in the pile is not significant, so the differences in behaviour according to the relative position of the presuppositional sentence and the negation of the presupposition (as discussed in chapter 1) cannot be taken into account. As a tentative approximation, it is assumed that whenever such two propositions coexist in a pile, the signal is not taken into account. This approximation gives good results over the framework developed later.

The results can be summarised in the following table:

Proposition to be interpreted

Relevant context		A^B	$\neg A^B$
	$\emptyset$	add B	add B
	B	add$B$ (redundant)	add B (redundant)
	$\neg B$	add B ($\perp$ results)	

So it seems that the signal that a presuppositional sentence carries is to be interpreted differently according to the context of interpretation. Falling back on Strawson's distinctions, the same sentence can be used in different ways, and some of these uses signal additional information and others do not.

The predictive power of these elementary criteria can be tested against a slightly more complex example. For instance, it is interesting to consider what predictions result from applying these criteria to the case where two signals appearing in the same pile do not conflict with their general context, but conflict with one another. The predictions vary according to whether the signals originate from positive or negative propositions. For the situation

$$\begin{array}{c} A^{\neg B} \\ C^{B} \end{array}$$

the representation would correspond to a state of the world described by the sentences

(65.a) *Bill has started smoking.*

(65.b) *Bill has stopped smoking.*

Both B and $\neg B$ are forced to be true simultaneously in the same world. As a result, the world represented by the pile should become inconsistent. The particular example used here allows both B and $\neg B$ to be true in the same state of the world by introducing a time difference between the moment at which one proposition is considered to be true and the moment at which the other one is considered to be true.

The difference in behaviour introduced by negation is apparent in the following situation.

$$\begin{array}{c} \neg A^{\neg B} \\ C^{B} \end{array}$$

This representation would correspond to a state of the world described by the sentences

(65.c) *Bill has not started smoking.*

(65.b) *Bill has stopped smoking.*

The criteria suggest that these sentences describe a state of the world as represented by the pile

$$\begin{array}{c} \neg A^{\neg B} \\ C^{B} \\ B \end{array}$$

In this case one signal (B) has prevailed over the other one because one originated from a negated proposition (which allows two interpretations, $\neg A^{B}, B$ and $\neg A^{B}, \neg B$) and the other one from a positive one (which only has one possible interpretation, C^{B}, B). The presence of the signal from the positive proposition forces one of the two possible interpretations for the signal originating in the negative proposition.

Not surprisingly, the choice of proposition that is negated does not greatly affect the behaviour. For

$$\begin{array}{c} A^{\neg B} \\ \neg C^{B} \end{array}$$

the representation would correspond to a state of the world described by the sentences

(65.a) *Bill has started smoking.*

(65.d) *Bill has not stopped smoking.*

The same behaviour described above here forces the signal $\neg B$ to prevail in this case.

The only surprise comes in the situation where both propositions are negated simultaneously.

$$\neg A^{\neg B}$$
$$\neg C^{B}$$

This representation would correspond to a state of the world described by the sentences

(65.c) *Bill has not started smoking.*

(65.d) *Bill has not stopped smoking.*

This case seems to require an addition to the criteria outlined earlier. The situation described seems not to signal anything whatsoever about the truth or falsity of B. In terms of the criteria presented earlier, this would correspond to adding an extra line to the table given earlier.

		Proposition to be interpreted	
		A^B	$\neg A^B$
Relevant context	$\emptyset$	add B	add B
	B	add B (redundant)	add B (redundant)
	$\neg B$	add B ($\perp$ results)	
	$\neg C^{\neg B}$	add B	

3.2.3 Representing the Connectives

The representation of propositional information in terms of piles can also be used to represent sentences constructed with the connectives of language L. In order to be able to use the intuitions resulting from examples built with natural language connectives, the choice of pile representation for the connectives of L is made based on intuitions about the corresponding natural language connectives.

A proposition of the form $A \wedge B$ corresponds to a sentence like

(66) *John is thin and Mary is fat.*

and intuition suggests that the information it contains should be represented as a pile of the form

$$\begin{array}{c} A \\ B \end{array}$$

Both propositions involved are intended to be true in the same state of the world. Natural language conjunction may involve many different nuances that depart from this basic analysis, but this approximation will do for a basic study of presupposition behaviour. Some of these additional complexities are considered in chapter 4.

A proposition of the form $A \vee B$ corresponds to a sentence like

(67) *Either John is thin or Mary is fat.*

In this case the intuitions are more difficult to represent in terms of piles. A first possibility would be to represent such propositions as pairs of piles, such as

$$A \qquad B$$

Indeed, this is the alternative traditionally followed in most logical representations. However, this representation may present problems when attempting to predict the behaviour of signals originating in either of the disjuncts. In each of the resulting piles there is only information about one of the propositions. To ignore the information about one of the propositions when trying to ascertain the behaviour of signals arising from the proposition in the other pile may result in confusing predictions. The possible combinations of information about A and B that shows explicitly information about all the propositions involved and still match the requirement that at least one of the propositions be true in each represented state of the world are captured by the set of piles

$$\begin{array}{ccc} A & \neg A & A \\ B & B & \neg B \end{array}$$

This corresponds to the interpretation of $\vee$ as inclusive or. Natural language *or* may carry the implicature that the alternative where both disjuncts are simultaneously true not be considered. This is not taken into account here.

A proposition of the form $A \rightarrow B$ corresponds to a sentence like

(68) *If John is thin then Mary is fat.*

As above, a first approximation would be to represent this as a pile

$$\begin{array}{cc} \neg A & A \\ & B \end{array}$$

This is the representation traditionally chosen for the material conditional in logical representations. It is important to consider that such sentences (and propositions) may refer to states of the world where simply the antecedent A is not true. This is obvious if one considers examples such as

(12.d) *If someone at the conference solved the problem, it was Julius who solved it, but if it was solved at Nijmegen University it certainly wasn't Julius.*

where it is clear that if each conditional $A \rightarrow B$ is only interpreted in terms of A and B being true simultaneously, part of the sense of the sentence is lost. Such a sentence refers to two distinct possible states of the world, each one identified by the antecedent of one conditional, and specifies for each one some additional information (given by the consequent of the corresponding conditional). In order to understand the sentence correctly it is crucial to accept that each conditional does not exclude the possibility of its antecedent being false.

The same considerations as in the case of disjunction concerning the prediction of the behaviour of signals suggest that an extension of the representation should be considered so that $A \rightarrow B$ is represented by the following set of piles

$$\begin{array}{ccc} \neg A & \neg A & A \\ B & \neg B & B \end{array}$$

3.2.4 Presuppositions of Compound Propositions and Context

Compound sentences can have presuppositions just as much as atomic sentences. But the way in which these 'presuppositions' are related to the presuppositions of the atomic sentences involved is not easy to determine at first sight.

The observations earlier on about the behaviour of presuppositions in individual piles suggest a simple answer to how the presuppositions of the atomic sentences in each pile of the representation for a compound sentence determine overall the 'presuppositions' of the compound sentence.

Example (69) is a case of behaviour of presuppositions originating in the antecedent of a conditional (example (11.a) in chapter 1). Sentence

(69) *If the typewriter is blue then Sue will be happy*

corresponds to the following representation.

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{b^t \rightarrow h}} \\ \hline \neg b^t & \neg b^t & b^t \\ h & \neg h & h \end{array}$$

In all the resulting piles there is a proposition b^t. The criteria described earlier for individual piles suggest that each of these piles signals the truth of t. Since all the possible piles for this sentence signal the same presupposition, it stands to reason that the proposition as a whole also signals the truth of the presupposition (it signals its truth in every possible state of the world that it refers to). These observations only hold universally when the sentence (proposition) is used in an empty context.

The sentence

(70.a) *Bill regrets that there is no hot water left*

presupposes

(70.b) *There is no hot water left.*

The problem is to determine what the presuppositions are for sentence

(70.c) *If Mary has had a bath, then Bill regrets that there is no hot water left*

(which corresponds to example (11.b) in chapter 1). Assume sentence (70.c) has the form $m \rightarrow r^w$. The representation for this sentence in this framework would be:

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{m \rightarrow r^w}} \\ \hline \neg m & \neg m & m \\ r^w & \neg r^w & r^w \end{array}$$

The same argument given above suggests that the sentence signals the truth of w.

Example (71) is a case of behaviour of presuppositions originating in the consequent of a conditional, in the particular case where the presupposition itself forms the antecedent (examples (9.a) to (9.d) in chapter 1). Sentence

(71) *If there is a typewriter then the typewriter is blue*

corresponds to the following representation.

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{t \rightarrow b^t}} \\ \hline \neg t & \neg t & t \\ b^t & \neg b^t & b^t \end{array}$$

The first pile is predicted to signal t, but it is also predicted to represent an inconsistent state of the world, and therefore need not be taken into account. For the rest of the individual piles, each one holds such propositions that the criteria predict that no additional information is signalled. The proposition represents only states of the world where no additional signals have to be considered, so the proposition cannot be said to signal anything. Incidentally, this does not mean that information on the truth of t is lost, since such information is already available by other means.

Disjunction presents the most problems in attempts to model intuitive behaviour in terms of compositionality. Example

(72.a) *Either there isn't a typewriter or the typewriter is blue*

shows the case where the presuppositions of one of the disjuncts are not presuppositions of the disjunction (example (15.b) in chapter 1).

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{\neg t \vee b^t}} \\ \hline \neg t & \neg t & t \\ b^t & \neg b^t & b^t \end{array}$$

This sentence corresponds to a structure that is equivalent to that for the conditional of example (71), and gives no additional signals for the compound.

For the sentence in example

(72.b) *Either the typewriter is blue or there isn't a typewriter*

(which corresponds to example (15.a) in chapter 1) the corresponding expansion contains the same literals in each pile but in a different order.

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{b^t \vee \neg t}} \\ \hline b^t & \neg b^t & b^t \\ \neg t & \neg t & t \end{array}$$

Since each pile represents no more than a static snapshot of a state of the world, the order in which the propositions appear in the pile has no significance[1]. The intuitions are the same as for the previous case.

Take for instance the sentence in example

(73) *Either Bill has started smoking or Bill has stopped smoking.*

(This is the same sentence that was given as example (15.b).) This has the following representation:

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{e^{\neg s} \vee p^s}} \\ \hline e^{\neg s} & e^{\neg s} & \neg e^{\neg s} \\ p^s & \neg p^s & p^s \end{array}$$

[1]Ordering does seem to play a role in the intuitive validity of conjunctions (*There is a typewriter and the typewriter is blue* is acceptable, *The typewriter is blue and there is a typewriter* is harder to accept). This issue is discussed further in chapter 4.

In this case the criteria applied to the piles give no signal for either, but for slightly different reasons. The piles involved here are instances of the examples used earlier to show the predictive power of the criteria for a single branch. Each pile signals a different proposition. In each case the signal can be used as additional information for the corresponding pile. But because the signals of the different piles don't match, there can be no overall signal for the proposition. Chapter 4 explores the possibility of expressing a complex (disjunctive) signal for these cases.

Sentences corresponding to more complex states of the world can be constructed using these same connectives. In order to obtain a representation for these sentences in the same way as those considered above, it is convenient to borrow a construction method used for analytic tableaux (Smullyan [38] and later Fitting [6]).

3.3 The Tableau Interpretation

3.3.1 Tableaux as Representation for Sentences and Discourses

The logical connectives I am considering have their own definition of compositionality with respect to truth value. The truth value for a compound can be obtained from the truth values of the propositions that go into building it by application of well known rules. These rules can also be used to establish the possible truth values of the propositions involved in a compound given an initial assumption that the compound is true. Because I am concerned with assertion of logical statements and the information that they convey (under a presumption of truth), I will be dealing mostly with this second use of the rules. This use can be specified in terms of truth tables. For each generic compound, the compound is assumed to be true, and the truth table is used to determine which are the possible values of the atomic propositions involved that would make that compound true.

In the present framework, these definitions are represented as tableaux expansion rules:

$$\neg - rules) \qquad \frac{\neg\neg P}{P}$$

$$\alpha - rules) \qquad \frac{P_1 \wedge P_2}{\begin{array}{c} P_1 \\ P_2 \end{array}} \qquad \frac{\neg(P_1 \rightarrow P_2)}{\begin{array}{c} P_1 \\ \neg P_2 \end{array}} \qquad \frac{\neg(P_1 \vee P_2)}{\begin{array}{c} \neg P_1 \\ \neg P_2 \end{array}}$$

$$\beta - rules) \qquad \frac{P_1 \vee P_2}{\begin{array}{ccc} P_1 & \neg P_1 & P_1 \\ P_2 & P_2 & \neg P_2 \end{array}} \qquad \frac{P_1 \rightarrow P_2}{\begin{array}{ccc} \neg P_1 & \neg P_1 & P_1 \\ P_2 & \neg P_2 & P_2 \end{array}} \qquad \frac{\neg(P_1 \wedge P_2)}{\begin{array}{ccc} \neg P_1 & \neg P_1 & P_1 \\ \neg P_2 & P_2 & \neg P_2 \end{array}}$$

Some additional definitions are needed to introduce this way of understanding the representation. Apart from the expansion rules, these definitions follow existing tableau frameworks for propositional logic. The interested reader is referred to Smullyan [38] and Fitting [6]. The relevant definitions are given here for ease of reference.

Definition 4 *An exhaustive tableau for X is a tree, whose points are (occurrences) of formulas, which is constructed as follows.*

A formula X is a tableau for X.

A tableau Γ for X can be extended into another tableau Γ' for X by applying an expansion rule to any of the leaf nodes of Γ.

A tableau where no more expansion rules are applicable is a completed tableau*.*

A *branch of a tableau* Δ can be defined as the path from the root of the tree to one of the leaf nodes. Δ can also be interpreted as the set of atomic propositions found along that path.

A branch Δ of a tableau is *closed* if it contains a formula and its negation. A tableau is closed if all its branches are closed.

Now each branch of a tableau represents a state of the world in the same way that piles did. The only difference is that several branches can share the same propositions whereas piles were given separately by listing all the propositions in each one.

A closed branch represents an inconsistent state of the world.

A completed tableau for X constitutes an explicit representation of the logical semantics of X in the same way that

$$\begin{array}{ccc} A & \neg A & A \\ B & B & \neg B \end{array}$$

represented $A \vee B$.

An exhaustive tableau for X must be a completed tableau in order to be considered a reliable representation of the information contained in X. This is because as long as there are in the tableau compound propositions to which expansion rules have not been applied, there will be information about atomic propositions missing in some branches.

The operation of the tableau construction method is better shown over an example. Sentence

(74) *If John is married and he has children, then his children are at school*

can act as an example. Assuming a logical form for this sentence $(j \wedge c) \rightarrow a^c$, the tableau for this sentence would be:

$$\begin{array}{ccccccc}
\multicolumn{7}{c}{\boldsymbol{(j \wedge c) \rightarrow a^c}} \\ \hline
\multicolumn{3}{c}{\neg(j \wedge c)} & \multicolumn{3}{c}{\neg(j \wedge c)} & j \wedge c \\
\multicolumn{3}{c}{a^c} & \multicolumn{3}{c}{\neg a^c} & a^c \\ \hline
\neg j & \neg j & j & \neg j & \neg j & j & j \\
\neg c & c & \neg c & \neg c & c & \neg c & c
\end{array}$$

The original proposition is represented in bold typeface in order to distinguish it from the propositions that result from its expansion. This notational convention becomes specially useful later when representing discourses.

The criteria given above for the behaviour of signals over piles can be applied to the pile corresponding to the set of atomic propositions along a branch. To determine the presuppositions of a tableau (as a representation of a proposition), the tableau is interpreted as the set of such piles as determined by its branches. The same criteria described earlier apply to determine the presuppositions of the corresponding proposition. Over this example, the same behaviour as in examples (71), (72.a) and (72.b) seems to be in operation.

Existing tableau frameworks provide definitions of a tableau for a set of formulas. In order to formalize discourses appropriately, a definition must be provided for the concept of tableau for a sequence of formulas.

The definition can be given recursively:

> 1. If X is a formula, the tableau for X is a tableau representation of the discourse (X),
>
> 2. If Γ is a tableau representation for a discourse, the tableau representation for the discourse $\Gamma \circ (X)$ is the tableau that results from adding the formula X to all the open branches of Γ and then expanding the resulting tableau.

Intuitively, interpretation of discourses relates to interpretation of sentences in the sense that for a discourse of sentences $(P_1) \circ (P_2) \circ \ldots \circ (P_n)$, the interpretation of the discourse is equivalent to the interpretation of sentence P_n *in the context of* $(P_1) \circ (P_2) \circ \ldots \circ (P_{n-1})$. It has been argued above that the completed tableau for a proposition (or a sequence of propositions) can be taken as a representation of the logical information contained in them. This allows a straightforward definition of context.

The tableau for a discourse $(P_1) \circ (P_2) \circ \ldots \circ (P_n)$ constitutes the *context* for the interpretation of any sentences that follow it.

For example, the interpretation of the discourse

> (75) (*If Mary has had a bath, then there is no hot water left*) $\circ$ (*If Mary has had a bath, then Bill regrets that there is no hot water left*)

or $(m \rightarrow \neg w) \circ (m \rightarrow r^{\neg w})$ in terms of a tableau would take place as follows:

A) the sentence $m \rightarrow \neg w$ is expanded using the corresponding β- and $\neg$-rules:

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{m \rightarrow \neg w}} \\ \hline \neg m & \neg m & m \\ \neg w & \neg\neg w & \neg w \\ & w & \end{array}$$

The representation obtained constitutes the context in which the proposition $m \rightarrow r^{\neg w}$ is interpreted.

B) The sentence $m \rightarrow r^{\neg w}$ is added to the representation so far:

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{m \rightarrow \neg w}} \\ \hline \neg m & \neg m & m \\ \neg w & \neg\neg w & \neg w \\ & w & \\ \boldsymbol{m \rightarrow r^{\neg w}} & \boldsymbol{m \rightarrow r^{\neg w}} & \boldsymbol{m \rightarrow r^{\neg w}} \end{array}$$

C) The representation is expanded using a β-rule:

$$\begin{array}{ccccccccc} \multicolumn{9}{c}{\boldsymbol{m \rightarrow \neg w}} \\ \hline \multicolumn{3}{c}{\neg m} & \multicolumn{3}{c}{\neg m} & \multicolumn{3}{c}{m} \\ \multicolumn{3}{c}{\neg w} & \multicolumn{3}{c}{\neg\neg w} & \multicolumn{3}{c}{\neg w} \\ \multicolumn{3}{c}{} & \multicolumn{3}{c}{w} & \multicolumn{3}{c}{} \\ \multicolumn{3}{c}{\boldsymbol{m \rightarrow r^{\neg w}}} & \multicolumn{3}{c}{\boldsymbol{m \rightarrow r^{\neg w}}} & \multicolumn{3}{c}{\boldsymbol{m \rightarrow r^{\neg w}}} \\ \hline \neg m & \neg m & m & \neg m & \neg m & m & \neg m & \neg m & m \\ r^{\neg w} & \neg r^{\neg w} & \underline{r^{\neg w}} & r^{\neg w} & \neg r^{\neg w} & \underline{r^{\neg w}} & \underline{r^{\neg w}} & \underline{\neg r^{\neg w}} & r^{\neg w} \end{array}$$

When it comes to considering the behaviour of the presupposition $\neg w$ of the second sentence of the discourse, no additional manipulations of the information are required. The same criteria discussed for deciding whether a given pile signals the truth of a presupposition can be applied to each one of the branches of the resulting tableau, and they will take into account the information that was in the context. In this example it is clear that the presupposition $\neg w$ is going to behave differently in each one of the different branches. It is also possible to see that the presupposition is going to behave differently in the tableau that results for the discourse than it would have done in a tableau for the second sentence on its own. These properties are used in the next chapter to formalize the behaviour of presupposition with little additional apparatus.

I am dealing with a language that has both a logical component and a presuppositional component. A decision must be made as to whether the logical component on its own can be considered as a representation of the context, or whether the presuppositional component must be taken into account. One of the aims of this work, is to consider the presuppositional component as an integral part of the context on the same level as the logical component.

Intuitions show that signals carried by signalled information must also be taken into account if no potential information is to be lost.

The following example is based on the sentences:

(76.a) *John has children.* (c)

(76.b) *John's children have forgotten Bill.* (f^c)

(76.c) *Bill regrets that John's children have forgotten him.* (g^{f^c})

Take the simplest form of sentence with higher order signals: g^{f^c}.

The tableau for sentence g^{f^c} would be:

$$\boldsymbol{g^{f^c}}$$

According to the criteria as given, f and c stand as signals of the tableau.

The next step up in complexity of the representation would be sentences involving conjunctions, such as:

(77) *John's children have forgotten Bill and Bill regrets that John's children have forgotten him* ($f^c \wedge g^{f^c}$).

$$\begin{array}{c} \boldsymbol{f^c \wedge g^{f^c}} \\ \hline f^c \\ g^{f^c} \end{array}$$

In this case the criteria predict that both c and f are signals of the proposition, but they also predict that f is only added as redundant information. It seems that c is signalled not only as a higher order signal of the second conjunct g, but also as a first order signal of the first conjunct f.

The case where the higher order signal appears as first conjunct would be:

(78) *John has children and Bill regrets that John's children have forgotten him* ($c \wedge g^{f^c}$).

$$\frac{\boldsymbol{c \wedge g^{f^c}}}{\begin{array}{c} c \\ g^{f^c} \end{array}}$$

In this case, f and c are signalled, but in this case it is c that can only act as redundant information.

3.3.2 Coverage Property

The definition of tableau expansion rules ensures that these tableaux obey a special property.

> *Coverage Property:*
>
> If one branch of a tableau holds a sentence X, then every (open) branch of that tableau will hold either X or $\neg X$.

This property can be seen to hold: 1) it applies to each one of the expansion rules, 2) the procedure for adding sentences to a tableau is defined in terms of adding the new sentence to every (open) branch (and then applying the expansion rules to it).

A consequence of this property in terms of the semantics, is that each branch of the tableau contains a truth-value assignment for all the atomic propositions in the original sentence. Each one of these atomic truth value assignments makes the sentence true. The structures that result are equivalent to classical truth tables. A tableau formulation is retained in spite of this fact because it takes into account that lines of the truth table that become inconsistent as more information is added are dropped out of the reckoning (as an effect of branch closure). The tableau framework also provides the idea of a branch as an abstract entity with certain conceptual significance to which information can be added individually (by means of expansion rules). It can be seen from the examples studied so far how this concept may be of use in modelling the behaviour of presupposition. This will constitute the basis of chapter 4.

3.3.3 The Formalism as a Decision Method for the Logic

Let the tableaux constructed using the expansion rules given above be called *exhaustive tableaux* or ET tableaux. Exhaustive tableaux (ET) have been designed as a representation of logical sentences that makes explicit all the information that plays a role in determining the behaviour of presuppositions. Exhaustive tableaux differ from traditional tableaux only in the definition of β- expansion rules. The traditional definition of β- expansion rules is not suitable for presupposition because it specifies the alternatives in the minimal form that preserves soundness and completeness of the logical calculus. This is done in order to simplify the computation of logical consequences. The information that is not specified explicitly as a result of this policy does not affect logical consequences, but it can be, as shown above, relevant to presupposition behaviour. In the traditional formulation of β-rules, some of the interactions between presuppositional and non-presuppositional information may be obscured. However, because exhaustive tableaux are constructed by using the same method as analytic tableaux, they inherit from this formalism its properties as a decision method for the logic being considered.

It can be shown that the logical calculus determined by exhaustive tableaux is equivalent to the logical calculus determined by analytic tableaux (simple tableaux or ST). This is formulated in the following theorem.

Theorem 1 *ET tableau are equivalent to ST tableaux.*

Proof: Given that the definitions of tableau and closure are the same in the ET framework as in ST tableaux, and that $\neg$-rules and α-rules are the same in both cases, it is enough to show that under the circumstances under which ST tableau for a β-formula becomes closed, the ET tableau for the same β-formula also becomes closed (and vice versa).

Here are the corresponding tableaux for some β-formulas:

$$\text{ET tableau} \quad \frac{P_1 \vee P_2}{\begin{array}{ccc} P_1 & \neg P_1 & P_1 \\ P_2 & P_2 & \neg P_2 \end{array}} \qquad \text{ST tableau} \quad \frac{P_1 \vee P_2}{P_1 \quad P_2}$$

To show that ST tableau closure for a β-formula implies ET tableau closure it is enough to say that an ST β-rule expansion would close only in case P_1 and P_2 are false. Under these circumstances, the ET framework expansion rule would also close.

To show that ET tableau closure for a β-formula implies ST tableau closure, it pays to formulate the problem in a different way. What is required is to show that there is no way of closing the expansion of a β-rule in the ET framework that would not close the ST expansion. This is equivalent to showing that the possible truth value assignments that leave open the ST expansion also leave open the ET expansion. These cases are $P_1 = true$ and $P_2 = true$, $P_1 = false$ and $P_2 = true$, and $P_1 = true$ and $P_2 = false$. Because these correspond exactly to the three possible assignments given as valid in the ET expansion, each one of them leaves open at least one branch of the expansion.

QED

Having established the relationship, the usual concepts apply.

A *tableau refutation* of X is a closed tableau for X.

A *tableau proof* of X is a closed tableau for $\neg X$.

A proposition X is *logically valid* iff there is a tableau proof of X.

A proposition X is *logically inconsistent* iff there is a tableau refutation of X.

A proposition X is a *logical consequence* of a discourse Γ ($\Gamma \vdash X$) iff adding $\neg X$ to the tableau for Γ results in a closed tableau.

3.4 A Summary of the Behaviour of Presupposition over ET Tableaux

The informal observations and criteria presented so far are summarised in this section in the form of rules for determining the behaviour of presuppositions over tableaux. These are not intended as an appropriate method of interpreting presupposition. They are given here to bring together all the intuitions presented in this chapter that are going to be used in the subsequent chapters.

3.4.1 Rules for Presupposition

Over this representation of the logical language, the behaviour of presupposition presented can be described in terms of the behaviour of presupposition in each individual branch.

Two issues need to be dealt with: 1) how the presuppositions of a branch are ascertained from the presuppositions of the atomic propositions in it, and 2) how the presuppositions of branches of the same tableau interact.

I notate a branch as Δ. I use $B \in \Delta$ as shorthand for 'the proposition B appears in the branch Δ'.

The notation for presupposition is extended to branches so that Δ^B stands for 'the branch Δ signals the truth of B in the states of the world that it represents'.

The notation for presupposition is extended to tableaux so that Γ^B stands for 'all the branches Δ_i of Γ signal the truth of B in the states of the world that they represent'. Since these should correspond to all the states of the world that the tableau represents, the tableau can be said to signal the truth of B.

The compositionality rules for presupposition do not take into account presuppositional information from branches of a tableau that are closed.

Rule 1

For an open branch Δ such that $A^B \in \Delta$, Δ^B unless: i)$B \in \Delta$, or ii) A^B is of the form $\neg D^B$ for some atomic proposition D, and either there is some $\neg B \in \Delta$, or there is some $C^{\neg B} \in \Delta$, for C atomic.

Rule 2

For a tableau Γ with valid open branches $\Delta_1, \ldots \Delta_n$, Γ^B if Δ_i^B for all i (by rule 1).

3.4.2 Traditional Presuppositional Concepts in Terms of Tableaux

The tableaux framework allows definition of some of the traditional concepts that surround presupposition (as described in chapter 1).

The presupposition B of a presuppositional sentence A^B added to a tableau Γ is *satisfied* if $\Gamma \vdash B$ (the tableau $\Gamma \cup \{\neg B\}$ is closed).

The presupposition B of a presuppositional sentence A^B added to a tableau Γ is *cancelled* if $\Gamma \vdash \neg B$ (the tableau $\Gamma \cup \{B\}$ is closed).

These two definitions correspond to the intuitive concepts of satisfaction and cancellation. The rules for the behaviour of presupposition over tableaux can be taken to capture the satisfaction of presuppositions in as much as, through condition 1.i, they predict no presuppositions whenever the presupposition is already present in the context (either locally within a given branch or overall in a more general context in cases of discourses). Condition 1.ii captures the cases of cancellation. However, it is clear that there will be cases when some branches of a tableau are closed by B and some by $\neg B$. These hybrid cases between satisfaction and cancellation escape the simpler analysis and gave rise to the need for complex projection rules in previous analysis of presupposition behaviour. In their simplest manifestation, hybrid cases occur as the traditional cases of problematic projection. These involve sentences (71), (72.a) and (72.b) given above. More complex manifestations concern discourses where the effect of context plays a role in the interpretation of presupposition. The discourse of example (75) is an instance of these cases. In all these examples it holds that for any of the tableau representations some branches of the tableau are closed by the presupposition involved and some by its negation. Under those circumstances, the traditional definitions of satisfaction and cancellation could not account for the resulting presuppositional behaviour.

The present framework achieves this by allowing presupposition to be blocked locally by either B or $\neg B$ (conditions 1.i and 1.ii).

3.5 Conclusions

At the beginning of the present chapter, the property of compositionality for truth values was described. Two different ways of understanding it were given: in one, the truth value of a compound is worked out from the truth values of the components, in the other, the truth value of the compound determines the truth values of the components.

For the case of presupposition only the first form of compositionality can be said to operate. The rules presented in this chapter allow the determination of the presuppositions of a compound in terms of the presuppositions of its components. However, given the presuppositions of a compound, the presuppositions of its components cannot be fully determined. The examples given show that there are cases when a compound does not have presuppositions and yet its components do. Furthermore, there are cases when not only the components have presuppositions that are not inherited by the compound but also some of the semantic alternatives that constitute the informative content of the compound have presuppositions that do not necessarily become presuppositions of the compound.

The compositionality of the presuppositions of compounds constructed with logical connectives has been formulated in two steps: first the compositionality of the presuppositions of a logical alternative (represented as a branch of a tableau) in terms of the propositions in them and their presuppositions, and second the compositionality of the presuppositions of a compound (represented as a tableau) in terms of the presuppositions of the different logical alternatives that it represents (its branches). The main contribution of this chapter is to show that compositionality for presuppositions of compound logical sentences is best described in two stages: first how the different literals in a semantic alternative determine the presuppositions that are operative in that semantic alternative, and second how the presupposition that are operative in all the valid semantic alternatives determine the presuppositions of the compound.

There can be serious objections to the claim that the proposed representation is compositional over the connectives as they have been listed. The expansion rules for propositions of the form $P_1 \wedge P_2$, $P_1 \vee P_2$, and $P_1 \rightarrow P_2$ are indeed compositional, but the rules dealing with $\neg\neg P$, $\neg(P_1 \wedge P_2)$, $\neg(P_1 \vee P_2)$, and $\neg(P_1 \rightarrow P_2)$ are not compositional in the sense that a compact definition is given for the combinations of negation with any other connectives, instead of defining the meaning of the combination as a function of the meaning for the original compound (for instance, show the meaning of $\neg(P_1 \wedge P_2)$ as a function of the meaning of $P_1 \wedge P_2$).

This problem is related with the choice of representation for the negation of complex propositions.

I have made the assumption that classical negation operates over the language wherever reasoning about negated sentences is required. This ensures that, potentially, reasoning can take place in this logical framework starting from sentences of any syntactic form. However the particular nature of the cases to be modelled (elementary natural language sentences) in actual fact restricts the syntactic forms that have to be processed to some specific configurations where negation is concerned.

The behaviour of presupposition with respect to negation that has to be modelled is found in natural language discourses. Assertion, and communication in general, seem to be restricted to sentences where negation appears only with narrow scopes. The forms of negation that are applied in these discourses to simple sentences become intuitively awkward when applied in the same way to more complex logical compounds. A sentence like

(79)*It is not the case that if John likes Mary then Mary is happy*

is logically correct, but it is not common in language (other than among those familiar with formal logic). Beaver [1] addresses this question with respect to the use of negation as a test for presupposition. Mercer [27] claims that 'too' behaves differently from other presuppositional constructions because there is no obvious negation of sentences constructed with it. As a result of these peculiarities of the way natural language behaves, there are no intuitive sources of how presupposition should behave when embedded in compounds that are themselves under the scope of negation. However, the procedure for logical interpretation would not be complete unless the negation of any sentence of the language can be given a meaning.

In the present framework I consider sentences of the form *It is not the case that ...* as metalevel operators to represent the denial of a previous sentence. As such, they ought to be considered together with belief revision operations (that have been deliberately left out of the present work). When such operators are considered, some form of compositionality would be required. The only negated compounds that appear in the present work are those that result from the expansion of complex sentences. For these, I have considered it advisable to include additional expansion rules that are not compositional with respect to the connectives.

The expansion rules used in the present framework are chosen to ensure that all the different valid alternatives implied by a sentence are listed explicitly in a tableau for that sentence, and all atomic formulas involved appear (either negated or not) in every branch (Coverage Property). These constraints on the semantics allow all the predictions of the framework for first order presuppositions to be explained in very clear terms that will make it easy to consider a formal representation in chapter 4.

A simpler formulation of the general interaction between assertions, first order presuppositions and higher order presuppositions may be obtained within a framework that allows simultaneous representations of the nested structure of presuppositions as well as the logical structure of assertions. In order for the intuitions above to be captured more directly in the representation, it would be necessary to develop a representation where not only presuppositions and higher order presuppositions are treated in a similar way, but also presuppositions and assertions are treated in a similar way. The development of some such a framework is undertaken in chapter 4.

The rules presented in this chapter provide a strong basis on which to build a more flexible proof theory that accounts for all these shortcomings. This is attempted in the following chapter. Because the issue of defeasibility is closely related to consistency checking, the framework proposed here presents advantages for this purpose by having the semantics of each connective made fully explicit.

Chapter 4

Tableau Expansion Rules for Presuppositions

4.1 The Approach

4.1.1 The Argument for Presupposition as an Inference

Certain constructions in natural language show uses of the conditional as means of encoding that a presupposition must necessarily follow from a sentence that presupposes it.

> (80) *If the King of France is bald, then there is a King of France.*
>
> (10.a) *If all Bill's friends have encouraged him, he must have friends. (Gazdar)*

This interpretation is further supported by the fact that sentences

> (81) **There is no King of France. The King of France came to the party.*

seem to be mutually incompatible, just as they would be if the presupposition of the second sentence (there is a King of France) were also an entailment.

Behaviour discussed for A^B suggests that A^B entails B. This presents a representation problem in as much as it does not allow an extension of the same solution to explain the relationship between $\neg A^B$ and B. In classical logic, A and $\neg A$ do not share the same entailments. Furthermore, $\neg A^B$ is compatible both with B and with $\neg B$, so there seems to be no special link of entailment between $\neg A^B$ and B. If the entailment solution is adopted for the relationship between A^B and B, the relationship between $\neg A^B$ and B has to be explained in a different way. This is the approach favoured by Mercer [27]. Mercer suggests that B should be considered as a default inference from $\neg A^B$, to be blocked if $\neg B$ is in the context. This characterization is essentially correct.

However, it has the disadvantage of dissociating the conceptual nature of presuppositions in positive environments from that of presuppositions in negative environments. Intuition suggests that a common concept of presupposition for both cases should be preferred, even if negation introduces differences of behaviour between them. Both types of presupposition originate from the same linguistic elements. The general tendency towards compositionality suggests that they both be explained in terms of the same underlying mechanism.

On the other hand, there are also arguments for retaining a distinction between traditional entailments and presuppositions of positive sentences.

As argued by Sandt [47], presuppositions share some of the characteristics of anaphora in that they can constitute indications to search for a referent fulfilling certain conditions earlier in the discourse. Traditional entailments do not have this property.

This chapter presents a formalization of presupposition as inference that addresses the differences in behaviour between the presuppositions of positive and negative sentences, and the anaphoric ingredient involved in presupposition satisfaction.

The operation of the tableau representation can be used as a framework to study defeasibility. In the tableau framework, the basic operation on the representation is an expansion rule. Each expansion rule operates on a single branch. This means that defeasibility would also have to be treated at branch level. Conjunctions give rise to a single branch with the conjuncts in it. Conditionals and disjunctions give rise to three branches, each one with different literals built from atomic subformulas of the original proposition.

This results in an additional type of conflict between negated presuppositions and negated presuppositional sentences: the one that arises within a branch as a result of the expansion of a conditional or a disjunction. This was already apparent in examples (71), (72.a) and (72.b).

4.1.2 Using Tableaux to Represent Presupposition as well as Assertion

There are two different issues at play here:

- to include, in an appropriate way, presupposition as part of the information content of an utterance (or a context)
- to retain the distinction between information that is presuppositional in origin and the rest of the information of an utterance (or a context)

First one must show how one unified explicit representation for the logical and the presuppositional contents of the language L can be given. The construction of this unified representation also operates as a proof theory based on tableau methodology. The logic that results from interpreting the resulting representation as a calculus constitutes a logic of the total information contained in the propositions of the language L, designed to include not only the logical consequences of any asserted propositions, but also to integrate their presuppositions and any logical consequences of bringing both ingredients together[1]. Section 4.2 presents such a unified representation, and studies some of its basic properties and how they relate to the intuitions behind the corresponding linguistic examples.

On the other hand, it is important not to lose sight of which information has been asserted and which information has been presupposed. Section 4.3 considers this issue, studying how much of the information in the given representation can be classed as presupposition in the traditional sense of the word, and how much must be considered as logical inferences from the context, or logical inferences from the context extended with the actual presuppositions.

[1]The formal properties of this representation when it is considered as a logical calculus are considered in chapter 5.

4.2 The Unified Assertion–Presupposition Representation

4.2.1 Presuppositional Expansion Rules

Semantic tableaux are originally intended as a decision method for propositions of classical logic. It would be interesting if they could be extended so as to become a decision method for presuppositions as well. This can be achieved in the following way. Tableaux closure defines a decision method for propositions of the logic (representation language). If extra rules are provided to include presuppositional information in the branches, a new concept of closure results. This new concept of closure defines a new decision method. The presuppositional information of a tableau will be that information that is validated by the second decision method but not by the first. This is the approach followed here.

Presuppositional information is made explicit in the present formalism by additional expansion rules for presuppositions. It must be kept in mind that presuppositional expansion rules only apply to atomic propositions, because the relation of presupposition is only defined as a primitive for atomic propositions.

The behaviour described above suggests that presuppositions may be modelled as some kind of logical consequences of their presuppositional sentences. This is similar to associating with each atomic sentence of the form A^B a semantics related to that of $A \rightarrow B$. If we want to be coherent with the semantics of our connectives, this would result in an expansion rule of the following form:

$$\frac{A^B}{\begin{array}{ccc} \neg A & \neg A & A \\ B & \neg B & B \end{array}}$$

It must be borne in mind that A^B is only a notational device to indicate the fact that A presupposes B. For all logical purposes other than presupposition, A^B is equivalent to A. Under this light the first two branches of the expansion above are always closed. Therefore the required expansion rule can be reduced to:

$$\sigma\text{-rule}) \qquad \frac{\sigma^\pi}{\pi}$$

The behaviour in the case of negative presuppositional sentences corresponds to allowing the inference of the presupposition to be blocked under certain circumstances. This can be captured as an expansion rule by the addition of a constraint to be fulfilled by the corresponding branch for the rule to be applicable. The best solution is obtained by expanding first without constraints (but marking those formulas that originate from expansion of negative presuppositional sentences) and then eliminating those marked formulas that stand in closed branches (this should turn the branches into open ones). The constraint on expansion corresponding to this procedure can be formulated as follows.

ξ-rule) $$\frac{\neg\xi^{\pi}}{\pi}$$

unless the expansion of π closes the branch

The intuitions behind this formulation of the rule may be easier to understand with some examples.

Assume it is known in the context that

(82) *John does not have children* ($\neg c$).

This can be represented by assuming that all interpretation takes place as extension of a tableau of the form:

$$\begin{array}{c} \neg\boldsymbol{c} \\ \vdots \end{array}$$

Now processing of

(83) *John's children have not forgotten Bill* ($\neg f^c$)

will result in the tableau:

$$\begin{array}{c} \neg\boldsymbol{c} \\ \vdots \\ \neg\boldsymbol{f^c} \end{array}$$

The potential application of the ξ-rule has not taken place because c would close the branch.

Once the presuppositions are accepted into the logic, higher order presuppositions play a role. In a presuppositional tableau the presuppositions of a sentence are expanded irrespective of whether the sentence was a presupposition itself or not. If presupposition have other presuppositions themselves, these presuppositions are expanded as well. The problem in this case is not in expanding higher order presuppositions themselves, but rather in constraining the expansion of presuppositions that have presuppositions, so that they are blocked whenever their higher order presuppositions conflict with the context. This is done by extending the constraint on expansion of negative sentence to the whole subtree instead of limiting it to the presupposition itself.

The need for this extension becomes apparent in the interpretation of

> (84) *Bill does not regret that John's children have forgotten Bill* $(\neg g^{f^c})$

in the same context:

$$\begin{array}{c} \boldsymbol{\neg c} \\ \vdots \\ \boldsymbol{\neg g^{f^c}} \end{array}$$

The proposition $\neg g$ is not expanded because the expansion of f would close the branch.

4.2.2 UAP Tableaux Using Presuppositional Expansion Rules

Given a sentence X, a *UAP tableau* (Unified Assertion-Presupposition tableau) is obtained by applying the rules in the following manner:

- expand the tableau for X using all the tableau rules ($\neg$-rules, α-rules, β-rules, σ-rules, or ξ-rules); mark all additions resulting from a ξ-rule; if a formula is marked, mark its expansion as well
- once no more rules are applicable, retract those marked additions that contribute to the closure of a branch (either by themselves or through their expansion)

- eliminate all remaining markings[2]

The process of construction can be optimised by applying the rules in the following order. At any stage in the process of constructing the tableau: a) if any $\neg$-rule, α-rule, β-rule or σ-rule is applicable, apply that rule; otherwise b) if any ξ-rule is applicable, apply that rule. This order of application is determined by the fact that from the moment that a ξ-rule is applied any further expansion is tentative. The proposed order of application minimises the need to backtrack during systematic application of the rules.

This approach would also favour early closure of branches, because all the rules that can close a branch are given priority over those that can be blocked. This would lead to a more efficient method.

The definition of tableau for a discourse given in chapter 3 still applies.

A proposition P is *UAP valid* iff the UAP tableau for $\neg P$ is closed.

A proposition P is *UAP inconsistent* iff the UAP tableau for P is closed.

A consequence relation can be defined over these tableaux. Two observations are in order at this point. The first observation concerns the relation between this consequence relation and presupposition. The consequence relation so obtained would not correspond to the relation of presupposition. Instead, it would encompass both logical entailment and presupposition as a single consequence relation. The second observation concerns the role of this consequence relation when the tableaux are being used in the interpretation of propositions (considered as utterances). This consequence relation plays no role in the interpretation of a proposition into a tableau. The tableau rules given are sufficient for that purpose. The consequence relation is simply a means of accessing or querying the tableau for the information that is contained in it. This is an unorthodox use of tableaux, as a representational device as well as a decision method. If the consequence relation is to be used in this way, it is important that the tableau be fully expanded before it is queried in any way. This can be formalised adequately by defining the consequence relation in terms of tableaux for discourses. However, it is

[2]Without this step, the defeasibility of presuppositions would also operate long range over discourses. As has been discussed at the beginning of the chapter, observed behaviour seems to depend on the length of the range. Ideally, a parameter such as 'number of consecutive sentences after which a defeasible presupposition stops being defeasible' ought to be defined. However, in practice this parameter seems to be variable as well. The present system is developed under the assumption that this parameter has a value of 0.

important to note that those additions to a tableau that occur during the querying process should not be taken as information updates, and must be retracted once the querying process is over if the tableau is to be used for representation purposes.

A proposition X is a *UAP consequence* of a discourse Γ ($\Gamma \vdash_{UAP} X$) iff the tableau for $\Gamma \circ (\neg X)$ is a closed UAP tableau.

For instance, assume you want to test whether $(b^t \to h) \vdash_{UAP} t$. The first step of the construction of the UAP tableau for this formula results in:

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{b^t \to h}} \\ \hline \neg b^t & \neg b^t & b^t \\ h & \neg h & h \\ (t) & (t) & t \end{array}$$

where the propositions in brackets are the ones marked as being defeasible. In this case, the second step is vacuous, because none of the marked formulas closes a branch. The third and final step produces the UAP tableau for $(b^t \to h)$:

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{b^t \to h}} \\ \hline \neg b^t & \neg b^t & b^t \\ h & \neg h & h \\ t & t & t \end{array}$$

By definition, $(b^t \to h) \vdash_{UAP} t$ iff the tableau for $(b^t \to h) \circ (\neg t)$ is closed. This tableau would be:

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{b^t \to h}} \\ \hline \neg b^t & \neg b^t & b^t \\ h & \neg h & h \\ t & t & t \\ \underline{\neg t} & \underline{\neg t} & \underline{\neg t} \end{array}$$

So $(b^t \to h) \vdash_{UAP} t$. However, the tableau that can be used as a representation of the information contained in $(b^t \to h)$ is the first one and not the second one.

4.2.3 Defeasibility

The sentence

(71) *If there is a typewriter then the typewriter is blue*

provides an example of how the mechanism of retracting problematic expansions of ξ -rules operates.

The first step of the construction produces:

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{t \rightarrow b^t}} \\ \hline \neg t & \neg t & t \\ b^t & \neg b^t & b^t \\ \underline{t} & \underline{(t)} & t \end{array}$$

and the second and third steps lead to a final representation:

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{t \rightarrow b^t}} \\ \hline \neg t & \neg t & t \\ b^t & \neg b^t & b^t \\ \underline{t} & & t \end{array}$$

In this case, the presuppositional sentence in the middle column is negated, and the branch holds $\neg t$, so no expansion takes place. As a result, adding $\neg t$ to the resulting tableau does not close it. The presupposition t is not a UAP consequence of the sentence. It is also apparent that one of the alternatives has become closed as a result of expanding the presupposition. The tableau now holds only those alternatives that make intuitive sense.

The sentence

(85) *If John's children have forgotten Bill, Bill does not regret it*

has the form $f^c \rightarrow \neg g^{f^c}$. In this example one can see how expansions of ξ-rules may remain in the representation after the two final steps of construction of the tableau.

The construction of the UAP tableau produces first:

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{f^c \rightarrow \neg g^{f^c}}} \\
\hline
\neg f^c & \neg f^c & f^c \\
\neg g^{f^c} & g^{f^c} & \neg g^{f^c} \\
c & c & c \\
\underline{(f^c)} & \underline{f^c} & (f^c) \\
 & & (c)
\end{array}
$$

which then reduces to:

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{f^c \rightarrow \neg g^{f^c}}} \\
\hline
\neg f^c & \neg f^c & f^c \\
\neg g^{f^c} & g^{f^c} & \neg g^{f^c} \\
c & c & c \\
 & \underline{f^c} & f^c \\
 & & c
\end{array}
$$

The tableau is reduced to two branches that contains the literals $c, \neg f, \neg g$ and $c, f, \neg g$. (The possibility of g being true has been lost through closure). In the last branch, the expansion of ξ-rule has remained.

The sentence

(73) *Either Bill has started smoking or Bill has stopped smoking.*

constitutes an example where the proposition that is eliminated in the second step of construction is not the latest one to be added, but rather an earlier expansion of a ξ -rule. The representation after the first step of construction:

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{e^{\neg s} \vee p^s}} \\
\hline
e^{\neg s} & e^{\neg s} & \neg e^{\neg s} \\
p^s & \neg p^s & p^s \\
\neg s & \neg s & (\neg s) \\
\underline{s} & \underline{(s)} & \underline{s}
\end{array}
$$

reduces finally to:

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{e^{\neg s} \vee p^s}} \\
\hline
e^{\neg s} & e^{\neg s} & \neg e^{\neg s} \\
p^s & \neg p^s & p^s \\
\neg s & \neg s & \\
\underline{s} & & s
\end{array}
$$

For the sake of clarity, I have left a gap in the tree wherever the presupposition of a negative sentence is blocked; these gaps can play a role in understanding where each of the propositions in the tableau comes from, but they do not have any logical meaning.

The order imposed on the application of expansion rules for positive and negative sentence gives in this case a very interesting result. The presuppositional information of the branches still does not become presuppositional information of the sentence. Neither adding s on its own nor adding $\neg s$ on its own closes the tableau, so none can be said to be a consequence of the sentence. However, the representation obtained by the use of the new expansion rules captures some additional information that the original representation could not capture: in which of the branches s holds and in which of the branches $\neg s$ holds. Sentence (73) is therefore interpreted as

> (86) *Either Bill did not smoke and Bill has started smoking or Bill did smoke and Bill has stopped smoking.*

This is intuitively correct.

4.2.4 Defeasibility and Violations of the Coverage Property

The sentence

> (87) *If John has children, then Bill does not regret that they have forgotten him*

has the form $c \rightarrow \neg g^{f^c}$.

An initial UAP tableau for this expression:

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{c \rightarrow \neg g^{f^c}}} \\ \hline \neg c & \neg c & c \\ \neg g^{f^c} & g^{f^c} & \neg g^{f^c} \\ (f^c) & f^c & (f^c) \\ \underline{(c)} & \underline{c} & (c) \end{array}$$

reduces to:

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{c \rightarrow \neg g^{f^c}}} \\
\hline
\neg c & \neg c & c \\
\neg g^{f^c} & g^{f^c} & \neg g^{f^c} \\
 & f^c & f^c \\
 & \underline{c} & c
\end{array}
$$

The application of the rules to this example results in a tableau that does not fulfill the Coverage Property. The framework allows adequate analysis of this problem. In terms of the underlying semantics, this implies that the tableau can no longer be represented as a set of actual states of affairs over the set of atomic propositions that appear in it. One of the open branches of this tableau can be thought of as an actual state of affairs over the atomic propositions that the tableau contains, but the other branch only fixes truth values for some of the atomic propositions in the tableau (c, g) but says nothing about f. For the correct interpretation of these tableaux as actual states of affairs it must be understood that whenever the Coverage Property fails, the proposition that is missing in one branch should be interpreted as false in that branch. This corresponds to an inference of the form $\neg c \vdash_{UAP} \neg f^c$ which can be shown to be a valid inference. By means of this rule, the tableau above can be interpreted as corresponding to the following truth value assignments to the atomic subformulas of the proposition $\{(\neg c, \neg f, \neg g), (c, f, \neg g)\}$. It would be interesting to represent this rule as a tableau expansion rule. However, the fact that $\neg c$ does not determine the proposition f in $\neg f^c$ makes it impossible to provide a general rule.

In spite of this, the framework does capture the inference as required. This can be seen in example:

> (88) *If John has no children, then John's children didn't come to the party.*

Such a conditional has the form $\neg c \rightarrow \neg p^c$.

The validity of this sentence could be tested by constructing the tableau for the negation of the sentence $\neg(\neg c \rightarrow \neg p^c)$.

$$
\begin{array}{c}
\boldsymbol{\neg(\neg c \rightarrow \neg p^c)} \\
\hline
\neg c \\
p^c \\
\underline{c}
\end{array}
$$

This tableau is closed. So the sentence $\neg c \to \neg p^c$ is a UAP tautology.

However, even though a general rule is out of the question, it is possible to complement the specification of the method of application of the rules with an additional procedure to supplement the information vacuum that results from the retraction of expansions of ξ-rules that close branches. Such a procedure would add the complement of the offending proposition. This safeguards the Coverage Property. This supplementary procedure may require further application of the construction method to expand the propositions that result from it. The supplementary procedure would run as follows.

Given a sentence X, and a *UAP tableau* (Unified Assertion-Presupposition tableau) as obtained by applying the rules in the manner described above:

- for each negated presuppositional sentence whose expansion has been retracted, add to the corresponding branch the complement of the presupposition that gave rise to the expansion and
- expand this presupposition according to the method.

Such a procedure can be exemplified over example (87) given above.

The UAP tableau is the one given above:

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{c \to \neg g^{f^c}}} \\
\hline
\neg c & \neg c & c \\
\neg g^{f^c} & g^{f^c} & \neg g^{f^c} \\
 & f^c & f^c \\
 & \underline{c} & c
\end{array}
$$

Then the proposition $\neg f^c$ is added and expanded (tentatively, because it is now a negated proposition).

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{c \to \neg g^{f^c}}} \\
\hline
\neg c & \neg c & c \\
\neg g^{f^c} & g^{f^c} & \neg g^{f^c} \\
\neg f^c & f^c & f^c \\
\underline{(c)} & \underline{c} & c
\end{array}
$$

the usual steps of retraction take place and give rise to the representation:

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{c \rightarrow \neg g^{f^c}}} \\
\hline
\neg c & \neg c & c \\
\neg g^{f^c} & g^{f^c} & \neg g^{f^c} \\
\neg f^c & f^c & f^c \\
 & \underline{c} & c
\end{array}
$$

which represents correctly the intuitions that underlie the example, and satisfies the Coverage Property. This addition to the UAP construction method ensures that complete representations satisfying the Coverage Property can always be obtained. However, such procedure is required very rarely.

From this point on, only the final representation for a UAP tableau is given, and the corresponding steps of expanding tentatively, marking, retracting, and eliminating the markings are ommited.

4.2.5 Explanations

A discourse $(P_1) \circ \ldots \circ (P_n)$ can be interpreted as involving an explanation if $(P_1) \circ \ldots \circ (P_{n-2}) \circ (P_n) \vdash_{UAP} P_{n-1}$.

This interpretation of a discourse arises from a human tendency to 'make sense' of what is being said.

Explicit Explanations

The set of examples given in chapter 1 for cases where defeasibility of presupposition is related with sentences in which it is made explicit that one of the propositions involved is intended as an explanation of the other. They are cases like:

> (89) *There is no King of France. Therefore the King of France isn't in hiding.*

These sentences have the form $(\neg B) \circ (\neg A^B)$. The tableau for this discourse is not very informative.

If 'therefore' is given a logical interpretation, then the discourse is equivalent to the logical statement $\neg B \vdash_{UAP} \neg A^B$. According to this interpretation, the second sentence is being put forward as some sort of consequence of the first one.

In that case, what needs to be tested is whether the UAP tableau for $\neg B$ with the addition of $\neg(\neg A^B)$ is closed.

$$\begin{array}{c} \boldsymbol{\neg B} \\ \neg\neg A^B \\ A^B \\ \underline{B} \end{array}$$

Implicit Explanations

There is an interesting set of examples involving discourses with the negation of a presuppositional sentence as first sentence. In chapter 1 it was argued that because of the defeasible nature of the presuppositions of these type of sentences (as preferences between possible interpretations), the concepts of inconsistency and redundancy in these cases are not as clear cut as in other cases.

First of all, there are examples like

> (90) (*The King of France is not bald*) $\circ$ (*There is no King of France*)

of the form $(\neg A^B) \circ (\neg B)$. These have a UAP tableau representation:

$$\begin{array}{c} \boldsymbol{\neg A^B} \\ B \\ \underline{\boldsymbol{\neg B}} \end{array}$$

The tableau interpretation classifies these as inconsistent. However, it is also possible to interpret them as inverted explanations, in as much as the first sentence can be interpreted as a UAP consequence of the second sentence $\neg B \vdash_{UAP} \neg A^B$. This ambiguity is due to the fact that the discourse in fact constitutes a refutation of the preferred interpretation of its first sentence. It is therefore an invitation to retract those propositions that are licensed only by the preferred interpretation.

4.2.6 Acceptability

Conjunctions

An additional application of UAP consequence is that intuitive validity of some sentences seems to depend on the sentences being UAP valid rather than logically valid.

This is the case with conjunctions of the form:

> (19.a) **There is no King of France and the King of France plays golf.*

This sentence corresponds to the logical form $\neg B \wedge A^B$. The UAP tableau for this logical form would be:

$$\begin{array}{c} \underline{\boldsymbol{\neg B \wedge A^B}} \\ \neg B \\ A^B \\ \underline{B} \end{array}$$

This tableau is closed. This sentence is UAP inconsistent.

Conditionals and Disjunctions

In the case of conditionals and disjunctions there were unacceptable sentences in a pragmatic sense. The introduction of expansion rules for presupposition provides some means for understanding intuitively what is wrong about them.

Take an unacceptable sentence of this type like:

> (13) **If there is no King of France, then the King of France plays golf.*

This sentence has the form $\neg B \rightarrow A^B$. From its logical form it is not obvious what the problem is with the sentence. With the new expansion rule it becomes clearer. The full UAP tableau for the sentence would be:

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{\neg B \rightarrow A^B}} \\ \hline B & B & \neg B \\ A^B & \neg A^B & A^B \\ B & B & \underline{B} \end{array}$$

The last branch of the tableau is closed. Because A^B is false in one of the remaining alternatives, and true in the other, in terms of information the resulting tableau is equivalent to stating B. The sentence conveys the same information as a statement of B, and it is much more complex. This is already indicating that there is something seriously wrong with such a sentence.

This proposition is also predicted to signal the truth of B, which seems counterintuitive.

However, the problem is more serious than this. The problem with such a conditional statement is that it is not compatible with its antecedent being true. Take the discourse

> (91) *There is no King of France. *If there is no King of France, then the King of France plays golf.*

In terms of tableaux, this would correspond to the following representation:

$$\begin{array}{ccc} & \boldsymbol{\neg B} & \\ & \boldsymbol{\neg B \to A^B} & \\ \hline B & B & \neg B \\ A^B & \neg A^B & A^B \\ \underline{B} & \underline{B} & \underline{B} \end{array}$$

which is a closed tableau.

The same analysis applies to sentences

> (16.a) **Either there is a King of France or the King of France plays golf*

In terms of tableaux, this would correspond to the following representation:

$$\begin{array}{ccc} & \boldsymbol{B \vee A^B} & \\ \hline B & \neg B & B \\ A^B & A^B & \neg A^B \\ B & \underline{B} & B \end{array}$$

This one is only compatible with B being true. It is predicted to signal B, whereas this is counterintuitive.

Similar considerations apply to:

> (16.b) **Either the King of France plays golf or there is a King of France.*

Inconsistent Discourses

An example of inconsistent discourse involving presupposition is

(02) (*The King of France is bald*) $\circ$ (*There is no King of France*).

This has the form $(A^B) \circ (\neg B)$, and a tableau representation like:

$$\begin{array}{c} \boldsymbol{A^B} \\ B \\ \underline{\boldsymbol{\neg B}} \end{array}$$

Once a discourse has become inconsistent it is no longer useful as a source of information. Because of this, such a discourse can be interpreted as an invitation to revise the information state to obtain a consistent set of beliefs.

Similar, but involving a different relative position of the conflicting information, is

(93) (*There is no King of France*) $\circ$ (*The King of France is bald*).

This has the form $(\neg B) \circ (A^B)$, and a tableau representation:

$$\begin{array}{c} \boldsymbol{\neg B} \\ \boldsymbol{A^B} \\ \underline{B} \end{array}$$

In this case, the first sentence of the discourse is inconsistent with the presupposition of the second sentence.

A similar case is that of discourses where a sentence with certain presuppositions is followed by a sentence with contradicting presuppositions. This corresponds to examples such as:

(94) (*Bill did not start smoking*) $\circ$ (*Bill stopped smoking*)

of the form $(\neg A^{\neg B}) \circ (C^B)$.

The UAP tableau for this discourse is:

$$\begin{array}{c} \boldsymbol{\neg A^{\neg B}} \\ \neg B \\ \boldsymbol{C^B} \\ \underline{B} \end{array}$$

This is related to the previous examples in the sense that it evokes an intermediate example like:

(95) (*Bill did not start smoking*) $\circ$ (*Bill used to smoke before*) $\circ$ (*Bill stopped smoking*).

In the original example, the second sentence of this discourse has been ommited from the asserted component of the discourse but appears in the presuppositional contribution of the second sentence. It is this ghostly sentence that is inconsistent with the presuppositions of the first sentence in the same way as those above.

4.3 Distinguishing Presupposition from Other Information

4.3.1 Distinguishing Presupposition and Assertion

Ideally, it should be possible to identify, for a given representation of the context, the following distinct elements: the logical element of the discourse, the presuppositional element of the discourse, the logical contribution of a given sentence to the discourse, and the presuppositional contribution of a given sentence to the discourse. The context itself should include all these in a unified representation that can be used for interpretation.

The framework as defined has *logical expansion rules* ($\neg$-rules, α-rules, and β-rules) and *presuppositional expansion rules* (σ-rules and ξ-rules). The logical component of a sentence, the information that the sentence conveys without taking its presuppositional information into account, can be represented by a tableau for the sentence constructed using only logical expansion rules. This logical component can be taken to represent the asserted content of the sentence. The full representation of a sentence in the framework is a tableau constructed using all applicable rules in an appropriate order. This representation includes both logical and presuppositional information.

UAP tableau are based on the idea that the presuppositions of a sentence are added to its representation during interpretation, and become indistinguishable from other information in that representation. As a result, the difference between presupposition and assertion is lost (both asserted and presupposed propositions have the same final representation as atomic propositions that are added to the tableau).

The presuppositional component can be extracted from the full representation by comparison with the logical component. To be able to isolate the logical and the presuppositional element of a discourse would require a parallel process of constructing a tableau for the same discourse in which only $\neg$-rules, α-rules, and β-rules are used. Such a tableau would represent the logical element of the discourse. The presuppositional element would be all the information that is present in the representation of context but not in the logical element of the discourse.

The *presuppositional contribution* of a sentence can be defined as all the propositions that are added to the corresponding branch during a given stage of interpretation from the moment that the first σ or ξ rule is applied to the sentence (all presuppositions of the sentence irrespective of whether they are first order or higher order). The constraint on application of the rule for negative presuppositional sentences is defined in terms of the whole presuppositional contribution of a presuppositional sentence rather than its first order presuppositions only.

For sentence

> (70.c) *If Mary has had a bath, then Bill regrets that there is no hot water left*

the representation in this framework would be:

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{m \to r^{\neg w}}} \\ \hline \neg m & \neg m & m \\ r^{\neg w} & \neg r^{\neg w} & r^{\neg w} \\ \neg w & \neg w & \neg w \end{array}$$

The presuppositional contribution of this sentence is $\neg w$.

4.3.2 The Traditional Concept of Presuppositions

UAP consequence can be used to define the concept of presupposition of a logical statement in terms of the presuppositions of its atomic subformulas.

Given a compound P and an atomic formula A such that A is a subformula of P and A is known to presuppose a proposition B, B can be said to be a *presupposition* of P iff $P \vdash_{UAP} B$ and $P \not\vdash B$.

This apparently trivial definition contains a lot of information as to what can be considered a presupposition.

A proposition B can be considered a presupposition of (a use of) a proposition P iff:

- B appears as a possible presupposition (signal) of some atomic proposition A^B that is a subformula of P
- B cannot be said to be a logical consequence of P
- B is a UAP consequence of P

None of these conditions on its own would make a proposition B count as a presupposition of P. There will be countless propositions that are not logical consequences of P without being presupposed by P.

There may be presuppositions of atomic subformulas of P that are not presuppositions of P. These correspond to cases where traditionally the presupposition was said not to project. Take a proposition P that contains A^B as a subformula. In the most general case, the expansion of P may have A^B in some branches and $\neg A^B$ in others. This means that in some branches the addition of presupposition B will be attempted by application of σ-rules and in others by application of ξ-rules. In the case of ξ-rules, where the addition succeeds or not will depend on the context as determined by that branch. However, whether or not the proposition P can be said to presuppose B will depend simply on whether at the final stage of interpretation B is present in all branches of the tableau.

In the following example,

(69) *If the typewriter is blue then Sue will be happy*

let $P \equiv b^t \rightarrow h$ and $A^B \equiv b^t$. To this sentence corresponds the representation given above.

$\boldsymbol{b^t \rightarrow h}$		
$\neg b^t$	$\neg b^t$	b^t
h	$\neg h$	h
t	t	t

Adding $\neg t$ to this tableau closes it, so the presupposition t is now a UAP consequence of the sentence, as desired in this case. Because $b^t \rightarrow h \not\models t$, the proposition t can be said to be a presupposition of the proposition $b^t \rightarrow h$. This matches the intuitive predictions that the sentence

(69) *If the typewriter is blue then Sue will be happy*

presupposes the sentence

(59) *There is a typewriter.*

The proposition t appears in the first two branches as a result of the application of a ξ-rule and in the last one as the result of an application of a σ-rule.

If A appears as a negative literal in any of the (open) branches, then in each of those branches the decision whether to expand it must be taken locally. This local approach to deciding reflects an aspect of the operation of defaults: in a disjunction where defaults apply to both disjuncts, the only way to decide whether the default applies for the disjunction is to consider the case of each disjunct separately and allow the default only if it applies in both cases. This is known as reasoning by cases. Just like in the case of defaults, presupposition only holds as a UAP consequence of the whole tableau if it survives in all the open branches. In this way, the tableau framework captures the mechanism of reasoning by cases implicitly (each branch corresponds to a possible case).

If A appears as a positive literal in all the (open) branches, then the presupposition works as an entailment just as Mercer postulates: because it is not defeasible in any of the cases, the expansion rules operate directly and give the presupposition as a UAP consequence of the context.

The UAP tableau for sentence

(74) *If John is married and he has children, then his children are at school*

would be:

$$
\begin{array}{ccccccc}
\multicolumn{7}{c}{\boldsymbol{(j \wedge c) \rightarrow a^c}} \\
\hline
\multicolumn{3}{c}{\neg(j \wedge c)} & \multicolumn{3}{c}{\neg(j \wedge c)} & j \wedge c \\
\multicolumn{3}{c}{a^c} & \multicolumn{3}{c}{\neg a^c} & a^c \\
\hline
\neg j & \neg j & j & \neg j & \neg j & j & j \\
\neg c & c & \neg c & \neg c & c & \neg c & c \\
\underline{c} & c & \underline{c} & & c & & c
\end{array}
$$

c is not a UAP consequence of this tableau, therefore it is not a presupposition of the sentence. This prediction is intuitively correct.

There may be UAP consequences of P that are not presuppositions of P in the traditional sense. These are the propositions that become logical consequences of the information state that results from extending the context with the presuppositions of P.

Only those propositions in the presuppositional contribution that are also UAP consequences of the final tableau are considered as *presuppositions of the discourse*.

The presuppositional contribution of a compound X to a discourse Γ may be different from the presuppositions of X as defined by the UAP tableau for X on its own. The fact that the constraint on ξ-rules is checked against the whole length of the branch of the context to which the sentence is added in the case of interpretation in context, whereas the same constraint is only checked against the representation of the sentence itself when it is interpreted on its own. This leads to the contribution of ξ-rules to the extension of the sentence being different in one case and the other.

The interpretation of the discourse

> (75) (*If Mary has had a bath, then there is no hot water left*) $\circ$ (*If Mary has had a bath, then Bill regrets that there is no hot water left*)

already discussed in terms of ET tableaux in chapter 4 can now be analysed fully in terms of the new concepts of presuppositional contribution and presupposition introduced in this chapter. The interpretation would take place as follows:

A) the sentence $m \rightarrow \neg w$ is expanded using the corresponding β- and $\neg$-rules:

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{m \rightarrow \neg w}} \\ \hline \neg m & \neg m & m \\ \neg w & \neg\neg w & \neg w \\ & w & \end{array}$$

The representation obtained constitutes the logical contribution of the sentence $m \rightarrow \neg w$ to the discourse.

Because the sentence $m \rightarrow \neg w$ contains no presuppositional atomic propositions, no σ- or ξ- rules can be applied to it. The sentence $m \rightarrow \neg w$ is said to make no presuppositional contribution to the discourse. The resulting representation constitutes the context for the interpretation of the following sentence.

B) The next sentence of the discourse is added and expanded.

$$\begin{array}{ccccccccc} \multicolumn{9}{c}{\boldsymbol{m \rightarrow \neg w}} \\ \hline \multicolumn{3}{c}{\neg m} & \multicolumn{3}{c}{\neg m} & \multicolumn{3}{c}{m} \\ \multicolumn{3}{c}{\neg w} & \multicolumn{3}{c}{\neg\neg w} & \multicolumn{3}{c}{\neg w} \\ \multicolumn{3}{c}{} & \multicolumn{3}{c}{w} & \multicolumn{3}{c}{} \\ \multicolumn{3}{c}{\boldsymbol{m \rightarrow r^{\neg w}}} & \multicolumn{3}{c}{\boldsymbol{m \rightarrow r^{\neg w}}} & \multicolumn{3}{c}{\boldsymbol{m \rightarrow r^{\neg w}}} \\ \hline \neg m & \neg m & m & \neg m & \neg m & m & \neg m & \neg m & m \\ r^{\neg w} & \neg r^{\neg w} & \underline{r^{\neg w}} & r^{\neg w} & \neg r^{\neg w} & \underline{r^{\neg w}} & \underline{r^{\neg w}} & \underline{\neg r^{\neg w}} & r^{\neg w} \\ \neg w & \neg w & & \underline{\neg w} & & & & & \neg w \end{array}$$

The last line of this tableau constitutes the presuppositional contributions of the sentence $m \rightarrow r^{\neg w}$ to the discourse.

At this stage, it is important to take into account the fact that only propositions that are UAP consequences of the discourse can be considered as presuppositions of the discourse (or presuppositions of the sentence in the context). In this case, there have been alterations to the representation resulting from presuppositional expansion rules (in stage B), but these alterations affect only some of the branches of the tableau. This presuppositional contribution is different from the presuppositions of sentence $m \rightarrow r^{\neg w}$ on its own in as much as $\neg w$ now appears only in some branches. Because they do not affect all the open branches of the tableau, they do not lead to the addition of presuppositional consequences to the tableau. The sentence $m \to r^{\neg w}$ makes a presuppositional contribution to some branches of the discourse. However, the discourse does not presuppose $\neg w$ even though the sentence itself on its own did.

In chapter 3 the different cases of presupposition behaviour where described as cases of satisfaction, cases of cancelation or hybrid cases. This section shows that the introduction of the presuppositional rules preserves these distinctions.

The presuppositional expansion rule would close branches that hold the negation of the presupposition, $\neg B$. This may have an effect on the definition of cases of cancellation and hybrid cases. Sentences (or discourses) that were cases of cancellation under the compositional rules of chapter 3 may become altogether cases of inconsistency whenever the presuppositional sentence was positive. This has been shown above to have an intuitive interpretation in ruling out sentences that are intuitively unacceptable, like

> (19.a) **There is no King of France and the King of France plays golf.*

Hybrid cases are characterised by having both branches that become closed on addition of B and branches that become closed on addition of $\neg B$. As such, under the new rule they could become cases of satisfaction. This would happen if all their branches that became closed on addition of B during testing are actually closed during construction of the tableau by the effect of the new rule. This may not happen. Branches that become closed in such a way must hold both $\neg B$ and A^B, where A is a positive sentence. For each such branch (by Coverage Property over the initial tableau) there will be another branch that only differs from the original branch in that it holds $\neg A^B$ instead of A^B. This branch would not be closed by the expansion rule during construction but would still become closed on addition of B during testing. So the conceptual differences between hybrid cases and cases of satisfaction are preserved.

With the expansion rule for presuppositions all presuppositions of a tableau (as given by the rules in chapter 3) become presuppositional consequences of the discourse.

The argument runs as follows:

- given a tableau that presupposes B according to the compositional rules of chapter 3, all its branches will hold either A^B or $\neg A^B$, none of its branches can hold B or $\neg B$, and none of its branches can hold $C^{\neg B}$ (by Rules)

- therefore either a σ- or a ξ- expansion rule for the presupposition B will be applied in every branch to obtain a presuppositional tableau

- if every branch of the presuppositional tableau holds B then the tableau closes on addition of $\neg B$

- therefore B is a presuppositional consequence of the discourse.

So the presuppositional expansion rules of chapter 4 subsume the compositional rules of chapter 3, and also add information that was not available before (presuppositional axioms and default extensions).

If a method is given for stringing together the tableau representations of two sentences A and B into a single tableau, the resulting tableau can be used as a representation of the discourse $(A) \circ (B)$.

Given the representation of a sentence, a representation of a discourse could be obtained by adding the representation of a second sentence to each and every one of the branches of the representation of the first sentence. This method can be reiterated, the next sentence being added to each and every one of the branches of the resulting discourse.

In this approach, the presuppositions of a discourse have to be worked out separately from the presuppositions of each of the sentences in it, because when the final representation of the discourse is constructed, the presuppositions of each sentence have already been added to the representation of the sentence.

In order to obtain the presuppositions of a discourse some additional rules would have to be defined. These rules are the same as the compositional rules given in chapter 3.

The distinction between presupposition and assertion remains clear: each sentence can be said to contribute a presuppositional element and an assertive element.

This is the conceptual approach underlying original ideas of compositionality, or *projection*.

Given the definition above for tableau for a sequence of sentences, the compositional rules for presupposition given in chapter 3 can be applied to the examples presented in this chapter. These rules give the correct predictions with respect to presupposition. The advantages of the expansion rules over the compositional rules are: the UAP tableau representation is homogeneous, first and higher order presuppositions are governed by the same rules; and UAP tableaux provide predictions about acceptability that are not captured by compositional rules.

4.3.3 Dealing with Compound Presuppositions and Constructed Presuppositions

The specification given for the application of expansion rules would also capture the possibility that presuppositions may be compound sentences that require expansion by means of $\neg$-, α- or β-rules.

This has not been taken into account in earlier chapters. So far I have been taking as primitive presuppositions those that originate from atomic presuppositional sentences, and I have tried to show how the presuppositions of compounds can be worked out from those by using a set of tableau expansion rules. The presuppositions obtained so far have been either atomic propositions or negated atomic propositions. There are cases, as shown in chapter 1, where an atomic presuppositional sentence can be seen to have a conjunction of propositions as its presupposition. Such presuppositions may be treated like atomic presuppositions, added to the tableau using σ- or ξ-rules, and then expanded using α-rules. It may be interesting to consider whether the order of expansion of the conjuncts of this presupposition determined by the definition of UAP tableau is significant to the predictions for these cases.

There is a different type of compound presuppositions that may arise. These do not originate through a process of selection among the presuppositions of a compound, but rather through a process of combination of the presuppositions of the components by means of logical connectives (disjunction, in particular). Because of the conceptual difference in the type of compositionality involved, these are treated separately.

Treatment of the problem of compound presuppositions has been postponed until this point due to the conceptual differences between the type of projection involved in this case and that of simple atomic presuppositions. The theory presented here attributes only atomic presuppositions to atomic sentences. It is only during projection that presuppositions can become disjunctions or implications, but then they are no longer presuppositions of atomic sentences. This section outlines some problems that the treatment of these presuppositions in this framework would encounter.

Within the framework of UAP tableaux, the sentence

(73) *Either Bill has started smoking or Bill has stopped smoking*

had the following representation:

$$\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{e^{\neg s} \vee p^{s}}} \\
\hline
e^{\neg s} & e^{\neg s} & \neg e^{\neg s} \\
p^{s} & \neg p^{s} & p^{s} \\
\neg s & \neg s & \\
\underline{s} & & s
\end{array}$$

There is a certain amount of controversy in the literature as to whether compounds with conflicting presuppositions such as this one should be taken to presuppose nothing at all or to presuppose a proposition of the form $X \vee \neg X$. In this case, sentence (37) could be said to presuppose

(96) *Either Bill did not smoke or Bill did smoke.*

Mercer [27] defends that such presuppositions may be meaningful when the presuppositions involved are mutually exclusive propositions that do not exhaust the whole range of possibilities. In such cases, the practice of according informative value only to those presuppositions that can be seen as presuppositions of the whole assertion being considered may lead to loss of information. The cases that Mercer refers to are beyond the capabilities of the present framework, because the framework relies heavily on fact that any proposition under consideration must be either true or false in a given alternative. However, disjunctive presuppositions of the type considered could be obtained from the framework as it stands through an additional process of interpretation.

The question is whether presuppositions are informative only in those cases where they can be considered presuppositions of the whole compound, or whether they can also be said to contribute information locally in cases where they are not presuppositions of the whole.

If local presuppositions are to be considered by means of an extra process of interpretation, a compound will have two slightly different types of presuppositions. These presuppositions will differ in how they originate from the representation.

The first kind of presuppositions are presuppositions of the compound by virtue of being presuppositions of every one of the columns that represent it.

The second kind of presuppositions are constructed from those of the presuppositions of the columns that did not fall under the first category. They form a logical compound where the local presuppositions appear related with connectives as constrained by the relative positions of their presuppositional sentences within the tableau. So if two columns Γ_1 and Γ_2 under a compound have presuppositions P_1 and P_2 respectively (the tableaux as given restrict the possible cases to those where $P_1 = \neg P_2$ or vice versa) then the compound can be said to have a presupposition $P_1 \vee P_2$. I refer to these presuppositions as *constructed presuppositions*.

For example (73) above can be attributed a presupposition of the form $s \vee \neg s$. The same can be said of a whole class of examples of this type, such as:

> (15.a) *Either John has stopped beating his wife or he hasn't begun yet. (Gazdar)*
>
> (15.d)*Either Bill has just started smoking or he has just stopped smoking. (Soames)*

However, because of the problems with examples of this type involving temporal considerations, it is more interesting to consider a different sort of example. Some good candidates might be:

> (15.c) *Bill has met either the king or the president of Slobovia. (Karttunen)*
>
> (15.e) *Your teacher is a bachelor or a spinster. (Mercer1988)*

However, these examples would also require complex modelling in propositional logic if the full behaviour is to be captured. For instance, axioms would be required to represent the ideas that if Buganda has a King it does not have a President and if Buganda has a President it does not have a King, which underlie the reasoning on which the first example above is based[3].

Once an appropriate formalization of this mechanism is given, presuppositions of this type might be incorporated into the given framework. For this purpose the expansion rules would have to be modified so that they be applicable to any A^B that is an atomic proposition or any formula X such that the UAP tableau for X licenses a constructed presupposition B. However, in the present framework all constructed presuppositions will be tautologies of the form $P \vee \neg P$. Adding them to a tableau would not play any significant role in the interpretation and it would lead to redundant branching.

4.4 Conclusions

The tableau expansion rules for presuppositions build on the intuitions discussed over the examples given in chapter 3. The tableau proof theory accounts for the defeasibility of presuppositions of negative sentences, and for the unacceptability of a number of unintuitive sentences (including sentences that denote an unnecessarily complex logical structure, and sentences where some propositions are redundant).

[3] Incidentally, it is possible for a country to have both a king and a president. In Spain, for instance, the acting prime minister is referred to as the 'president of the government', or 'the president' for short'.

Chapter 5

Propositional UAP Tableaux and Semantics

5.1 Soundness and Completeness of the UAP Formalism with Respect to Classical Semantics

The present section sets out to address the question of how (or whether) the inclusion of presupposition in a logical calculus forces the semantics of the resulting new calculus to depart from classical models.

5.1.1 Classical Semantics

The main advantage of having a homogeneous representation of assertion and presupposition is that a common semantics for both can be considered. To serve as starting point for the development of this common semantics, the following elementary semantics is given for the ET tableau of chapter 3 (see Fitting [6]).

An L-structure is a function from the set X of atomic formulas of the language L to the set $\{T, F\}$ of truth-values.

A formula P of L is true in an L-structure U ($U \models P$), as given below.

For each sentence letter p, $U \models p$ iff the L-structure U assigns T to p.

For all formulas A, B of L

- $U \models \neg A$ iff it is not true that $U \models A$
- $U \models (A \wedge B)$ iff $U \models A$ and $U \models B$
- $U \models (A \vee B)$ iff either $U \models A$ or $U \models B$ or both
- $U \models (A \rightarrow B)$ iff either $U \models A$ and $U \models B$ or neither $U \models A$ nor $U \models B$

The statement $U \models A$ is read as 'U is a model of A'.

The statement $\models A$ means that for every L-structure U, $U \models A$. In that case, A is called a *tautology*.

For A, B any formulas of L, the statement $A \models B$ means: for every L-structure U, if $U \models A$, then $U \models B$.

The consideration of discourses allows extension of the semantic concepts to include logical consequence of a sequence of sentences.

For $P_1, \ldots, P_n, Q$ any formulas of L, the statement $(P_1) \circ \ldots \circ (P_n) \models Q$ means: for every L-structure U, if $U \models P_1$ and $\ldots$ and $U \models P_n$, then $U \models Q$.

(These are statements about formulas and not formulas or discourses themselves).

In terms of these semantics, the concept of a state of the world as used informally in the discussions about the choice of representation in chapter 3, corresponds to a set of L-structures. Both a 'state of the world' and a set of L-structures constitute representations of an information state.

The set of all possible L-structures corresponds to the zero information state. The L-structures that represent a given information state can be considered as the *valid* L-structures in that state. An increase in information corresponds to a decrease in the number of valid L-structures.

5.1.2 Classical Validity and UAP Validity of Sentences

Lemma 1 *If* $\models X$ *then* $\vdash_{UAP} X$.

Proof: If $\models X$ then $\vdash X$. If $\vdash X$ then the simple tableau for $\neg X$ is closed. If the simple tableau for $\neg X$ is closed the presuppositional tableau is also closed. If the presuppositional tableau is closed then $\vdash_{UAP} X$.
QED

The new calculus is complete with respect to validity over classical models. Valid statements in classical models will be given as valid by the decision method.

Lemma 2 *There are cases where* $\vdash_{UAP} X$ *and* $\not\models X$.

Proof:
If X is a PP axiom, then $\vdash_{UAP} X$ and $\not\models X$.
QED

This can be seen in the example

(97) *If John's children are at school then John has children.*

The rules give $\vdash_{UAP} a^c \rightarrow c$, but $\not\models a^c \rightarrow c$.

The new calculus is not sound with respect to validity over classical models. Statements given as valid by the decision method do not correspond to valid statements in classical models. This need not be dangerous because the new calculus can be proved to be sound with respect to classical models restricted by the PP axioms. Let Π stand for the set of PP axioms.

Lemma 3 *If* $\vdash_{UAP} X$ *then* $\Pi \models X$.

Proof: Given a proposition X, let Γ stand for the tableau for X built using only $\neg$-, α- and β- rules, and let Γ_p stand for the presuppositional tableau for X. If $\vdash_{UAP} X$ then Γ_p is closed. There are two different ways in which this can happen, depending on whether the tableau Γ is closed or not. If Γ is closed then $\vdash X$ and $\models X$ follows from the soundness of classical logic. If Γ is not closed then it must be the case that Γ_p closes in virtue of some proposition introduced by σ-rules (ξ-rules cannot close branches).

These rules would apply to propositions of the form $A_i^{B_i}$. For each such proposition there is a PP axiom $A_i \rightarrow B_i \in \Pi$. By the definitions of the corresponding rules (σ-rules and β-rules), $\vdash_{UAP} X \equiv \Pi \vdash X$. It would only be possible to have a counterexample if $\vdash_{UAP} X$ relies on the tableau for $\neg X$ closing due to the application of a ξ-rule. But this is expressly forbidden by the constraint on ξ-rules. If the tableau for $\neg X$ closes, it must either be because $\vdash X$ or because $\{PPAx\} \vdash X$. In either case, $\Pi \models X$.

QED

For example (71) given earlier, if $\vdash_{UAP} a^c \rightarrow c$ then $\{\ldots, a \rightarrow c, \ldots\} \models a \rightarrow c$.

Lemma 4 *If* $\Pi \models X$ *then* $\vdash_{UAP} X$.

Proof: By lemma 1, if $\models X$ then $\vdash_{UAP} X$. Because $\models$ stands for classical semantic consequence, it is monotonic. So there can be no sentences X such that $\models X$ and $\Pi \not\models X$. To prove the present lemma it is enough to show that $\vdash_{UAP} p$ for any $p \in \Pi$. Every $p \in \Pi$ is of the form $A^B \rightarrow B$ To show $\vdash_{UAP} A^B \rightarrow B$, the tableau for $\neg(A^B \rightarrow B)$ is constructed.

$$\begin{array}{c} \neg(A^B \rightarrow B) \\ \hline \neg A^B \\ \neg B \\ \underline{B} \end{array}$$

Such tableaux are always closed.

QED

5.1.3 Classical Consequence and UAP Consequence

Lemma 5 *If* $Y \models X$ *then* $Y \vdash_{UAP} X$.

Proof: If $Y \models X$ then $Y \vdash X$. If $Y \vdash X$ then the simple tableau for Y extended with $\neg X$ is closed. Applying σ- and ξ- rules to the simple tableau for Y cannot eliminate any of the atomic propositions in its branches. If the simple tableau for Y closed when extended with $\neg X$, the presuppositional tableau for Y also closes when extended with $\neg X$. If the presuppositional tableau for Y closes when extended with $\neg X$, then $Y \vdash_{UAP} X$ (from the definition of presuppositional consequence).

QED

The new calculus is complete with respect to logical consequence over classical models. Logical consequences as defined over classical models will be given as logical consequences by the decision method.

Lemma 6 *There are cases where $Y \vdash_{UAP} X$ and $Y \not\models X$.*

Proof: If $Y \rightarrow X$ is a PP axiom, then $Y \vdash_{UAP} X$ and $Y \not\models X$.
QED

Take

(98) *John's children are at school*

and

(99) *John has children.*

Then (98) $\vdash_{UAP}$ (99) but (98) $\not\vdash$ (99), or $a^c \vdash_{UAP} c$ but $a^c \not\models c$.

The new calculus is not sound with respect to logical consequence over classical models. Logical consequences as defined by the decision method (presuppositional consequences) do not correspond to logical consequences as defined over classical models.

Lemma 7 *There are cases where $Y \vdash_{UAP} X$ and $\Pi \cup \{Y\} \not\models X$.*

Proof: If Y is of the form $\neg A^X$ then $Y \vdash_{UAP} X$, and $\Pi \cup \{Y\} \not\models X$.
QED

Because the formulation of the Lemma allows full interpretation of negated sentences to have taken place before the consequence relation is tested (unlike the case for validity), ξ-rules may play a role in the closure of the presuppositional tableaux. Since ξ-rules are not captured by the PP axioms, the correspondence to classical consequence is lost.

For examples

(100) *John's children are not at school*

and (99) as above, $\neg a^c \vdash_{UAP} c$ but $\{\neg a^c\} \cup \{\ldots, a \rightarrow c, \ldots\} \not\models c$.

Logical consequences as defined by the decision method (presuppositional consequences) do not correspond to logical consequences as defined over classical models, even if consideration is restricted to classical models where the PP axioms hold.

5.1.4 Conclusions

The issue of soundness and completeness of the different types of tableaux described in chapters 3 and 4 is best summarised in terms of how the corresponding semantics progressively depart from the classical models, and how radical the departure involved is.

The ET tableaux (as shown in chapter 3) are sound and complete with respect to traditional tableaux, and therefore also with respect to classical semantics.

The introduction of σ-rules to the proof theory restricts the models that can be considered to those where the PP axioms hold. The tableaux that would result from using only $\neg$-, α-, β- and σ-rules would be sound and complete with respect to the classical models restricted by the PP axioms.

The introduction of ξ-rules produces a greater departure from classical models. The ξ-rules are defeasible, and they involve a choice of interpretation constrained by context. The information introduced by ξ-rules is unsound in classical terms. It corresponds to information that is true in some models but not in others, because ξ-rules involve a choice of model.

5.2 Properties of the UAP Consequence Relation

UAP consequence can be shown to be reflexive, and non transitive. The UAP consequence is monotonic over discourses but non monotonic over sentences.

The assumption that presuppositions are not defeasible by information received after they have been interpreted is shown below to ensure that the consequence relation that operates over discourses in the final system is monotonic. This corresponds to accepting the interpretation process as a source of beliefs rather than a source of truth. Such an assumption results in the need for a mechanism for revising the set of beliefs whenever they become inconsistent.

It must be noted that the concept of UAP consequence as defined in the present framework is accessory to the concept of presupposition, but different from it. The relation of UAP consequence that this section is concerned with is developed as a tool for the framework to handle the relation of presupposition, not as direct model of it. The relation of presupposition is defined in terms of the relation of UAP consequence and the relation of logical consequence as described in chapter 4.

5.2.1 Reflexivity

Reflexivity still holds in a framework with the new expansion rules.

Lemma 8

$$X \vdash_{UAP} X$$

Proof :
The tableau for $(X) \circ (\neg X)$ is always closed.
QED

Lemma 9 $(P_1) \circ \ldots \circ (X) \circ \ldots \circ (P_n) \vdash_{UAP} X$

Proof : Given the definition of tableau for a discourse, X will appear in all open branches of the tableau for this discourse. So the tableau will close on addition of $\neg X$.
QED

5.2.2 Presupposition as Inference

Lemma 10 $A^B \vdash_{UAP} B$

Proof :
This can be shown to hold by applying the definition of UAP consequence. The tableau for the discourse $(A^B) \circ (\neg B)$ is always closed because the expansion rule for A^B has been applied before $\neg B$ is interpreted, so that B is always present in the tableau.
QED

Lemma 11 $\neg A^B \vdash_{UAP} B$

Proof : This can be shown to hold by applying the definition of UAP consequence. The tableau for the discourse $(\neg A^B) \circ (\neg B)$ is always closed because there is no context that can block application of the ξ-rule, so the expansion rule for $\neg A^B$ will always have been applied before adding $\neg B$ for the purpose of testing logical consequence, and therefore B is always present in the tableau.
QED

Lemma 12 $(P_1) \circ \ldots \circ (A^B) \circ \ldots \circ (P_n) \vdash_{UAP} B$

Proof :
Given the definition of tableau for a discourse, and the application criteria for σ-rules, A^B will appear in all open branches of the tableau for this discourse and it will be expanded with B in every case. So the tableau will close on addition of $\neg B$.
QED

Lemma 13 *There are cases where* $(P_1) \circ \ldots \circ (P_{j-1}) \circ (\neg A^B) \circ (P_{j+1}) \circ \ldots \circ (P_n) \not\vdash_{UAP} B$.

Proof : If there is some P_i , $i < j-1$ such that $P_i \equiv \neg B$, then $(P_1) \circ \ldots \circ (P_{j-1}) \circ (\neg A^B) \circ (P_{j+1}) \circ \ldots \circ (P_n) \not\vdash_{UAP} B$.
QED

5.2.3 Monotonicity

The resulting consequence relation is not always monotonic. Its behaviour varies depending on whether $\wedge$ or $\circ$ are used to put propositions together. The following lemmas illustrate this point more clearly.

Lemma 14 *There are cases where $X \vdash_{UAP} Y$ and $X \wedge Z \not\vdash_{UAP} Y$ for some Z.*

Proof : If $X \equiv \neg A^B$, $Y \equiv B$ and $Z \equiv \neg B$ then $X \vdash_{UAP} Y$ but $X \wedge Z \not\vdash_{UAP} Y$ ($\neg A^B \wedge \neg B \not\vdash_{UAP} B$).
QED

Lemma 15 *If $\Delta \vdash_{UAP} X$ then $\Delta \circ \Gamma \vdash_{UAP} X$ for any Γ.*

Proof : If $\Delta \vdash_{UAP} X$ then the tableau for $\Delta \circ (\neg X)$ closes. Given the definitions of tableau for a discourse, the tableau for $\Delta \circ \Gamma$ must include the tableau for Δ in every one of its branches, because it will be built by starting an expansion of Γ from each branch of the tableau for Δ. If adding $\neg\phi$ closed the tableau for Δ no defeasible expansion from $\neg X$ may have been involved (by definition, defeasible inferences do not close branches). So the same non-defeasible expansions will apply in the case of $\Delta \circ \Gamma \circ (\neg X)$. And because the tableau for Δ is included in the tableau for $\Delta \circ \Gamma$, this one will also be closed.
QED

This result may be surprising given that presuppositions are allowed to be defeasible. However, defeasibility is only allowed with respect to previous or simultaneous information as part of the interpretation process [1].

(At most, when $\Gamma \vdash \neg X$, then $\Delta, \Gamma \vdash X$ and $\Delta, \Gamma \vdash \neg X$).

The differences between $\wedge$ and $\circ$ lie in whether the interaction that threatens monotonicity (defeasibility of presuppositions) is computed across a sentence boundary or not. This can be interpreted as follows. Information built up from complete sentences behaves monotonically. The relation between a sentence A and it s information content can be considered non-monotonic in the sense that if the sentence A is used as a subpart of a bigger sentence B, the contribution of A to the information content of B may be less than the information content of A when used on its own.

5.2.4 Transitivity

The resulting consequence relation is not always transitive.

Lemma 16 *There are cases where* $X \vdash_{UAP} Y$ *and* $Y \vdash_{UAP} Z$ *but* $X \not\vdash_{UAP} Z$.

Proof: If $X \equiv \neg B \wedge \neg A^B$, $Y \equiv \neg A^B$ and $Z \equiv B$ then $X \vdash_{UAP} Y$ and $Y \vdash_{UAP} Z$ but $X \not\vdash_{UAP} Z$.

QED

Lemma 17 *There are cases where* $\Delta \vdash_{UAP} X$ *and* $X \vdash Y$ *but* $\Delta \not\vdash_{UAP} Y$.

Proof : If $\Delta \vdash_{UAP} \neg B$, $X \equiv \neg A^B$ and $Y \equiv B$, then $X \vdash Y$ but $\Delta \not\vdash_{UAP} B$.

QED

[1]The cases of inconsistent discourses given above constitute good examples of this behaviour. Only those cases where the subsequent information is inconsistent with the preferred interpretation could give rise to a different interpretation of the logical consequence. If these examples were to be interpreted not as inconsistent but as clarifications of which interpretation to prefer, a defeasible consequence relation – like the one presented for sentences – would be required for discourses. This would correspond to not eliminating the 'defeasibility' markings after expanding each sentence.

5.2.5 Conclusions

The consequence relation that results from interpreting UAP tableaux to determine a logical calculus behaves in a very unconventional manner. This suggests that it may be unwise to integrate into a single formal system both the logical and the presuppositional ingredient of language.

The following sections address the question of what other possible interpretations might be given to presupposition and its relation to the logic.

5.3 The Semantic Interpretation of Presuppositional Rules

The decision method obtained by considering all the rules corresponds to a different calculus from the one defined by tableau constructed with only logical expansion rules. This new calculus need not be sound and complete with respect to classical models. This section studies this issue. Because the rules for computing the presuppositions of a tableau take into account the presuppositions of every branch irrespective of whether they are presuppositions of positive or negative sentences, and because the rules for expanding one and the other are slightly different, the general effect of these rules on presupposition behaviour cannot be explained without studying the conceptual implications of both types of rules.

5.3.1 σ -Rule: PP Axioms

Because branches that were not closed in the original formalism may be closed by the σ-rule, tableaux that were not proofs in the original formalism may become proofs in the new one. This effect, however, is in accord with the intuition that taking presuppositions into account produces extra information. The extra information that is being added is equivalent to a set of conditionals of the form $A \rightarrow B$, where A is an atomic proposition that presupposes a proposition (not necessarily atomic) B (so these are conditionals where the antecedent corresponds to the first element of the ordered pair that represents a presuppositional relation, and the consequent corresponds to the second element).

This set of conditionals can be understood as a set of axioms for the logical calculus obtained by extending classical logic with presuppositions of positive presuppositional sentences. I refer to this set of axioms as the *positive presupposition axioms*, or *PP-axioms.* In order for them to be informative at all, these axioms must be stated in terms of actual descriptive propositional letters, and not in terms of metavariables of any sort.

This addition has an intuitive explanation.

Logic is defined as a calculus within very strict conventions. These specify the extent to which the expressions of a logical language can be said to refer to objects or sentences. Logical calculi are built on the assumption that the expressions employed in them are in fact totally ambiguous symbols that can stand for any of the sentences (or objects) that the logic operates on. This means essentially that propositional calculi are defined over metavariables that stand for sentences, rather than over sentences themselves. Usually p and q are taken to be metavariables [2] that stand for any propositional letter that can be fitted to the logical schema being represented. A letter p can be substituted for any other. When using a logic to represent language this convention breaks up. The PP-axioms of the form $p \rightarrow q$ imply that p can no longer be substituted for any other t when it appears in a formula, because only t such that $t \rightarrow q$ can be considered.

If logic is used as a language, or used to represent language, it looses this intrinsic ambiguity. When a proposition of the logic is obtained from a communication act in language, it is a specific proposition that is being used. This becomes even clearer when presupposition is added to the logic. Such a logic is then built of specific instances of propositions, differentiated between them by the fact that some of them presuppose different propositions. Specific propositions have specific presuppositions. Whether an atomic proposition presupposes another proposition depends on which specific atomic proposition is being considered. The inference operating over this language can only be described in terms of inference schema involving metavariables if some way is provided to signal that a metavariable A presupposes another metavariable B. This role is played in the present framework by the notational device A^B. The relation of presupposing would have to be specified separately for the actual propositions of the logic (in terms, for

[2]For further consideration of the tacit traditional use of logical symbols as metavariables, see Hodges [16].

instance, of presuppositional axioms).

The inference operating over this language should take into account that propositions may presuppose other propositions, so that reasoning about A is not the same as reasoning about A^B. The present framework constitutes a formalization of this inference as a calculus.

An immediate consequence of the distinction between the two uses of logic (logic as a metalanguage to describe the structure of an abstract argument, and logic as an instantiation of one particular argument that had been originally phrased in natural language) is that when a logic with presupposition is used as an instantiation of a particular argument, the propositions involved are being used in a particular way, and therefore must be either true or false, and similarly the terms in them must refer to particular objects. This constrains the way in which interpretation of the propositions of such a language must take place. Under these circumstances, issues like soundness and completeness of the original logical calculus with respect to the semantic models of classical logic no longer determine unequivocally the acceptability or unacceptability of the propositions of the logic extended with presupposition –even if it is still the same semantics models that underlie the calculus operating over the extended logic.

There are two basic results of the addition of the PP axioms. On one hand, the axioms themselves (and their contraposed forms) become valid. On the other hand, all statements that are incompatible with the PP axioms become invalid. Since the axioms are of the form $p \rightarrow q$, such statements would be any sentences of the form $p \wedge \neg q$ or $\neg q \wedge p$.

Semantically, these two results can be interpreted as follows. The consideration of presupposition alters the information state described by a given set of propositions. For a given set of propositions, the introduction of the axioms restricts the set of L-structures that can be considered valid by eliminating those where the axioms do not hold. Certain L-structures that were valid for the original language become invalid. These correspond to those where the PP axioms do not hold. Only L-structures where the PP axioms are valid are allowed, so the axioms become sentences that are true in every valid structure (tautologies). These results are apparent in the examples discussed in chapter 4.

5.3.2 ξ-Rule: A Preference Between Models

The ξ rule has a conceptual similarity to default rules. The information represented in a tableau is partial information about the world in the sense that it determines a set of L-structures only over the set of formulas in it, but not over all the formulas of the language. However, an L-structure always covers all the atomic propositions in the language. Because of this, in every L-structure in which $\neg A^B$ is true either B will be true or $\neg B$ will be true. Even if σ-rules are considered, the PP axioms allow both L-structures where $\neg A^B$ and B are true, and L-structures where $\neg A^B$ and $\neg B$ are true. Therefore, a sentence of the form $\neg A^B$ determines an information state that includes both L-structures where B is true and L-structures where $\neg B$ is true. The ξ rule makes presuppositions of negated sentences act as defaults to extend this information: B is added to the branch, which is equivalent to saying that L-structures where $\neg A^B$ and B hold are preferred.

The constraint on the ξ-rule plays the role of ensuring that whenever the ongoing branch of the tableau where the negative presuppositional sentence appears does not in fact correspond to a situation of incomplete information as described above, the addition of the presupposition is blocked if it leads to inconsistency with the existing information.

In terms of the traditional approach to defaults, the rule for presuppositions of negative sentences as it stands is akin to computing defaults over models instead of over theories. Tableau systems are based on the idea of searching for a countermodel. Default reasoning is based on the idea of checking consistency of a possible conclusion. From these starting points, there is an intuitive way to combine the two frameworks. The default approach can be rephrased as: 'add B unless this results in a countermodel'. This default-like condition is applied only locally in the framework because it is triggered only by negated instances of atomic propositions.

The effect of adding the ξ-rule cannot be described in general terms equivalent to the PP axioms because this effect is dependent on the context in which the negative presuppositional sentence appears.

5.4 Presuppositional Redundancy and Presupposition Satisfaction

5.4.1 Redundancy

A discourse $(P_1) \circ \ldots \circ (P_n)$ is redundant if either $P_n \equiv P_i$ for some i, or $(P_1) \circ \ldots \circ (P_{n-1}) \vdash_{UAP} P_n$.

A proposition X added to a branch of a tableau is *redundant* iff X already appeared in that branch prior to the addition.

This definition will prove useful to address the issue of presupposition satisfaction.

Using this definition, an argument can be given for the unacceptability of conjunctions of the type:

(20.a) **The King of France is bald and there is a King of France.*

(20.b) **Bill's friends have encouraged him and he has friends.*

These sentences have the form $A^B \wedge B$. UAP tableaux for them result in:

$$\begin{array}{c} \underline{\boldsymbol{A^B \wedge B}} \\ A^B \\ B \\ B \end{array}$$

The tableau shows a redundant addition of proposition B inherent in the structure of such sentences. It seems reasonable to assume that the intuitive unacceptability of these sentences is related to this redundancy and the fact that they convey the same information that a sentence of the form A^B would have done on its own. This can be argued in terms of pragmatics by involving Grice's maxim of quantity.

A sentence is *pragmatically unacceptable* if there is another sentence involving a smaller number of atomic propositions of the language that conveys the same information.

The issue of redundancy can be reconsidered in the context of discourses. For this purpose, the concept of redundancy is subdivided into two different cases, according to the origin of the redundant proposition.

A redundant proposition X added to a branch of a tableau is *logically redundant* if X is added as an assertion or X is added as the result of the application of a $\neg$-rule, an α-rule or a β-rule.

A redundant proposition X added to a branch of a tableau is *presuppositionally redundant* if X is added as the result of the application of a σ-rule or a ξ-rule.

The definitions of redundancy given allow analysis of examples like the discourse

(101) (*The King of France is bald*) $\circ$ (*There is a King of France*)

or the discourse

(102) (*Bill's friends have encouraged him*) $\circ$ (*Bill has friends*).

These discourses have the form $(A^B) \circ (B)$. Full UAP tableaux for them result in:

$$\begin{array}{c} \boldsymbol{A^B} \\ B \\ \boldsymbol{B} \end{array}$$

The tableau shows that the addition of the second sentence of the discourse, proposition B, is logically redundant.

A discourse is *pragmatically unacceptable* if some sentence in the discourse was a redundant addition to its context.

For the symmetrical version of the discourses given above:

(103) (*There is a King of France*) $\circ$ (*The King of France is bald*).

(104) (*Bill has friends*) $\circ$ (*His friends have encouraged him*) .

the predictions are different.

These discourses have the form $(B) \circ (A^B)$. Full UAP tableaux for them result in:

$$\begin{array}{c} \boldsymbol{B} \\ \boldsymbol{A^B} \\ B \end{array}$$

In this case, the second addition of B is presuppositionally redundant.

It seems reasonable to say that it is only logical redundancy that makes sentences pragmatically unacceptable. Presuppositional redundancy is related to satisfaction of presuppositions and does not give rise to unacceptability.

Presuppositional Redundancy

In certain cases, there is no presuppositional contribution from propositions whose presuppositions have been expanded. This is because such additions to the tableau during the presuppositional extension are presuppositionally redundant propositions.

Some examples are given below. For each example, a table gives the logical transcription of the discourse, its UAP tableau representation, and the predictions made by the proof theory with respect to the presuppositions of the propositions involved.

(105.a) (*John's children have forgotten Bill*) $\circ$ (*Bill regrets that John's children have forgotten him*)

$(B^P) \circ (A^{B^P})$	$\boldsymbol{B^P}$ P $\boldsymbol{A^{B^P}}$ B^P P	presupposes P	does not presuppose B^P

(105.b) (*John has children*) $\circ$ (*Bill regrets that John's children have forgotten him*)

$(P) \circ (A^{B^P})$	$\boldsymbol{P}$ $\boldsymbol{A^{B^P}}$ B^P P	presupposes B	does not presuppose P

(105.c) (*John's children have forgotten Bill*) $\circ$ (*Bill does not regret that John's children have forgotten him*)

$(B^P) \circ (\neg A^{B^P})$	$\boldsymbol{B^P}$ P $\neg\boldsymbol{A^{B^P}}$ B^P P	presupposes P	does not presuppose B^P

(105.d) (*John has children*) $\circ$ (*Bill does not regret that John's children have forgotten him*)

$(P) \circ (\neg A^{B^P})$	$\boldsymbol{P}$ $\boldsymbol{\neg A^{B^P}}$ B^P P	presupposes B does not presuppose P

Logical Redundancy

Certain discourses are logically redundant altogether.

(106.a) (*Bill regrets that John's children have forgotten him*) $\circ$ (*John's children have forgotten Bill*)

$(A^{B^P}) \circ (B^P)$	$\boldsymbol{A^{B^P}}$ B^P P $\boldsymbol{B^P}$ P	B^P is logically redundant

(106.b) (*Bill regrets that John's children have forgotten him*) $\circ$ (*John has children*)

$(A^{B^P}) \circ (P)$	$\boldsymbol{A^{B^P}}$ B^P P $\boldsymbol{P}$	P is logically redundant

A problem surrounds examples like

(107) (*The King of France is not bald*) $\circ$ (*There is a King of France*)

of the form $(\neg A^B) \circ (B)$. These have a tableau representation:

$$\begin{array}{c} \boldsymbol{\neg A^{B}} \\ B \\ \boldsymbol{B} \end{array}$$

In this case, the discourse is classified as logically redundant. The second sentence could be taken to have informative value in the context as an asserted confirmation that the preferred interpretation for the first sentence is correct. The fact that the discourse sounds nonsensical can be taken as confirmation that there actually is a preferred interpretation.

5.4.2 Dealing with Presuppositional Redundancy

The discussion so far in the present section has shown that presuppositional redundancy has no effect on interpretation, even when pragmatic criteria of acceptability are taken into account. This implies that the expansion of presuppositional contributions that are redundant is a waste of time. Furthermore, it suggests that the UAP system as it stands is missing a point somewhere. Ideally, an interpretation system should not carry out operations that are not significant to the final interpretation.

The presuppositions of a sentence that are redundant additions to the context in which it appears should not qualify as presuppositional contributions of that sentence to the resulting discourse. This is related to presupposition satisfaction.

On the other hand, there is still the difference in anaphoric properties between presupposition and traditional entailment to be accounted for. The presuppositional redundancy does involve a certain degree of anaphoric reference by the redundant appearance to the original instance.

This is particularly apparent in some of the examples presented so far. If the presuppositionally redundant propositions are eliminated, the resulting representation is as informative as the original but much less cluttered. In order to keep track of where such eliminations have taken place, I add the symbol $\Uparrow$ wherever a presuppositionally redundant proposition has been eliminated. The symbol $\Uparrow$ is therefore acting as a marker for instances of presupposition satisfaction within a tableau. In order to make explicit in the framework as much as possible of the relevant information, each occurrence of $\Uparrow$ is given as a subscript the redundant proposition that it substitutes. This notation keeps track of which presupposition is being satisfied by each instance of $\Uparrow$.

The sentence

(71) *If there is a typewriter then the typewriter is blue*

corresponds to the following representation.

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{t \rightarrow b^{t}}} \\
\hline
\neg t & \neg t & t \\
b^{t} & \neg b^{t} & b^{t} \\
\underline{t} & & \Uparrow_{t}
\end{array}
$$

the last column of this representation now has no redundant addition of t.

The sentence

(85) *If John's children have forgotten Bill, Bill does not regret it*

has the form $f^{c} \rightarrow \neg g^{f^{c}}$.

The tableau for that would be:

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{f^{c} \rightarrow \neg g^{f^{c}}}} \\
\hline
\neg f^{c} & \neg f^{c} & f^{c} \\
\neg g^{f^{c}} & g^{f^{c}} & \neg g^{f^{c}} \\
c & & c \\
 & \underline{f^{c}} & \Uparrow_{f^{c}} \\
 & & \Uparrow_{c}
\end{array}
$$

The last column shows redundant instances of f^{c} and c.

The non-redundant UAP tableau for sentence

(74) *If John is married and he has children, then his children are at school*

would be:

$$
\begin{array}{ccc|ccc|c}
\multicolumn{7}{c}{\boldsymbol{(j \wedge c) \rightarrow a^{c}}} \\
\hline
\multicolumn{3}{c}{\neg(j \wedge c)} & \multicolumn{3}{c}{\neg(j \wedge c)} & j \wedge c \\
\multicolumn{3}{c}{a^{c}} & \multicolumn{3}{c}{\neg a^{c}} & a^{c} \\
\hline
\neg j & \neg j & j & \neg j & \neg j & j & j \\
\neg c & c & \neg c & \neg c & c & \neg c & c \\
\underline{c} & \Uparrow_{c} & \underline{c} & & \Uparrow_{c} & & \Uparrow_{c}
\end{array}
$$

Comparing with the UAP representation obtained in chapter 4, the non-redundant UAP tableau is very economical in that the only additions required are those that close branches.

However, the addition of redundant information to branches of a tableau is an integral part of the tableaux formalism. This can be easily seen by considering the representation for the discourse

> (75) (*If Mary has had a bath, then there is no hot water left*) ∘ (*If Mary has had a bath, then Bill regrets that there is no hot water left*)

or

$$(\alpha \to \beta) \circ (\alpha \to \delta^{\beta}).$$

If all the presuppositionally redundant information were eliminated the following tableau would result:

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{m \to \neg w}} \\
\hline
\begin{array}{c} \neg m \\ \neg w \\ \\ \boldsymbol{m \to r^{\neg w}} \\ \hline \begin{array}{ccc} \neg m & \neg m & m \\ r^{\neg w} & \neg r^{\neg w} & \underline{r^{\neg w}} \\ \Uparrow_{\neg w} & \Uparrow_{\neg w} & \end{array} \end{array}
&
\begin{array}{c} \neg m \\ \neg\neg w \\ w \\ \boldsymbol{m \to r^{\neg w}} \\ \hline \begin{array}{ccc} \neg m & \neg m & m \\ r^{\neg w} & \neg r^{\neg w} & \underline{r^{\neg w}} \\ \underline{\neg w} & & \end{array} \end{array}
&
\begin{array}{c} m \\ \neg w \\ \\ \boldsymbol{m \to r^{\neg w}} \\ \hline \begin{array}{ccc} \neg m & \neg m & m \\ \underline{r^{\neg w}} & \underline{\neg r^{\neg w}} & r^{\neg w} \\ & & \Uparrow_{\neg w} \end{array} \end{array}
\end{array}
$$

This is almost the same representation as that obtained in chapter 4, but a lot of redundant information no longer appears. This is due to the fact that the non-redundant approach models the concept of satisfaction more closely than the expansion rule paradigm.

However, if all the redundant information (whether presuppositionally redundant or logically redundant) were to be eliminated, the resulting tableau would show a different form:

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{m \to \neg w}} \\
\hline
\begin{array}{c} \neg m \\ \neg w \\ \\ \boldsymbol{m \to r^{\neg w}} \\ \hline \begin{array}{ccc} & & \underline{m} \\ r^{\neg w} & \neg r^{\neg w} & \end{array} \end{array}
&
\begin{array}{c} \neg m \\ \neg\neg w \\ w \\ \boldsymbol{m \to r^{\neg w}} \\ \hline \begin{array}{ccc} & & \underline{m} \\ r^{\neg w} & \neg r^{\neg w} & \\ \underline{\neg w} & & \end{array} \end{array}
&
\begin{array}{c} m \\ \neg w \\ \\ \boldsymbol{m \to r^{\neg w}} \\ \hline \begin{array}{ccc} \underline{\neg m} & \underline{\neg m} & m \\ & & r^{\neg w} \end{array} \end{array}
\end{array}
$$

This is an interesting issue since it brings up the fact that there is a certain redundancy inherent to the tableau formalism. In view of this fact it may not be worth the effort to consider modifications to the rules that eliminate exclusively presuppositional redundancy. However, the redundancy that results from the application of $\neg$-, α- or β-rules does not have the same characteristic of the redundant addition referring to the earlier appearance.

5.4.3 The Problem with Conjunction

A conjunction like

(108) *There is a King of France and the King of France is bald*

behaves just like the discourse

(103) (*There is a King of France*) $\circ$ (*The King of France is bald*).

A conjunction like

(20.a) *The King of France is bald and there is a King of France*

behaves just like the discourse

(101) (*The King of France is bald*) $\circ$ (*There is a King of France*).

There is a great similarity between the tableaux for these conjunctions and the tableau for discourses of the form $(A^B) \circ (B)$ studied earlier. The differing behaviour for the two versions of the conjuncts – $A^B \wedge B$ on one hand and $B \wedge A^B$ on the other – suggests that conjunctions can sometimes be processed as if they constituted simple discourses or sequences of sentences instead of individual formulas.

Redundancy and Conjunction

If an interpretation of conjunctions as discourses is allowed, the distinction between presuppositional and logical redundancy explains the behaviour of the conjunctions.

Applying this alternative interpretation, the tableau for the unacceptable conjunctions would be:

$$\begin{array}{c} \underline{\boldsymbol{A^B \wedge B}} \\ A^B \\ B \\ \boldsymbol{B} \end{array}$$

Where the second addition of B is now the result of the suspended expansion of the α-rule. Because of this, the addition is now logically redundant instead of presuppositionally redundant and the conjunction is correctly predicted to be unacceptable.

This solution also works for the symmetrical conjunctions to those given above:

(109) *There is a King of France and the King of France is bald.*

(18) *Bill has friends and all his friends have encouraged him .*

These sentences have the form $B \wedge A^B$. UAP tableaux for them result in:

$$\begin{array}{c} \boldsymbol{B \wedge A^B} \\ B \\ \boldsymbol{A^B} \\ B \end{array}$$

Under this solution, the addition of B is now only presuppositionally redundant.

However, this modification of the interpretation cannot be taken as a general procedure, but only when the conjunction is the main connective of an assertion, and not when it results from expansion of a larger compound. In the case of $\Gamma \circ (A \wedge B)$, the discourse is expanded as if it were of the form $\Gamma \circ (A) \circ (B)$. However, the same procedure cannot be applied in cases like $\Gamma \circ ((A \wedge B) \rightarrow C)$.

Defeasibility and Conjunction

The sentence

(110) *John has no children and John's children didn't come to the party*

is a problem. This sentence corresponds to the logical form $\neg c \wedge \neg p^c$. The UAP tableau for this logical form would be:

$$\frac{\boldsymbol{\neg c \wedge \neg p^c}}{\begin{array}{c}\neg c \\ \neg p^c\end{array}}$$

As the presuppositional sentences appears negated, the presupposition is not expanded, so the tableau is not closed. Therefore the sentence would be deemed acceptable. The actual intuitions behind this example are not as clear as would be desired. Imagine a situation in which two speakers are debating whether or not John has children. Speaker A argues that John must have children because he saw them at the party. Speaker B is more familiar with John than speaker A, but speaker A is stubborn. After the dispute has gone on for some time, speaker B may attempt to close it off by uttering a sentence like (48). In such sentence, speaker B is actually using the conjunction to bring together in one sentence two propositions that are contextually distant but the speaker wants to put forward together. Because the two propositions are being 'picked out' from their original positions in discourse, an interpretation of one as explanation of the other is possible even though they appear close together when actually presented again. In this sense, the factor of distance is actually cancelled by the pragmatic choice taken by speaker B when she expresses her idea as a conjunction rather than as a discourse of two sentences.

5.5 Presupposition Satisfaction as Abduction

Presupposition as an information handling operation has a tendency to operate only when it adds relevant information. In this sense it behaves like the logical operation of abduction. Starting from this similarity, a comparison between the two brings out interesting results.

In view of all these observed characteristics in presupposition, it is interesting to consider the possibility that there may be an abductive ingredient in presupposition. This would capture on one hand the sense of reference that distinguishes presupposition satisfaction from logical redundancy. On the other hand, if all the inference steps involving presuppositional expansion rules can be interpreted as steps of abductive inference applied as an additional operation to support an ordinary system of deductive inference, the general picture of UAP tableaux and the calculus that results from them would be greatly simplified.

5.5.1 Previous Work

General Concepts

Peirce [35] proposes abduction as the first inferential stage in inquiry, which consists in 'the first stating of a hypothesis and the entertaining of it'.

Peirce gives the following basic schema for abduction.

> The surprising fact C is observed.
>
> But if A were true, C would be a matter of course.
>
> Hence there is reason to suspect that A is true.

According to Peirce, an *abductive hypothesis* or *abductive conjecture* is 'any proposition added to observed facts, tending to make them applicable in any way to other circumstances than those under which they were observed'.

This hypothesis must only be entertained interrogatively and must be tested afterwards by experiment. As long as hypotheses are only accepted interrogatively, the matter of selecting a hypothesis becomes one of economy. As a guideline for this selection, Peirce refers to Galileo's maxim of selecting always the simpler hypothesis. He adds a warning that this should not be taken to refer to the logically simpler hypothesis (that which adds least to what has been observed) because that is too restrictive, but rather to 'the more facile and natural, the one that instinct suggests'.

Formalization

Popple [36] present the first formal specification of a mechanisable abductive procedure for logic. This procedure is based on resolution methods in conjunctive normal form.

An abductive procedure is outlined specifically in the context of S-linear resolution. Applying this method, literals that would have been abandoned by the deduction method on the grounds of not having any successor nodes are taken as candidate hypotheses.

Regarding selection among candidate hypotheses, Popple says: 'A candidate hypothesis is entertained seriously if it arises in the partial search trees of two or more of the data making up the conjunctive observation'. A procedure of *synthesis* is implemented to factor across partial trees. This is presented as a way of implementing the principle of Occam's razor (adopt the hypothesis which is the simplest). Popple interprets this (contradicting Peirce's intuition on the subject) 'in the sense that it contains the smallest number of independent types of elements, adding the least to what has been observed'.

This procedure does not give a unique explanation of the original problem. In order to select among the possible explanations, Popple proposes that the same model be used to generate predictions (by deduction) from the explanations, and that these predictions be then verified empirically (the process may be iterated).

Pirri and Cialdea [3] present a proof theoretical abduction method for first order classical logic. Two versions are defined (a version for the sequent calculus, and a dual version for semantic tableaux). Because the present framework is based on tableaux, this method is described in closer detail below.

The Mechanics of Abduction in Tableaux

Given a current theory and an observation that the theory does not explain, abduction is the production of a hypothesis that would explain the observation if added to the theory.

An abduction problem is a tuple $< \Theta, P >$, where Θ is a logical theory and P is an observation that should be explained by the theory.

An abduction problem $< \Theta, P >$ may have several solutions A such that $\Theta \cup \{A\} \models P$.

Abduction on its own cannot guarantee that any of the solutions be logically valid as an addition to the theory.

Any abductive method that provides only one of the possible solutions must allow for that solution to be defeasible.

Abduction as a process to obtain one explanation is unsound nondeterministic inference, therefore must be tentative and defeasible.

Abduction infers premises from a conclusion. It is an unsound form of inference.

The solution to an *abduction problem* is given by formulas A such that $\Theta \cup \{A\} \models P$, where Θ is a background theory and P is a formula and $\Theta \not\models P$ and $\Theta \not\models \neg P$. For the tableaux case, an abduction problem is expressed as a tableau for $\Theta \cup \{\neg P\}$. A solution can be found among the formulas that force the closure of this tableau.

Certain conditions are imposed on solutions to abduction problems to be considered interesting:

- (a) A is consistent with Θ, i.e. $\Theta \not\models \neg A$,
- (b) A is a minimal explanation, i.e., for any formula B, if $\Theta \cup \{B\} \models P$ and $A \models B$ then $\models B \equiv A$,
- (c) A has the form of a conjunction of literals. [3]

For the propositional case, Cialdea and Pirri give two different characterisations of abductive explanations: (a) generating the whole set of minimal and Θ-consistent explanations, and (b) generating (non-deterministically) a single minimal explanation that is Θ-consistent.

The solution given in (b) is not relevant in this context because there are linguistic constraints to be satisfied by the 'correct' explanation that make some explanations 'better' than others in a way that is not captured by the original algorithm.

In case (a), having ensured an abduction problem is expanded as far as the system in which it is specified allows (definitions of fundamental and

[3]The restriction that an explanation be of conjunctive form ensures that each explanation contains no alternatives, so that each different alternative constitutes a different independent explanation. If disjunctive formulas had been allowed in explanations, one single explanation could have covered several alternatives. An option still open for exploration is to consider synthesis in terms of abstracting a more complex form (including disjunction and/or conditionals) from a selected set of conjunctive explanations.

acceptable for branches/sequents and trees), an abductive explanation (*closure*) is obtained by choosing, for each open branch of the problem, one literal that closes it.

The procedure followed is:

- for each branch construct the set of all possible literals that close it (*closing set*), then
- construct the set of minimal closing sets by taking only those branches that do not include other smaller branches,
- for obtaining each abductive explanation (a conjunction of literals), gather one literal from each set-element of this set of minimal consistent sets.

  This procedure defines the set of abductive explanations for the problem.

5.5.2 Presupposition of Negative Sentences

In cases of presuppositions of negative sentences, presupposition has the same behaviour with respect to validity and defeasibility as information obtained through abduction.

On one hand, because abduction is non deterministic (and unsound), any information obtained by this method must be subject to cancellation on the face of contradiction. This matches the constraints on the ξ-rules, which block the expansion if a contradiction results.

On the other hand, abduction of a proposition B in a context that already held B results in the same context with no modification. This corresponds to the characteristic behaviour of presupposition with respect to redundancy.

In that sense the expansion of negative sentences $\neg A^B$ can be considered as some sort of instruction given by the linguistic form of a sentence to abduce the presupposition B in the context.

Such an abduction would relate to the formal specification given by Cialdea and Pirri for abduction in tableaux as follows. The main differences lie in the different conditions that are imposed on what is considered an interesting abductive solution. Because presupposition as presented so far only considers addition to the context of the proposition that is presupposed, the condition of minimality that Cialdea and Pirri require ought to be abandoned for this other type of abduction. This also forces the condition that abductive explanations have conjunctive form to be abandoned. Abductive explanations of the new form would be single propositions.

Under these modifications, the presuppositions of negative sentences can be explained very well in terms of abduction. If the ξ rules were to be described out of the context of the tableaux in which they originate, they might be formulated as follows:

ξ^*-rule

$\neg\alpha^{\beta}$ abduce β (within the branch)

where abduction could be described in terms of the new conditions.

Abduction provides a preferred interpretation whenever there is an information vacuum. When there is some information concerning the presupposition (either β or $\neg\beta$), the preference is not required to operate. In other cases, given two possibilities of extending a given branch (one with β and one with $\neg\beta$), the alternative with β is preferred.

5.5.3 Presupposition of Positive Sentences

The problem with this interpretation is that the presuppositions of positive sentences do not operate in the same way. Instead of being similar to abduction, the presuppositions of positive sentences behave much more like deduction. This has given rise to all the approaches in the literature that consider the presuppositions of positive sentences as entailments of the sentence.

It is possible to consider that one and the other form of presupposition are completely different forms of inference, or as having different behaviour. But there are reasons in favour of considering a unified conceptual description of the two behaviours. On one hand, the differences are introduced only by negating the original presuppositional sentence, and accepting a totally different explanation for each of the behaviours would impose a strong role on negation as a trigger for the transition from one to the other. On the other hand, disjunctive statements may have to 'presuppose' a proposition in one or the other specific form even if they involve consideration of one alternative where the presuppositional sentence is negated and another alternative where it is not.

For these reasons, the following attempt to relate the presuppositions of positive sentences and abduction as described for the presuppositions of negative sentences was carried out.

The σ rule is not governed by only one of the two constraints that drive the ξ: that concerning redundancy. For this reason, it cannot be classed directly as the same kind of abduction. The form of reasoning that results from σ rules is more assertive than traditional abduction. Yet it behaves in a similar way with respect to being blocked when the information obtained is already present in the context.

An analysis of this rule has to be carried out in a wider setting that includes some pragmatic considerations. Assuming presupposition is understood as a linguistic instruction to carry out abduction of a certain information in the given context, presuppositions of negative sentences do this in strict terms, providing new information only if it is consistent with the context. Presuppositions of positive sentences, as has been considered thoroughly in chapter 4, do more than that, and they can actually make a given context inconsistent. If they are interpreted in terms of abduction, presuppositions of positive sentences would be abductive conjectures that can actually override previous information.

Comparing with Cialdea and Pirri's method, this interpretation would correspond to dropping the condition on abductive solutions that they be consistent with the context. At this stage, all of the conditions on abductive solutions have been rejected. This may seem a radical departure from the proposed formulation of abduction. However, it is important to note that the conditions that have to be abandoned are only those that described which abductive solutions are considered interesting. The solutions obtained without the conditions maybe not be interesting for the purposes of abduction as a logical operation, but they qualify as abductive solutions nonetheless.

If these considerations are accepted, the rules for positive sentences could be phrased in a similar way to those for negative sentences:

σ^*-rule

α^β abduce β (within the branch)

where abduction in this case obeys only the condition that after the addition β be a logical consequence of the context (even if the context becomes inconsistent in doing so).

This analysis results in two different kinds of abduction: a weak abduction (that of presuppositions of negative sentences), and a strong abduction (that of presuppositions of positive sentences).

5.5.4 Joint Effect

Each presupposition (by Coverage Property) will spring from both α^β and $\neg\alpha^\beta$ in different branches. Abduction may have to be considered locally in each branch in the same way that defaults were computed locally with respect to the information in the branch.

Given the whole tree and assuming the presuppositional sentence A^B appears somewhere in the tableau, there are several possible cases depending on whether β is in the different branches.

- B does not appear in the tree
- B appears in all branches
- $\neg B$ appears in all branches
- B appears in some branches and $\neg B$ appears in (all) others

An overall result for these local operations can be obtained along the lines given in chapter 4 for presuppositional consequence. As in the case of presuppositional consequence with respect to defaults, the overall behaviour is a mixture between the two alternatives that make it up. In general terms, it can behave as strong abduction if no negative presuppositional sentences play a role in it, but it will behave as weak abduction otherwise.

This abductive interpretation of presupposition may lead to a compact formulation in which the view of presupposition in terms of satisfaction and the view of presupposition in terms of inference would be unified.

5.5.5 Abduction and Presuppositions that are Conditionals

The form of abduction considered above retained a condition on possible abductive solutions so that only the actual presuppositions of presuppositional sentences in the context could be considered as abductive explanations. If this condition is abandoned, a whole range of new solutions is available.

The interpretation of presupposition as unrestricted abduction allows an interesting consideration regarding interpretation of sentences like

> (70.c) *If Mary has had a bath, then Bill regrets that there is no hot water left.*

If C^B is taken as a request to 'explain B' then a sentence of the form $B \to C^B$ explains itself wherever necessary. This corresponds to examples like

> (71) *If there is a typewriter then the typewriter is blue.*

However, the case of (70.c), transcribed as $A \to C^B$, allows two possibilities: interpretation may consist either of explaining B by adding B to the whole context (this is the case where the sentence does presuppose that there is no hot water left) or explain B by adding $A \to B$, from which addition B follows in the context because A is already present (an interpretation in which the context is enriched with the information

> (111) *If Mary has had a bath, then there is no hot water left.*

This second interpretation is the one that suggests the existence of presuppositions that have the form of conditionals. The abductive account of presupposition shows how two interpretations are possible. How language users select one or the other interpretation in actual language use still has to be studied.

5.5.6 Abduction and Earlier Accounts of Presupposition

Accommodation and Abduction

Heim's definition of accommodation may be represented by the following schema:

Linguistic information indicates context should satisfy P

Check for P in context

P fails

Adding P to the context would make context satisfy P

There are grounds for adding P .

This schema can easily be related to Peirce's basic schema for abduction.

The surprising fact C is observed.

But if A were true, C would be a matter of course.

Hence there is reason to suspect that A is true.

It may seem excessive to describe accommodation in terms of abduction, on the grounds that no logical operation is involved in accommodation as described by Heim. However, abduction has the properties needed to represent with one single operation the dual nature of presupposition as a test and as an informative contribution.

This view of accommodation as abduction also allows clear criteria for when accommodation is restricted to some local contexts and blocked in others (whenever it is inconsistent).

Default Logic and Abduction

Presuppositions of negative sentences can only be considered as inferences if they are considered as defeasible inferences. The defeasible status of these presuppositions can be interpreted as equivalent to the tentative status traditionally associated with explanations obtained through abduction. When such tentative explanations are confronted with inconsistencies with information obtained previously or simultaneously, they are likely to be abandoned.

Presuppositional sentences can be seen as sentences that carry in their linguistic form an indication that a certain abduction may be carried out. Presupposition can be seen as a linguistic way of marking a certain abductive path as preferred, but with no actual commitment to that path. The linguistically annotated abduction is only a suggestion. This would fit the descriptions provided by Mercer of presuppositions in terms of defaults. The problematic examples that Mercer's framework is designed to tackle constitute cases where previous or simultaneous information make this abduction invalid.

5.6 Conclusions

The presuppositional tableau framework provides a good formalization of the concept of context. Tableaux are constructed incrementally, and all the information of previous utterances is automatically stored and used in the interpretation of subsequent ones.

Addition of presuppositions of positive sentences to a language can be interpreted as imposing a set of presuppositional axioms on the logic. This addition does affect the logical consequences of particular sentences of the language, but the nature of the logic itself as a logical calculus is not drastically affected. This is due to the fact that the PP axioms have a very simple form and each one only has the effect of linking together the truth-values for two given atomic propositions. On the other hand, addition of presuppositions of negative sentences must be interpreted as imposing a preference on the models given as logically possible by the interpretation of negated presuppositional sentences. This addition does affect the nature of the logical calculus.

The fact that repair operations are not allowed over the system makes the consequence relation defined by the presuppositional tableaux for discourses monotonic.

Chapter 6

A Preliminary Study of Presuppositions in First Order Logic: Proof Theory

6.1 Introduction

At this stage it is important to introduce a distinction between two different kinds of presupposition. One the one hand, presuppositions of definite descriptions, such as the presuppositions of

(3.a) The typewriter is broken.

(that there is a typewriter) or

(3.b) Sam broke his typewriter.

(3.c) Sam's typewriter is broken.

(that Sam has a typewriter). When transcribing natural language examples into predicate logic, these presuppositions take the form of conditions on the terms used to represent the corresponding noun phrases (that the objects they refer to exist in the domain under consideration). Section 6.2 provides a formal notation for these conditions. This formalization allows a specific treatment of the presuppositions that makes formally explicit their interaction with quantification. This is discussed in section 6.3.

On the other hand, presuppositions of other types (as in examples (3.d) to (3.n), for instance), which require a representation in terms of relations between propositions even in a predicate logic framework. These presuppositions must be dealt with along the same lines as in the propositional case, making the corresponding allowances for the different concept of 'proposition'. The interaction between these two types of presupposition is discussed in section 6.4.

6.1.1 Overview

This chapter focuses on the role that information about the domain of predication plays in the interpretation of sentences. Although the study is aimed at clarifying the interpretation of presuppositions of existence, additional issues such as quantification are shown to be affected. The formalism obtained is not intended as a useful application in its own right, but simply as a case study on the problems of interaction between quantification and existence of referents in the domain. This requires that it be constrained to the simplest formalism that can represent the necessary elementary issues. As such, the formalisations of presuppositions that it provides are restricted to a few particular types of presupposition, and even those can only be represented in a crude fashion. However, the study does bring to the surface interactions between the different phenomena involved (existence predicates, presupposition, quantification) that had not been specifically addressed before.

The idea is to define a representation language that refers to a set of models with domains that may be different from one model to another. The information about existence of a given object in the domain can be reflected by additional existence predicates, that are true iff the object is in the domain. Truth for this language is defined for a proposition in a possible world in terms of a) existence (or non-existence) of the necessary constants in the domain of the possible world, and b) relation between the existing constants as given by the extension of the corresponding predicate in that possible world. A presumption of truth for asserted propositions constrains the domain under consideration, in as much as it extends to a presumption of existence of the objects referred by the terms used as arguments.

The idea behind this system is to model the way in which simple mention of an object as argument of a predicate conveys the 'presupposition' that it exists in the domain and does not need to rely on the presence of an existence predicate.

6.1.2 Previous Work

In the realm of modal logic, Kripke [22], Hughes and Creswell [17], and Fitting [6] consider different possible world semantics that account for different worlds having different domains. But they do not attempt to represent in their system the presupposition of existence associated with a zero order predicate sentence: a sentence of the form $P(a)$ 'presupposes' that there is some object a in the domain. It is this issue that this chapter sets out to address.

The main problem in attempting a first order approach is dealing with quantification. Lejewski [24] presents a formalism in which quantifiers are allowed to scope over both existing and non-existing objects. Hirst [15] reviews work in philosophy and logic on this topic. Marcu [26] proposes a formalism that can represent several of the types of existence that Hirst lists.

These approaches provide very powerful formalisms for extending the power of first order logic to deal with the problem of existence. The present framework does not aspire to compete with them in the same category. In the same vein as the study carried out for the propositional case, the aim is to consider the subset of first order logic that can be said to mirror the structure of natural language in a close manner, and consider what formal implications the introduction of presupposition has on this subset of the logic.

In this case, the results obtained give some insights on the intrinsic ambiguity of quantified sentences in natural language.

6.1.3 The Importance of Domains

The key question is to establish the difference between two kinds of domain: what can be talked about (what the language has words for) and what actually exists. Classical logic traditionally assumes that one can talk about what exists, thereby fusing these two concepts of domain together. There are actually many different such concepts of domain (what existed in the past, what exists in fiction . . . ; see Hirst [15] for a review of these) but here I will be concerned only with these two basic ones. In what follows, I assume a global domain D of objects that can be talked about, and a set of local domains D_i, where each D_i is a subset of D, comprising all the objects that exist in a given model.

In order to capture the behaviour that has been described in the examples given for the propositional case, I am committed to the assumption that a sentence *about* an object that does not *exist* in the local domain must be false in that domain. This is the behaviour that underlies the examples of defeasible presupposition such as

(112) *If John has no children, then John's children didn't come to the party.*

6.1.4 The Problems with Quantification

The first question that has to be addressed is whether quantifiers have scope over what one can talk about (D) or over what exists (D_i). This is the main issue addressed in Lejewsky's work. He defines two different modes of quantification: restricted quantification (where one quantifies only over what exists) and unrestricted (where one can quantify over everything). The second option relies on specifying when something does exist.

The present work is concerned with a more pragmatic issue. Lejewsky's alternatives assume that, whatever the choice of what to quantify over, the information about what actually exists is available (or has to be provided). In a communication situation in natural language this is not normally the case. In a everyday conversation the information about what actually exists is not necessarily available to participants. Even if it is crystal clear to one participant what he is taking to actually exist, he has no means of knowing that the other participants share this information with him. Neither do they make this information explicit during conversation. Instead, it has to be cobbled together from occasional statements about existence and, in most cases, from the presuppositions of existence of the utterances used in the conversation. In a given context, some objects may have been asserted to exist, some may have been presupposed to exist, and some may have been asserted not to exist. A speaker putting together a quantified statement may have a fair idea of which objects he is quantifying over. The listener that has to interpret the quantified statement has incomplete information and is faced with a choice of possible interpretations.

Language users seem to cope with this situation in simple ways. On hearing a statement like

(113) *Every man drinks*

a hearer is faced with the question of how to interpret the quantifier. Should it range over all men, over the men that have been mentioned so far, or all the men except those that the listener has been explicitly asked not to include? Intuition suggests a pragmatic interpretation of the form

(114.a) *Every man that I'm aware of drinks.*

Or maybe

(114.b) *Every man relevant to the present issue drinks.*

In most cases, if faced with a choice between assuming the quantification to range over everything that has been accepted to be in the domain or to range over everything that has not been explicitly excluded from the domain, language users seem to prefer the first option. On the other hand, a statement like

(115) *Some man drinks*

shows different preferences. Faced with the same choice, the second option seems to capture better the idea that the quantifier conveys. After all, this type of quantified sentence seems to work well as means of introducing a given man to the domain. This difference of behaviour of the quantifiers with respect to domain information has to be explained.

6.2 Zero Order Predicate Logic

If the language is provided with a specific predicate to convey existence of its argument in the domain (for instance $\varepsilon(x)$), a zero order logic for a language with presupposition would be equivalent to the propositional case. Every proposition of the form $P(x)$ would simply have to be transcribed as $P(x)^{\varepsilon(x)}$. However, this section provides a syntactic reformulation that has the advantage of making the information about local domains explicit in the object language. Over the resulting framework, the first order logic is more intuitive to study.

6.2.1 The Language

L^0 Representation Language

Information will be represented by a language L^0.

Definition 5 *The alphabet of L^0 consists of the non-descriptive symbols of L together with descriptive symbols:*

- *a set C of sorted individual constants $a_1, \ldots a_n \in C_a, b_1, \ldots b_n \in C_b, \ldots$ ($C = \bigcup C_i$)*
- *a set R of n-place predicate constants*
- *one existence predicate constant ε*
- *a set F of n-place operation constants*

Definition 6 *t is an individual term in a language L^0 if and only if t is a (finite) symbol string and either*
1) t is an individual constant in C, or
2) $t = f(t_1, t_2, \ldots t_n)$ where $t_1, t_2, \ldots t_n$ are individual terms of L^0 and f is an n-place operation constant in F.

Definition 7 *A is an atomic formula of L^0 if and only if either:*
1) A consists of an n-place predicate constant in R followed by n individual terms of L^0, or
2) A consists of the ε existence predicate constant of L^0 followed by an individual term of L^0.

The inductive definition of formula in L^0 is analogous to the inductive definition of formula in L.

For the present section, it is assumed that there are no previously defined relations of presupposition (in the sense of chapters 3 and 4) between the propositions of L^0.

Mapping Natural Language onto L^0

A domain of sorted constants is required to capture the structure of natural language sentences without resorting to translating expressions like *the man*, which are not of sentence type, as $man(x)$, which is of proposition type. The introduction of sorts into the language allows *the man* to be translated as a constant $m_1 \in M$, where M is the sort of men. The complexity of the logic is not affected. This policy of maintaining a structure in the logical representation as close as possible to the linguistic structure is followed throughout this study. This policy is motivated by the conviction that it should be the actual linguistic structure of sentences that determines the compositional behaviour of presuppositions, so it should be retained as syntactically explicit information for the proof theory to operate on.

A mapping is assumed from nouns into constants of the logic of a given sort. Noun phrases of the form *noun of noun (f of t)* or *noun's noun (t's f)* are mapped in to terms of the form $f(t)$ were f is a function corresponding to the constructions *noun of* or *'s noun*, and t is a term. Declarative sentences with a structure *subject verb (s P)* are mapped onto propositions $P(s)$. Declarative sentences with a structure *subject verb object (s P o)* are mapped onto propositions $P(s, o)$.

6.2.2 Tableau Proof Theory

The issue of information about the domains must be addressed by the proof theory.

A formula of L^0 is *simple* if it is an atomic formula of L^0 or the negation of an atomic formula of L^0.

With respect to the semantics outlined above, a formula of L^0 contains three types of information:

- the actual formula
- information about what exists
- information about what does not exist

In order to make use of all this information, it is useful to make it explicit in the proof theory. For this purpose, an extension of the tableaux used in the previous chapter is presented here. In this extension, the three types of information are set out as three separate columns within a branch. Each simple formula of L^0 is expanded into an extended segment of branch of this form.

$$\overbrace{\begin{array}{|c|c|c|} Ex & Nex & \text{Formula} \end{array}}$$

The Ex column lists all the terms corresponding to objects that exist.

The Nex column lists all the terms corresponding to objects that do not exist.

The final column presents the formula from which this information is extracted.

For a given branch Δ of a tableau, a set of terms $EX(\Delta)$ can be defined as the union of all the Ex columns that appear along the branch, and a set $NEX(\Delta)$ as the union of all the Nex columns that appear along the branch.

This domain information is relevant for purposes of computation as given by the following extension of the definition of branch closure.

A branch Δ of an L^0 tableau is closed if:

- there are formulas ϕ and $\neg\phi$ in the branch, or
- $EX(\Delta) \cap NEX(\Delta) \neq \emptyset$

An L^0 tableau is closed iff all its branches are closed.

The expansion of the domain information from a given formula is carried out by applying the following rules:

ε rules

$$\overbrace{\begin{array}{|c|c|c|} t & & \varepsilon(t) \end{array}}^{\varepsilon(t)} \qquad \overbrace{\begin{array}{|c|c|c|} & t & \neg\varepsilon(t) \end{array}}^{\neg\varepsilon(t)}$$

η rules

(for formulas)

$$\overbrace{\begin{array}{|c|c|c|} s_1 & & \phi \\ \vdots & & \\ s_m & & \end{array}}^{\phi}$$

where s_i are all the terms in ϕ

(for terms)

$$\begin{array}{|c|c|c|} f(t) & & \\ t & & \end{array}$$

where $f(t)$ was already in the column and t is its expansion

ν rules

(for formulas)

$$\overbrace{\begin{array}{|c|c|c|} s'_1 & & \neg\phi \\ \vdots & & \\ s'_p & & \end{array}}^{\neg\phi}$$

where s'_i are all the terms in ϕ

(for terms)

$$\begin{array}{|c|c|c|} & f(t) & \\ t' & & \end{array}$$

where $f(t)$ was already in the column and t' is its expansion

An L^0 tableau is constructed by using rules α, β, ε, η and ν rules as follows:

- expand using all the rules (mark all additions resulting from ν rules, if a formula or term is marked, mark its expansion as well)
- once no more rules are applicable, retract those marked additions whose expansions contribute to the closure of a branch

Simple sentences are easily represented:

(116) *France cherishes its king.*

$$\overbrace{\begin{array}{|c|c|c|} f & & C(f,k(f)) \\ k(f) & & \\ f & & \end{array}}^{\boldsymbol{C(f,k(f))}}$$

Those terms used in the sentence ($k(f)$ and f) are updated as domain information into the EX column.

Existence statements involve the NEX column when they are negated:

(117) *There is no king of France.*

$$\overbrace{\begin{array}{|c|c|c|} f & k(f) & \neg\varepsilon(k(f)) \end{array}}^{\boldsymbol{\neg\varepsilon(k(f))}}$$

Sentence

(69) *If the typewriter is blue then Sue will be happy*

corresponds to the following representation.

$$\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{B(a) \to H(s)}} \\
\hline
\begin{array}{r} \neg B(a) \\ H(s) \end{array} & \begin{array}{r} \neg B(a) \\ \neg H(s) \end{array} & \begin{array}{r} B(a) \\ H(s) \end{array} \\
\overbrace{\begin{array}{|c|c|r|} a & & \neg B(a) \\ s & & H(s) \end{array}} & \overbrace{\begin{array}{|c|c|r|} a & & \neg B(a) \\ s & & \neg H(s) \end{array}} & \overbrace{\begin{array}{|c|c|r|} a & & B(a) \\ s & & H(s) \end{array}}
\end{array}$$

The three branches of this tableau have local domains $EX(1), EX(2), EX(3)$, where $EX(1) = \{a, s\}$ and $EX(1) = EX(2) = EX(3)$.

Sentence

(71) *If there is a typewriter then the typewriter is blue*

corresponds to the following representation.

$$\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{\varepsilon(a) \to B(a)}} \\
\hline
\begin{array}{r} \neg\varepsilon(a) \\ B(a) \end{array} & \begin{array}{r} \neg\varepsilon(a) \\ \neg B(a) \end{array} & \begin{array}{r} \varepsilon(a) \\ B(a) \end{array} \\
\overbrace{\begin{array}{|c|c|r|} & a & \neg\varepsilon(a) \\ a & & B(a) \\ \hline\hline \end{array}} & \overbrace{\begin{array}{|c|c|r|} & a & \neg\varepsilon(a) \\ & & \neg B(a) \end{array}} & \overbrace{\begin{array}{|c|c|r|} a & & \varepsilon(a) \\ a & & B(a) \end{array}}
\end{array}$$

In this case:

- the first branch is closed ($EX(1) \cap NEX(1) = \{a\}$)
- $EX(2) = \emptyset$ and $NEX(2) = \{a\}$

- $EX(3) = \{a\}$ and $NEX(3) = \emptyset$.

The tableau does not presuppose the existence of a.

The interaction between negation and presupposition is apparent in the following example.

> (118) *If there is no king of France then France does not cherish its king.*

This sentence is represented as follows:

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{\neg\varepsilon(k(f)) \rightarrow \neg C(f, k(f)}} \\
\hline
\varepsilon(k(f)) & \varepsilon(k(f)) & \neg\varepsilon(k(f)) \\
\neg C(f, k(f)) & C(f, k(f)) & \neg C(f, k(f)) \\
\overbrace{\begin{array}{|c|c|r|}
k(f) & & \varepsilon(k(f)) \\
f & & \\
f & & \neg C(f, k(f)) \\
k(f) & & \\
f & &
\end{array}}
&
\overbrace{\begin{array}{|c|c|r|}
k(f) & & \varepsilon(k(f)) \\
f & & \\
f & & C(f, k(f)) \\
k(f) & & \\
f & &
\end{array}}
&
\overbrace{\begin{array}{|c|c|r|}
f & k(f) & \neg\varepsilon(k(f)) \\
f & & \neg C(f, k(f))
\end{array}}
\end{array}
$$

In this case, the proposition $\neg C(f, k(f))$ in the last column only gives rise to part of the expansion that it could have done. This is clearly apparent if its expansion is compared to that of the same expression in the first branch of the same tableau. The different local contexts of each branch determine different expansions for the same expression[1].

Similar predictions concern example

> (119) *If Buganda does not exist then the king of Buganda did not open the exhibition.*

where the corresponding relation involves an additional step akin to the treatment of higher order presuppositions in chapter 4. The sentence is represented as:

[1] Although it is not apparent from the representation as given her, it is important to remember, that the construction method relies on expanding all the possible information and then retracting the unnecessary information.

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{\neg\varepsilon(b) \rightarrow \neg O(e, k(b))}} \\
\hline
\begin{array}{r} \varepsilon(b) \\ \neg O(e,k(b)) \end{array} & \begin{array}{r} \varepsilon(b) \\ O(e,k(b)) \end{array} & \begin{array}{r} \neg\varepsilon(b) \\ \neg O(e,k(b)) \end{array} \\
\overbrace{\begin{array}{|c|c|r|} b & & \varepsilon(b) \\ e & & \neg O(e,k(b)) \\ k(b) & & \\ b & & \end{array}} & \overbrace{\begin{array}{|c|c|r|} b & & \varepsilon(b) \\ e & & O(e,k(b)) \\ k(b) & & \\ b & & \end{array}} & \overbrace{\begin{array}{|c|c|r|} & b & \neg\varepsilon(b) \\ e & & \neg O(e,k(b)) \end{array}}
\end{array}
$$

It can be shown that this way of applying the rules corresponds to treating ν-rules as default rules and defining an extension of the tableau (as built with α-, β-, ε- and η-rules) in terms of a fix point operator.

The need to retract can be minimised by applying ν rules last wherever possible.

The usual concepts of refutation, proof, validity, inconsistency and consequence as defined for tableaux in terms of closure apply to this proof theory.

For instance, for sentence

(118) *If there is no king of France then France does not cherish its king.*

given above, it is interesting to note that the sentence is considered a valid sentence by the framework. The validity of the same sentence can be tested using the tableau method:

$$
\begin{array}{c}
\boldsymbol{\neg(\neg\varepsilon(k(f)) \rightarrow \neg C(f, k(f))} \\
\hline
\neg\varepsilon(k(f)) \\
C(f,k(f)) \\
\overbrace{\begin{array}{|c|c|r|} f & k(f) & \neg\varepsilon(k(f)) \\ f & & C(f,k(f)) \\ k(f) & & \\ \hline\hline \end{array}}
\end{array}
$$

The sentence

(119) *If Buganda does not exist then the king of Buganda did not open the exhibition.*

not surprisingly, shows the same kind of validity:

$$\begin{array}{c} \underline{\boldsymbol{\neg(\neg\varepsilon(b) \rightarrow \neg O(e,k(b)))}} \\ \neg\varepsilon(b) \\ O(e,k(b)) \\ \overbrace{\begin{array}{|c|c|r|} & b & \neg\varepsilon(b) \\ e & & O(e,k(b)) \\ k(b) & & \\ b & & \\ \hline\hline \end{array}} \end{array}$$

6.2.3 Pragmatic Concepts

In this formalism, the concept of presupposition becomes blurred. Objects may come to be in EX as a result of either a 'presupposition' (in the traditional sense) of a sentence where the object appears as an argument (*The king of France is bald* 'presupposes' $k(f) \in EX$), or as a result of an assertion of an existence predicate with the object as an argument (*There is a king of France* forces $k(f) \in EX$). The framework does not distinguish between these two processes in terms of the information that it stores about the object. This is equivalent to the way in which the propositional formalism represented asserted and presuppositional information in a homogeneous way.

Another peculiarity of the system that contributes to this blurring of the concept of presupposition is the fact that existence predicates involving functions 'presuppose' the existence of the arguments of those functions (*There is a king of France* forces $k(f) \in EX$, but it 'presupposes' $f \in EX$). This is intuitive in the sense that the existence predicate forces the introduction of its main argument, but its assertive power does not extend to any other terms being used within it (as arguments of a function). As a result, any such terms are presupposed just as much as the arguments of a predicate.

Addition of a formula ϕ to a branch Δ is *logically redundant* if :

- there is already some formula ϕ in the branch, or
- ϕ is an existence statement $\varepsilon(t)$ and $t \in EX(\Delta)$ (this may occur without $\varepsilon(t)$ being in Δ when t has been 'presupposed')

Addition of a term t to the Ex column of a branch Δ is *presuppositionally redundant* if :

- it is not the result of an expansion of a formula of the form $\varepsilon(t)$, and
- $t \in EX(\Delta)$ already

6.3 First Order Logic

First order logic is obtained from zero order logic by allowing quantification over terms.

6.3.1 The Language

L^1 Language

Information will be represented by a language L^1.

Definition 8 *The alphabet of L^1 consists of the non-descriptive symbols of L^0 together with:*

- *individual sorted variables (only used bound) x,y,z*
- *quantifiers $\forall, \exists$*

L^1 has the same descriptive symbols as L^0.

The definitions for individual term and atomic formula for L^1 are the same as for L^0.

Definition 9 *Inductive definition of formula in L^1 :*
1) an atomic formula in L^1 is a formula in L^1
2) If A and B are formulas in L^1, then so are $(A \wedge B)$, $(A \vee B)$, and $(A \rightarrow B)$.
3) If A is a formula in L^1, then so are $\forall x A^*$ and $\exists x A^*$, where A^* is A or obtained from A by replacing occurrences of an individual constant with the variable x.

Mapping Natural Language onto L^1

This preliminary study is intended to point out some of the issues in quantification that are affected by the consideration of presupposition and existence predicates in the definition of domains. In order to keep the discussion simple, only a very basic form of quantification is considered.

The constructions *every*, *some*, and *no* are mapped onto $\forall$, $\exists$ and $\neg\exists$ respectively. This is not intended as a claim of semantic equivalence between these constructions, but simply as a working hypothesis on which to base the study, in the same vein as in the propositional case.

The introduction of sorts also simplifies the treatment of quantifiers, since quantification must then range only over the sort to which the quantified variable belongs. It also allows sentences like

(120) *Every man works*

to be represented as $\forall m W(m)$ (for some sorted variable m belonging to the sort of 'men'), without having to resort to the construction $\forall x M(x) \rightarrow W(x)$, which introduces additional logical connectives that were not linguistically explicit.

6.3.2 Quantification and Proof Theory

Quantification and Sorted Domains

The domains that are considered range over constants that are sorted. This implies that quantification is restricted to the sort to which the variable belongs. For simplicity of notation and argumentation, this is taken for granted in what follows.

The Free Variable Approach

In order to obtain a more elegant formulation, Fitting's [6] free variable tableaux formulation is preferred over other formulations of the tableaux rules for quantifiers. This allows the issues related with domains to be considered only when attempting to find substitutions to close a tableau, and allows the actual tableaux rules for quantifiers to be independent of the choice of domain over which to quantify.

Instead of choosing a specific instantiation t for applications of γ rule, $\gamma([x])$ is added, where $[x]$ is a new free variable[2]. This free variable can be later instantiated. This allows an efficient choice of instantiation: by considering only instantiations that close the tableau, the search for instantiations is restricted to those that might constitute counterexamples.

This approach creates problems in the application of the δ rule: in order to ensure that the instantiations for δ rules are new, Skolem functions $f(x_1, \ldots, x_n)$ are used instead of parameters (where $x_1, \ldots, x_n$ are all the free variables occurring on that branch). Whatever value is assigned to $x_1, \ldots, x_n$ during unification, $f(x_1, \ldots, x_n)$ must be different from any of them. If the δ rule is applied before any γ rule has been applied, the required Skolem function has no variables. These zero place functions are equivalent to simple constants.

The proof language must be extended with the following:

Definition 10 *Let $L^1 = L(R, F, C)$ be a first-order language. Also let sko be a countable set of function symbols not in F (called* Skolem function symbols*), including infinitely many 1-place, infinitely many 2-place, and so on. By L^{sko} we mean the first-order language $L(R, F \cup sko, C)$.*

Substitution

The operation of instantiating a free variable in a tableau can be defined formally using the concept of substitution. A *substitution* is a mapping $\sigma : V \to T$ from the set of variables V to the set of terms T.

The definition of substitution is extended to all terms:

- $c\sigma = c$ for a constant symbol c
- $[f(t_1, \ldots, t_n)]\sigma = f(t_1\sigma, \ldots, t_n\sigma)$ for an n-place function symbol f.

Let σ be a substitution. By σ_x is meant the substitution that is like σ except that it doesn't change the variable x. For any variable y,

[2]The brackets are used to indicate that it is free and distinguish it from bound variables. Because the formalism operates independently on domain information, it is not always possible to rely on the context to determine whether a variable is free or bound, so this notational device is introduced. The conceptual complexity of the language is not affected.

$$y\sigma_x = \begin{cases} y\sigma & \text{if} \quad y \neq x \\ x & \text{if} \quad y = x \end{cases}$$

Substitutions are extended to all formulas.

- $[A(t_1, \ldots, t_n)]\sigma = A(t_1\sigma, \ldots, t_n\sigma)$
- $[\varepsilon(t)]\sigma = \varepsilon(t\sigma)$
- $[\neg X]\sigma = \neg[X\sigma]$
- $(X \circ Y)\sigma = (X\sigma \circ Y\sigma]$
- $[\forall x\Phi]\sigma = \forall x[\Phi\sigma_x]$
- $[\exists x\Phi]\sigma = \exists x[\Phi\sigma_x]$

A substitution is free for a formula:

- if A is atomic, σ is free for A
- σ is free for $\neg X$ if σ is free for X
- σ is free for $(X \circ Y)$ if σ is free for X and σ is free for Y
- σ is free for $\forall x\Phi$ and $\exists x\Phi$ if: σ_x is free for Φ and if $[y]$ is a free variable of Φ other than x, $[y]\sigma$ does not contain x.

Substitution is extended to tableau.

Let σ be a substitution and T be a tableau. We extend the action of σ to T by setting $T\sigma$ to be the result of replacing each formula X in T by $X\sigma$. Then σ is free for T if σ is free for every formula in T.

6.3.3 Quantification and Domain Information

The zero order tableaux presented so far represent contexts where there is no completely defined domain over which to quantify. At most the constraints established by EX and NEX are all the information that is available concerning the domain. The positive and negative restrictions on domains given by a context, EX and NEX, can be used as guidelines as to what the behaviour of quantifiers could be. There are two possible definitions of quantification according to that information.

The presence of the domain information in the object language allows modelling of quantification as it might be considered by a hearer interpreting the sentences with partial information only. All the information that the hearer has is that represented in the present framework by T (the set of all terms of the language), EX, and NEX.

This information influences the proof theory through the concept of substitution. This can be taken into account formally by considering the definition of substitution as different kinds of mapping. For a given branch, the following are possible alternative definitions of branch substitution:

- $\sigma^* : V \rightarrow T$,
- $\sigma^* : V \rightarrow EX$ or
- $\sigma^* : V \rightarrow T \backslash NEX$.

The first one results in traditional quantification (over all the objects that can be talked about). The other two I refer to as *cautious quantification* and *optimistic quantification.* Because they constitute a form of quantifying strongly dependent on context, I refer to them collectively as *contextual modes of quantification.*

Optimistic quantification will consider too many objects in most cases (objects in D that are not in the 'intended domain').

For example, an utterance of

(121) *Every window is closed*

is not normally intended to range over all the existing windows, but rather to range over the subset of these windows that can be considered salient in the given context.

Clark [4] studies the way language users trace the relations between occurrences of referring words and the clues earlier on in the discourse that allow them to locate the intended referent. He refers to this operation as *bridging*. Of all the varieties of bridging that he considers (direct reference, indirect reference by association, indirect reference by characterization, reasons, causes, consequences, and concurrences) the present work only deals with the simpler cases of direct reference. A practical treatment would have to consider all of them, together with extra information provided by sensorial channels other than strictly linguistic.

Cautious quantification will consider too few objects in most cases (objects in D that are in the 'intended domain' will not be considered). This is due to the fact that, using Clark's terminology, only bridging by direct reference is being considered, so objects that should be available by other varieties of bridging may be left out. But the present simplification allows adequate modelling of the basic underlying intuitions, and this is considered sufficient for the intended preliminary study.

If the contextual modes of quantification are allowed, the extension of substitutions to a tableau is no longer trivial. Each branch of a tableau may have different EX and different NEX, and so constrain differently the same substitution.

A substitution σ is *cautiously applicable* to a branch Δ of quantified tableau T iff for all the terms a that σ maps into, $a \in EX(\Delta)$.

A substitution σ is *optimistically applicable* to a branch Δ of quantified tableau T iff for all the terms a that σ maps into, $a \notin NEX(\Delta)$.

A substitution σ is *applicable* to a cautiously/optimistically quantified tableau T iff σ is cautiously/optimistically to all the branches of T.

In the rest of this preliminary study, cautious quantification is assumed. This choice is not meant to imply an utter rejection of optimistic quantification. A firm decision on this issue would have to consider in detail all the different forms of quantification available in natural language and would be beyond the scope of the present work.

Given the choice of contextual quantification, quantifiers cancel the presupposition of existence of objects. Because a quantified formula $\forall x E(x)$ can only refer to objects that have already been mentioned in the domain, it cannot introduce any object. Any a appearing as a result of instantiating a universally quantified statement must be an a that was already present in the domain.

6.3.4 Proof Rules for Quantification

The expansion of quantified statements is carried out by applying the following rules:

γ rules

$$\frac{\forall x \gamma(x)}{\gamma([x])} \qquad \frac{\neg\exists x \gamma(x)}{\neg\gamma([x])}$$

for an unbound variable $[x]$ for an unbound variable $[x]$

δ rules

$$\frac{\exists x \delta(x)}{\delta(h_0([x1], \ldots [xn]))} \qquad \frac{\neg\forall x \delta(x)}{\neg\delta(h_0([x1], \ldots [xn]))}$$

for h_0 new and $[x_1], \ldots, [x_n]$ all the free variables that appear in $EX(\Delta)$).

Free variables and Skolem functions are subject to the same domain rules as ordinary terms. However, in order to make the notation more easily understandable, free variables are added to the domain information between square brackets to distinguish them from other terms, and Skolem functions and Skolem constants are given a subscript.

The introduction of different alternative domains complicates the issue of instantiating the free variables that result from these rules, but the free variable version of the rules allows this to be handled independently of the rules.

The role played by instantiation of free variables in the proof theory is formally captured by a rule on the use of substitutions to obtain additional (instantiated) tableaux.

> **Tableau Substitution Rule:**
>
> If T is a tableau for the set S of sentences and the substitution σ is free for T and σ is applicable to T, then $T\sigma$ is also a tableau for S .

A proof for X is a closed tableau for $\neg X$ constructed using the propositional rules α and β, the domain rules ε, η and ν, the free variable quantifier rules γ and δ, and the Tableau Substitution Rule, as follows:

- expand using all the rules (mark all additions resulting from ν rules, if a formula or term is marked, mark its expansion as well)
- once no more rules are applicable, retract those marked additions whose expansions contribute to the closure of a branch

The tableaux for discourses are defined in terms of the tableaux for single sentences in the same way as in chapter 4.

The operation of the rules may be understood better if applied to some examples.

The sentence

(122) *Some farmer beats his donkey.*

is represented as:

$$\begin{array}{r}
\boldsymbol{\exists f B(f, d(f))} \\
B(f_0, d(f_0)) \\
\overbrace{\begin{array}{|c|c|c|}
f_0 & & B(f_0, d(f_0)) \\
d(f_0) & & \\
f_0 & &
\end{array}}
\end{array}$$

where f_0 is a Skolem constant.
For the universal quantifier:

(123) *Every nation cherishes its king.*

$$\overbrace{\begin{array}{|c||l|} [n] & C([n],k([n])) \\ k([n]) & \\ [n] & \end{array}}^{\substack{\boldsymbol{\forall n C(n,k(n))} \\ C([n],k([n]))}}$$

The choice of cautious quantification makes it very difficult to interpret sentences such as this one unless some previous context is provided where certain nations are mentioned.

The choice of quantification adopted here is better suited for discourses of the form:

(124) *There are three men. Every man wears a hat.*

The present framework does not provide a treatment for existence statements involving multiple objects. However, it may be assumed that sentences like

$S \equiv$ (*There are three men*)

produce a representation such as:

$$\overbrace{\begin{array}{|l||l|} m1 & S \\ m2 & \\ m3 & \end{array}}^{\boldsymbol{S}}$$

Against such a representation, the discourse above

$(S) \circ (\forall m H(m))$

can be interpreted as:

$$
\begin{array}{r}
\boldsymbol{S} \\
\overbrace{\left|\begin{array}{l} m1 \\ m2 \\ m3 \end{array}\right| \quad \left|\begin{array}{l} S \\ \\ \\ \end{array}\right|} \\
\boldsymbol{\forall m H(m)} \\
H([m]) \\
\overbrace{\left| [m] \right| \quad \left| H([m]) \right|}
\end{array}
$$

where $[m]$ can now be instantiated to any of the three men $m1, m2, m3$.

The operation of the δ rules can be observed if the sentence

(125) *Some man is bald*

is added to the previous discourse, to obtain the discourse

$(S) \circ (\forall m H(m)) \circ (\exists m B(m))$

$$
\begin{array}{r}
\boldsymbol{S} \\
\overbrace{\left|\begin{array}{l} m1 \\ m2 \\ m3 \end{array}\right| \quad \left|\begin{array}{l} S \\ \\ \\ \end{array}\right|} \\
\boldsymbol{\forall m H(m)} \\
H([m]) \\
\overbrace{\left| [m] \right| \quad \left| H([m]) \right|} \\
\boldsymbol{\exists m B(m)} \\
B(m_0[m]) \\
\overbrace{\left| m_0([m]) \right| \quad \left| B(m_0([m])) \right|}
\end{array}
$$

Here it can be seen that the Skolem function used to expand the existential statement is dependent on the free variable used earlier for the universal statement.

The discourse:

(126 *Not every nation cherishes its king. Some nations do not have a king.*

must be understood as a discourse involving an explanation (as described in chapter 4). The discourse would be of the form

$$(\neg\forall n C(n, k(n))) \circ (\exists n \neg\varepsilon(k(n)))$$

corresponding to

$$\exists n \neg\varepsilon(k(n)) \vdash_p \neg\forall n C(n, k(n)).$$

To test this statement in the framework involves testing the discourse

$$(\exists n \neg\varepsilon(k(n))) \circ (\neg(\neg\forall n C(n, k(n))))$$

or

$$(\exists n \neg\varepsilon(k(n))) \circ (\forall n C(n, k(n))):$$

$$\begin{array}{r}
\boldsymbol{\exists n \neg\varepsilon(k(n))} \\
\neg\varepsilon(k(n_0)) \\
\overbrace{\begin{array}{|c|c|c|} n_0 & k(n_0) & \neg\varepsilon(k(n_0)) \\ n_0 & & \end{array}} \\
\boldsymbol{\forall n C(n, k(n))} \\
C([n], k([n])) \\
\overbrace{\begin{array}{|r|c|c|} [n] & & C([n], k([n]))) \\ k([n]) & & \\ [n] & & \\ \hline\hline \end{array}}
\end{array}$$

where the tableau is closed with the instantiation of $k([n])$ to $k(n_0)$.

Quantifiers that have to be interpreted simultaneously in different branches of a tableau may get different domains, even if they originate from the expansion of the same formula. For the discourse:

(127) *John has a typewriter. Bill has a typewriter. If Sue has no typewriter then every typewriter is blue.*

the representation in the framework would be:

$$\overbrace{\left|\begin{array}{c} t(j)\\ j \end{array}\right| \left|\begin{array}{c} \varepsilon(t(j))\\ \end{array}\right|}^{\boldsymbol{\varepsilon(t(j))}}$$

$$\overbrace{\left|\begin{array}{c} t(b)\\ b \end{array}\right| \left|\begin{array}{c} \varepsilon(t(b))\\ \end{array}\right|}^{\boldsymbol{\varepsilon(t(b))}}$$

$$\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{\neg\varepsilon(t(s)) \rightarrow \forall t B(t)}} \\
\hline
\begin{array}{r} \varepsilon(t(s)) \\ \forall t B(t) \\ B([t]) \end{array} & \begin{array}{r} \varepsilon(t(s)) \\ \neg\forall t B(t) \\ \neg B(t_0) \end{array} & \begin{array}{r} \neg\varepsilon(t(s)) \\ \forall t B(t) \\ B([t]) \end{array} \\
\overbrace{\left|\begin{array}{c} t(s)\\ s \\ [t] \end{array}\right| \left|\begin{array}{c} \varepsilon(t(s))\\ \\ B([t]) \end{array}\right|} & \overbrace{\left|\begin{array}{c} t(s)\\ s \\ t_0 \end{array}\right| \left|\begin{array}{c} \varepsilon(t(s))\\ \\ \neg B(t_0) \end{array}\right|} & \overbrace{\left|\begin{array}{c} s\\ [t] \end{array}\right| \begin{array}{c} t(s)\\ \end{array} \left|\begin{array}{r} \neg\varepsilon(t(s))\\ B([t]) \end{array}\right|}
\end{array}$$

In this example, the peculiarities of the proof theory can be seen at work. In one of the branches the universally quantified proposition of the original discourse appears negated and therefore is expanded as existential. In the other two branches, the proposition is quantified universally, but in each case the domain is different. In the first branch the domain of quantification is $\{t(j), t(b), t(s)\}$ and in the last branch the domain of quantification is $\{t(j), t(b)\}$.

6.4 The Predicate and the Propositional Formalism

6.4.1 Zero Order Logic and the Propositional Formalism

The zero order formalism is equivalent to the propositional formalism. This is apparent in those examples that have been treated earlier using the propositional formalism in chapter 4 and the zero order formalism in the present chapter. For ease of reference, they are presented together below.

Sentence

(69) *If the typewriter is blue then Sue will be happy*

corresponds to the following representation in the propositional case

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{b^t \rightarrow h}} \\ \hline \neg b^t & \neg b^t & b^t \\ h & \neg h & h \\ t & t & t \end{array}$$

and to the following one in the first order formalism

$$\begin{array}{rrr} \multicolumn{3}{c}{\boldsymbol{B(a) \rightarrow H(s)}} \\ \hline \neg B(a) & \neg B(a) & B(a) \\ H(s) & \neg H(s) & H(s) \\ \overbrace{\left|\begin{array}{c|c|r} a & & \neg B(a) \\ s & & H(s) \end{array}\right|} & \overbrace{\left|\begin{array}{c|c|r} a & & \neg B(a) \\ s & & \neg H(s) \end{array}\right|} & \overbrace{\left|\begin{array}{c|c|r} a & & B(a) \\ s & & H(s) \end{array}\right|} \end{array}$$

The sentence

(71) *If there is a typewriter then the typewriter is blue*

corresponds to the following representation in the propositional case

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{t \rightarrow b^t}} \\ \hline \neg t & \neg t & t \\ b^t & \neg b^t & b^t \\ \underline{t} & & t \end{array}$$

and the following one in the first order case

$$\begin{array}{rrr} \multicolumn{3}{c}{\boldsymbol{\varepsilon(a) \rightarrow B(a)}} \\ \hline \neg\varepsilon(a) & \neg\varepsilon(a) & \varepsilon(a) \\ B(a) & \neg B(a) & B(a) \\ \overbrace{\left|\begin{array}{c|c|r} & a & \neg\varepsilon(a) \\ a & & B(a) \end{array}\right|} & \overbrace{\left|\begin{array}{c|c|r} & a & \neg\varepsilon(a) \\ & & \neg B(a) \end{array}\right|} & \overbrace{\left|\begin{array}{c|c|r} a & & \varepsilon(a) \\ a & & B(a) \end{array}\right|} \end{array}$$

These examples show that the zero order formalism is but a syntactic rewrite of the propositional case. The Coverage Property can still be seen to hold over these examples.

6.4.2 First Order Logic and the Propositional Formalism

In the case of the first order formalism, as it could have been expected, the analogy with the propositional case breaks down.

For example (124), the representation as given above:

$$
\begin{array}{r}
\overbrace{\begin{array}{|l|c|c|} m1 & & S \\ m2 & & \\ m3 & & \end{array}}^{\boldsymbol{S}} \\
\forall \boldsymbol{m} \boldsymbol{H}(\boldsymbol{m}) \\
H([m]) \\
\overbrace{\begin{array}{|c|c|c|} [m] & & H([m]) \end{array}}
\end{array}
$$

is equivalent to:

$$
\begin{array}{r}
\overbrace{\begin{array}{|l|c|c|} m1 & & S \\ m2 & & \\ m3 & & \end{array}}^{\boldsymbol{S}} \\
\forall \boldsymbol{m} \boldsymbol{H}(\boldsymbol{m}) \\
\overbrace{\begin{array}{c} H(m1) \\ H(m2) \\ H(m3) \end{array}}
\end{array}
$$

Comparing these two cases, it is easy to see that an appearance of $P([n])$ in a tableau branch is equivalent to an exhaustive listing of all the possible instantiations of the free variable $[n]$.

Along similar lines, it can be said that, an appearance of $[n]$ in EX is equivalent to an exhaustive listing of all the terms in the domain EX that belong to the same sort as the variable n.

For the corresponding existential discourse:

$(S) \circ (\exists m B(m))$

which would be represented as:

$$\overbrace{\begin{array}{|l|}m1\\ m2\\ m3\end{array}\;\begin{array}{|c|}S\\ \\ \\ \end{array}}^{\boldsymbol{S}}$$
$$\boldsymbol{\exists m B(m)}$$
$$B(m_0)$$
$$\overbrace{\begin{array}{|c|}m_0\end{array}\;\begin{array}{|c|}B(m_0)\end{array}}$$

is equivalent to:

$$\overbrace{\begin{array}{|l|}m1\\ m2\\ m3\end{array}\;\begin{array}{|c|}S\\ \\ \\ \end{array}}^{\boldsymbol{S}}$$
$$\boldsymbol{\exists m B(m)}$$
$$\overbrace{B(m1) \quad B(m2) \quad B(m3)}$$

Here again it is easy to see that an appearance of a $P(f_0(\ldots))$ in a tableau branch is equivalent to multiple branching of the same branch such that each of the daughter nodes holds some $P(x)$ for x a term resulting from instantiating the arguments of $f_0(\ldots)$ with appropriate terms from EX.

In this case, the Coverage Property can only be said to hold over the extended representation given for each case in second place. However, without recurring to this notation the complexity of the first order tableaux would be equivalent to that of the propositional case.

The method of representation chosen is a compromise between the loss of the Coverage Property and the complexity of the explicit representation. The solution to lies in the fact that the representation does not explicitly list all the possible alternatives as required by the Coverage Property, but the information that corresponds to all these alternatives is available from the representation if the tableaux are queried using the definition of consequence. This is explained below.

Quantified statements introduce information in a tableau. The information introduced by a quantified statement can be obtained by instantiating the free variables to every item in the domain (EX). This would be equivalent to constructing the explicit tableaux given earlier for example (124) and the corresponding existential discourse, or carrying out all the substitutions on the original tableau and collecting all the resulting new tableaux. Such an operation is unnecessary, because the information so obtained is already available from the initial tableau through querying. To query a particular instantiation, like $H(m2)$ of a universally quantified formula, such as $\forall x H(m)$, it is enough to add the negation of the instantiation, $\neg H(m2)$, to the tableau.

$$\begin{array}{r} \overbrace{\begin{array}{|c|c|c|} m1 & & S \\ m2 & & \\ m3 & & \end{array}}^{\boldsymbol{S}} \\ \boldsymbol{\forall m H(m)} \\ H([m]) \\ \overbrace{\begin{array}{|c|c|c|} [m] & & H([m]) \end{array}} \\ \neg H(m2) \end{array}$$

With no need to list the instantiations of the quantified statement, the proof method shows that the tableau closes (there is a substitution for the free variable $[m]$, that which instantiates it with the constant $m2$ used in the query, that closes the tableau). The reply to the query is positive.

In order to query a quantified statement, it is enough to ensure that there is one substitution that falsifies its negation.

The method chosen for querying the tableau for a proposition ϕ (testing if the negation of ϕ closes the tableau) ensures that all the information in the tableau is always available (under the same constraints on retrieval of the original tableau after querying outlined in chapter 4). The tableau no longer represents all the information contained in the propositions it is built from, but this information can be obtained from it through querying nonetheless.

6.4.3 The Interaction between the Predicate and the Propositional Formalism

Show how the proposed formalism for predicate logic is not enough to cover every presupposition and how the two proposed formalisms would interact.

L^+ Language

Information will be represented by a language L^+.

Definition 11 *The alphabet of L^+ is the same as that of L^1.*

The definitions for individual term and atomic formula for L^+ are the same as for L^1 and L^0.

Definition 12 *Inductive definition of formula in L^+ :*
1) an atomic formula in L^1 is a formula in L^+
2) If A and B are formulas in L^1, then so are A^B, $(A \wedge B)$, $(A \vee B)$, and $(A \rightarrow B)$.
3) If A is a formula in L^+, then so are $\forall x A^*$ and $\exists x A^*$, where A^* is A or obtained from A by replacing occurrences of an individual constant with the variable x.

Mapping Natural Language onto L^+

The language L^+ is an extension of the language L^1 to include the superscript notation for presuppositions that was used in the propositional framework. This allows to combine the two ways of dealing with presupposition described so far. In order to avoid confusion, it is assumed that the only presuppositional relations that are modelled in L^+ using the superscript notation A^B are those that cannot be captured in terms of existence in the domain of the objects referred to by terms used as arguments in the predicate transcription of a sentence as described above.

This is the case of presuppositions of the following types:

(3.k) Sam has stopped breaking the typewriter.

(Sam used to break the typewriter)

(3.m) Sam regrets breaking the typewriter.

(Sam has broken the typewriter)

(3.n) Bill realized Sam has broken the typewriter.

(Sam has broken the typewriter)

The transcription of such examples presents some problems, due to the fact that they contain subordinate sentences. The policy followed so far of retaining as much as possible of the structure explicit in the linguistic form of the sentences suggests that such subordinate sentences appear as arguments of the predicates chosen to represent the verb of the main clause. For example, the sentence

(3.n) Bill realized Sam has broken the typewriter.

might be transcribed as $R(b, S)$, where S would be the appropriate transcription for the sentence *Sam has broken the typewriter*.

ZD Tableaux

The given language would be interpreted in tableaux constructed using the following method.

An ZD tableau is constructed by using rules α, β, σ, ξ, ε, η and ν rules as follows:

- expand using all the rules (mark all additions resulting from ξ and ν rules, if a formula or term is marked, mark its expansion as well)
- once no more rules are applicable, retract those marked additions whose expansions contribute to the closure of a branch

Simpler representations may be obtained if, for the expansion of a given utterance, ε-, η- and ν-rules are applied last after all other applicable rules.

The interpretation of

(128) *Bill does not regret that John's children have forgotten Bill*

(or $\neg R(b, F(b, c(j))^{F(b,c(j))})$ in the context of :

(129) *John does not have children*

(or $\neg\varepsilon(c(j))$) would be:

$$\overbrace{\begin{array}{|c|c|c|}\hline j & c(j) & \neg\varepsilon(c(j))\end{array}}^{\boldsymbol{\neg\varepsilon(c(j))}}$$

$$\begin{array}{r}\boldsymbol{\neg R(b, F(b, c(j))^{F(b,c(j))}} \\ F(b, c(j)\end{array}$$

$$\overbrace{\begin{array}{|c|c|r|} b & & \neg R(b, F(b, c(j))^{F(b,c(j))} \\ b & & F(b, c(j)) \\ j & & \end{array}}$$

It is important to note in this example that the expansion of η-rules can also act in a defeasible way if it inherits the defeasibility from a ξ-rule (the η-rule is applied to a proposition that was marked as being the expansion of a ξ -rule; markings are inherited by every expansion).

The superscript notation for presuppositions and the σ- and ξ- expansion rules could be eliminated altogether if the η- and ν-rules where extended with a way of dealing with sentences that appear as arguments of predicates. Such an alternative formulation presents a strong appeal in that it would correspond closely to the unified concept of reference introduced by Frege (where the reference of an expression is an objects and the reference of a sentence is a truth value). However, because in the system the existence of a referent for an expression and the fact of a certain proposition being true are notated disjointly, the formalization in terms of A^B and σ- and ξ- expansion rules is preferred. An additional argument in favour of this stand is to be found in the fact that in many cases, like in factives, the actual meaning of the predicate that presents sentences as its arguments affects the conditions under which that sentence can be held to be true. For instance, the sentence

(130) *Bill believes that John has children.*

might be represented as $B(b, S)$, but in this case the meaning of *believes* forces a different treatment for the proposition S. Such issues must be dealt in a modal logic framework and are not considered here. to

The sentence

(85) *If John's children have forgotten Bill, Bill does not regret it*

has the form $F(b, c(j)) \to \neg R(b, F(b, c(j)))^{F(b,c(j))}$.

The tableau for that would be:

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{F(b, c(j)) \to \neg R(b, F(b, c(j)))^{F(b,c(j))}}} \\
\hline
\neg F(b, c(j)) & \neg F(b, c(j)) & F(b, c(j)) \\
\neg R(b, F(b, c(j)))^{F(b,c(j))} & R(b, F(b, c(j)))^{F(b,c(j))} & \neg R(b, F(b, c(j)))^{F(b,c(j))} \\
 & \underline{F(b, c(j))} & F(b, c(j))
\end{array}
$$

$$
\overbrace{\left|\begin{array}{c|c|r}
b & & \neg F(b, c(j)) \\
c(j) & & \\
j & & \\
b & & \neg R(b, F(b, c(j)))^{F(b,c(j))}
\end{array}\right|}
\qquad\qquad
\overbrace{\left|\begin{array}{c|c|r}
b & & F(b, c(j)) \\
c(j) & & \\
j & & \\
b & & \neg R(b, F(b, c(j))^{F(b,c(j)}
\end{array}\right|}
$$

The sentence

(73) *Either Bill has started smoking or Bill has stopped smoking.*

has the following representation:

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{A(b, s)^{\neg S(b)} \vee B(b, s)^{S(b)}}} \\
\hline
A(b, s)^{\neg S(b)} & A(b, s)^{\neg S(b)} & \neg A(b, s)^{\neg S(b)} \\
B(b, s)^{S(b)} & \neg B(b, s)^{S(b)} & B(b, s)^{S(b)} \\
\neg S(b) & \neg S(b) & \\
\underline{S(b)} & & S(b)
\end{array}
$$

$$
\overbrace{\left|\begin{array}{c|c|r}
b & & A(b, s)^{\neg S(b)} \\
b & & \neg B(b, s)^{S(b)} \\
b & & \neg S(b)
\end{array}\right|}
\quad
\overbrace{\left|\begin{array}{c|c|r}
b & & \neg A(b, s)^{\neg S(b)} \\
b & & B(b, s)^{S(b)} \\
b & & S(b)
\end{array}\right|}
$$

The sentence

(87) *If John has children, then Bill does not regret that they have forgotten him*

has the form $\varepsilon(c(j)) \rightarrow \neg R(b, F(b, c(j))^{F(b,c(j)}$.

The UAP tableau for that would be:

$$\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{F(b, c(j) \rightarrow \neg R(b, F(b, c(j))^{F(b,c(j)}}} \\
\hline
\begin{array}{c} \neg\varepsilon(c(j)) \\ \neg R(b, F(b, c(j))^{F(b,c(j)} \end{array} &
\begin{array}{c} \neg\varepsilon(c(j)) \\ R(b, F(b, c(j))^{F(b,c(j)} \\ F(b, c(j) \end{array} &
\begin{array}{c} \varepsilon(c(j)) \\ \neg R(b, F(b, c(j))^{F(b,c(j)} \\ F(b, c(j) \end{array} \\
\overbrace{\begin{array}{|c|c|r|} & c(j) & \neg\varepsilon(c(j)) \\ b & & \neg R(b, F(b, c(j))^{F(b,c(j)} \\ j? & & \end{array}} &
\overbrace{\begin{array}{|c|c|r|} & c(j) & \neg\varepsilon(c(j)) \\ b & & \neg R(b, F(b, c(j))^{F(b,c(j)} \\ b & & F(b, c(j)) \\ c(j) & & \\ \hline \end{array}} &
\overbrace{\begin{array}{|c|c|r|} c(j) & & \varepsilon(c(j)) \\ b & & \neg R(b, F(b, c(j))^{F(b,c(j)} \\ b & & F(b, c(j)) \\ c(j) & & \\ j & & \end{array}}
\end{array}$$

The interpretation of the discourse

(75) (*If Mary has had a bath, then there is no hot water left*) $\circ$ (*If Mary has had a bath, then Bill regrets that there is no hot water left*)

or $(B(m) \rightarrow \neg\varepsilon(w)) \circ (B(m) \rightarrow R(b, S)^{\neg\varepsilon(w)})$ shows how the effect of context is captured in the framework. The second sentence of this discourse, if interpreted on its own, gives the following interpretation:

would take place as follows:

$$\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{B(m) \rightarrow R(b, S)^{\neg\varepsilon(w)}}} \\
\hline
\begin{array}{c} \neg B(m) \\ R(b, S)^{\neg\varepsilon(w)} \\ \neg\varepsilon(w) \end{array} &
\begin{array}{c} \neg B(m) \\ \neg R(b, S)^{\neg\varepsilon(w)} \\ \neg\varepsilon(w) \end{array} &
\begin{array}{c} B(m) \\ R(b, S)^{\neg\varepsilon(w)} \\ \neg\varepsilon(w) \end{array} \\
\overbrace{\begin{array}{|c|c|r|} m & & \neg B(m) \\ b & & R(b, S)^{\neg\varepsilon(w)} \\ & w & \neg\varepsilon(w) \end{array}} &
\overbrace{\begin{array}{|c|c|r|} m & & \neg B(m) \\ b & & R(b, S)^{\neg\varepsilon(w)} \\ & w & \neg\varepsilon(w) \end{array}} &
\overbrace{\begin{array}{|c|c|r|} m & & \neg B(m) \\ b & & R(b, S)^{\neg\varepsilon(w)} \\ & w & \neg\varepsilon(w) \end{array}}
\end{array}$$

However, the interpretation of the whole discourse shows that the very same sentence is expanded differently along the different branches of the final representation.

In the tableau for the first sentence of discourse (75), it is apparent that the middle branch represents a different local context to the other two. This implies that the expansion of the second sentence is different in this case. The resulting branch has a different domain ($EX = \{m, w, b\}$) than the other two ($EX = \{m, b\}$).

6.5 Conclusions

An extended proof language is developed for the zero order predicate logic that makes the information about local domains explicit in the object language. This allows a set of alternative interpretations for quantifiers to be explored.

During dialogue, existence statements may be used by participants in a generic discourse in order to ascertain whether they share the necessary domain for the generic statements to be reliable.

The method given for quantification is similar to Lejewsky's unrestricted quantification, but with an additional operation of presupposition to determine what actually exists (in addition to existence predicates). The information specified jointly by presupposition and existence predicates is notated as domain information that constrains quantification. The notation also allows denotation of information about non-existence that also plays a role in interpretation.

Information obtained from quantified statements is shown to be liable to error unless the domain under consideration has been carefully delimited.

In contexts with non-fixed domain, universal and existential quantification become unreliable, just as in actual language. For quantifiers to be 'sound', a 'closed domain' statement would be needed. (Something to be interpreted as 'the objects mentioned so far *and only those* are considered as the present domain.')

Chapter 7

A Preliminary Study of Presuppositions in First Order Logic: Semantics

This chapter is only intended as a preliminary study. As such, it does not tackle in detail the semantic implications of the proposed proof theory. However, it is worth mentioning a few details concerning these issues.

7.1 Elementary Semantics for the Predicate Case

A model for the first-order language $L^0(R, F, C)$ is a tuple $M =< D, D_i, I >$ where D is a non-empty set, called the general domain of M, D_i is a non-empty set, such that $D_i \subseteq D$, called the local domain of M, I is a mapping, called an *interpretation* that associates :

- to every constant symbol $c \in C$ some member $c^I \in D$
- to every n-place function symbol $f \in F$ some n-ary function $f^I : D^n \rightarrow D$
- to every n-place relation symbol $P \in R$ some n-ary relation $P^I \subseteq D^n$

To each term t of $L(R, F, C)$ we associate a value t^I in D as follows for a function symbol f, $[f(t_1, \ldots, t_n)]^I = f^I(t_1^I, \ldots, t_n^I)$.

If any $t_i^I \notin D_i$ then $[f(t_1, \ldots, t_n)]^I \notin D_i$.

Let $M =< D, D_i, I >$ be a model of the language $L^0(R, F, C)$. To each formula ϕ of $L^0(R, F, C)$ we associate a truth value ϕ^I (t or f) as follows:

- for the atomic cases $[P(t_1, \ldots, t_n)]^I = t$ iff $t_1^I, \ldots, t_n^I \in D_i$ and $< t_1^I, \ldots, t_n^I >\in P^I$, and $[P(t_1, \ldots, t_n)]^I = f$ otherwise
- for each existence statement $[\varepsilon(t)]^I = t$ iff $t^I \in D_i$, and $[\varepsilon(t)]^I = f$ otherwise
- $[\neg X]^I = \neg[X^I]$
- $[X \circ Y]^I = X^I \circ Y^I$ (for $\circ$ any of the binary connectives)

A formula is *valid in D* if it is true in all interpretations in D, and *satisfiable in D* if it is true in at least one interpretation in D.

In each model one can talk about the whole domain D. Existence and non-existence predicates have a role as means to distinguish objects in the local domain D_i from objects being talked about from that model.

A formula ϕ of L is true in a model M ($M \models \phi$) iff $[\phi]^I = t$.

For $\phi_1, \ldots, \phi_n, \psi$ any formulas of L, the statement $\phi_1, \ldots, \phi_n \models \psi$ means: for every model M, if $M \models \phi_1$ and ... and $M \models \phi_n$, then $M \models \psi$.

The statement $\models \phi$ means that for every model M, $M \models \phi$.

(These are statements about formulas and not formulas themselves).

The representation that each individual participant in an exchange keeps, must be partial in terms of information (not determine all the possible information) and it must be possible to extend it continuously.

Assuming a possible world semantic model, the semantic representation of a set of sentences is the set of possible worlds in which the sentences are true.

The effect of adding an utterance to a set of sentences is eliminating from the semantic representation of the set of sentences those possible worlds where the utterance is not true.

Basic mechanism:

- Start with empty representation (the set of all possible worlds)
- $P(a)$ eliminates worlds w_i where $a \notin D_i$ and worlds w_j where $a \in D_j$ but $P(a)$ does not hold in w_j
- $\neg P(a)$ eliminates worlds w_i where $P(a)$ holds (this means that $a \in D_i$ for the worlds w_i that are eliminated, but says nothing regarding presence of a in the domains of the worlds that are retained)
- $\varepsilon(a)$ eliminates worlds w_i where $a \notin D_i$
- $\neg\varepsilon(a)$ eliminates worlds w_i where $a \in D_i$

7.2 The Concept of Reference

The formalism chosen as means of representation in this chapter allows definition of a rough and ready concept of reference that is useful in discriminating among the different intuitive concepts that are at play, such as reference, correference, anaphora ...

Every (syntactic) term of the language L^0, when it appears as an argument in a proposition, may be being used to *refer* to an object in the domain D_i under consideration.

In this way, the sets EX and NEX are only collections of terms. Semantically, these collections of terms can be interpreted as sets of objects in the domain D.

$$E = \{a \in D_i | t^I = a \text{ for some } t \in EX\}$$
$$N = \{b \in D_i | t^I = b \text{ for some } t \in NEX\}$$

As a listener interprets a sequence of sentences, he builds up a picture of the domain to which they refer. Such a picture is represented in the present formalism by the sets E and N, as determined by EX and NEX. The set E constitutes a positive restriction on the domains D_i that can be considered in the branch. Only domains D_i such that $E \subset D_i$ can be considered. The set

N constitutes a negative restriction on the domains D_i that can be considered in the branch. Only domains D_i such that $N \cap D_i = \emptyset$ can be considered.

7.3 The Semantics of Contextual Quantification

Semantically, the approach outlined in the proof rules given above implies that each quantified statement is evaluated against a partial model.

A sentence $\forall x \phi(x)$ is true in a context determined by EX and NEX iff for every model $M =< D, D_i, I >$ such that $D_i \cap NEX = \emptyset$ and $EX \subset D_i$ $[\phi(t)]^I = t$ for every $t \in EX$.

One problem is whether the information introduced by a quantified statement is preserved after the domain is altered. Because the instantiations are not actually carried out in the tableau, alterations to the domain may change this information.

> (131) *John arrived late. Bob was already there. Jane was wiping the tabletop. Someone had spilt some wine. Nobody spoke a word. Everyone had heard the story.*

At this stage, the tableau would contain the potential instantiations for

> (132.a) *John had heard the story.*
>
> (132.b) *Bob had heard the story.*
>
> (132.c)d *Jane had heard the story.*

Suppose the following sentences are added to the discourse:

> (133) *Bill walked in. He was whistling.*

Because of the way that quantified statements are stored in the tableau, an instantiation of the form

> (134) *Bill had heard the story*

is possible. However, such a statement should really be contingent. The actual discourse does not imply the statement, because the quantifier was used when the domain included only John, Bob, and Jane.

The key to the apparent difference in the mode of quantification (cautious quantification for the universal quantifier and optimistic quantification for the existential quantifier; as described in the first section of this chapter) lies in the fact that variables are all instantiated cautiously as given by the substitutions, but in the case of the existential quantifier, the quantified variable develops its actual reference through a Skolem function, which actually forces the referent to be other than the variables that are cautiously instantiated. Because a function with arguments in EX can refer to objects outside EX, existentially quantified variables seem to take values outside EX.

$$\frac{\exists x \phi(x)}{\phi(f(x_1, \ldots, x_n))}$$

where $x_1, \ldots, x_n \in EX$ but $f(x_1, \ldots, x_n)$ may be in D_i and not be in EX.

A sentence $\exists x \phi(x)$ is true in a context determined by EX and NEX iff for every model $M =< D, D_i, I >$ such that $D_i \cap NEX = \emptyset$ and $EX \subset D_i$ $[\phi(s)]^I = t$ for some s such that $s^I \in D$.

7.3.1 Assignments

The semantic definition of quantification is closely related with the definition of assignment.

An *assignment* in a model $M =< D, D_i, I >$ is a mapping A from the set of variables to the set D_i [1]. We denote the image of the variable v under an assignment A by v^A.

To each term t of $L(R, F, C)$ is associated a value $t^{I,A}$ in D_i as follows:

[1] The exact delimitation of what should take the place of this set in the case of contextual quantification defines the different modes.

- for a constant symbol c, $c^{I,A} = c^I$
- for a variable v, $v^{I,A} = v^A$
- for a function symbol f, $[f(t_1, \ldots, t_n)]^{I,A} = f^{I,A}(t_1^{I,A}, \ldots, t_n^{I,A})$.

Let x be a variable. The assignments A and B are *x-variants* provided they assign the same values to every variable except possibly x.

The different modes of quantification considered in the proof theory correspond to defining an assignment as mapping respectively to: D, E, and $D \backslash N$. However, if the semantics are to mirror the observed behaviour (as has been shown for the syntax), the semantics of quantifiers in terms of assignments cannot be achieved by a simple choice of the set that assignments map into. This is due to the fact that universal quantification works cautiously and existential quantification works optimistically.

The following definitions would be required.

Let $M =< D, D_i, I >$ be a model of the language $L(R, F, C)$. To each formula ϕ of $L(R, F, C)$ we associate a truth value ϕ^I (t or f) as follows. (Only additional clauses for quantifiers are given. For all other formulas, see above.)

- $[\forall x \phi]^{I,A} = t$ iff $\phi^{I,B} = t$ for every x-variant B of A, where A is defined over the domain defined by E.
- $[\exists x \phi]^{I,A} = t$ iff $\phi^{I,B} = t$ for some x-variant B of A, where A is defined over the domain defined by $D \backslash N$.

7.4 Correference: Definites

For the first order case there is also some redundancy apparent in the examples.

In this case, the issue has a different conceptual significance in that it involves the concept of reference. Throughout this chapter it has been assumed that terms of the language L^1 refer to objects (in the first approximation to objects in the semantic models; ultimately to objects in the real world). Because the question of the semantics of L^1 has not been addressed in detail, it is not possible to consider this issue in depth. However, every redundant addition of a term to EX corresponds to a ‘repeated reference’ by the second instance of the term to the same object ‘referred to’ by the first instance of the term. I refer to this phenomenon as *correference*.

Let terms of the language as they have been defined stand for definite constructions. A term may appear several times in a discourse, but in order to keep an accurate record of information about the domain it need only be counted once.

Because of this fact, it would be useful if the framework were extended with the appropriate constraints so that terms are only added to the EX column if they are not already present.

An additional mechanism would be desirable to keep track of the references, but such a mechanism would rely heavily on the semantics that underlie the language. In order to keep track of where in the tableau this additional semantic process is taking place, I add the symbol $\Uparrow$ to the EX and NEX columns wherever a term that is already present in either one of them EX is being used in a second ‘referring’ instance.

The efficiency obtained by adopting this procedure is apparent in the example

> (118) *If there is no king of France then France does not cherish its king.*

If redundant additions of terms are avoided, the representation given earlier:

$$\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{\neg\varepsilon(k(f)) \to \neg C(f,k(f))}} \\ \hline
\begin{array}{r}\varepsilon(k(f)) \\ \neg C(f,k(f))\end{array} & \begin{array}{r}\varepsilon(k(f)) \\ C(f,k(f))\end{array} & \begin{array}{r}\neg\varepsilon(k(f)) \\ \neg C(f,k(f))\end{array} \\
\overbrace{\left|\begin{array}{c|c|r} k(f) & & \varepsilon(k(f)) \\ f & & \\ f & & \neg C(f,k(f)) \\ k(f) & & \\ f & & \end{array}\right|} & \overbrace{\left|\begin{array}{c|c|r} k(f) & & \varepsilon(k(f)) \\ f & & \\ f & & C(f,k(f)) \\ k(f) & & \\ f & & \end{array}\right|} & \overbrace{\left|\begin{array}{c|c|r} f & k(f) & \neg\varepsilon(k(f)) \\ f & & \neg C(f,k(f)) \end{array}\right|}
\end{array}$$

is now reduced to:

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{\neg\varepsilon(k(f)) \to \neg C(f,k(f))}} \\
\hline
\begin{array}{r} \varepsilon(k(f)) \\ \neg C(f,k(f)) \end{array} & \begin{array}{r} \varepsilon(k(f)) \\ C(f,k(f)) \end{array} & \begin{array}{r} \neg\varepsilon(k(f)) \\ \neg C(f,k(f)) \end{array} \\
\overbrace{\begin{array}{|c|c|r|} k(f) & & \varepsilon(k(f)) \\ f & & \\ \Uparrow & & \neg C(f,k(f)) \end{array}} & \overbrace{\begin{array}{|c|c|r|} k(f) & & \varepsilon(k(f)) \\ f & & \\ \Uparrow & & C(f,k(f)) \end{array}} & \overbrace{\begin{array}{|c|c|r|} f & k(f) & \neg\varepsilon(k(f)) \\ \Uparrow & & \neg C(f,k(f)) \end{array}}
\end{array}
$$

It is interesting to note that in the last branch of this tableau the proposition $\neg C(f, k(f))$ has as an argument the term $k(f)$, in a context where it is known that there is no object in the domain that corresponds to it, that it can refer to (because there is no King of France). Such a term does not refer to any objects in the sense of referring that has been considered so far. Yet in syntactic terms, there is a relationship between the occurrence (always within this particular branch) of the term $k(f)$ as an argument of $\neg C(f, k(f))$ and the occurrence of the same term $K(f)$ as an argument of $\varepsilon(k(f))$. If $\neg\varepsilon(k(f))$ can be considered to state the non-reference of the term $k(f)$ as it appears within it, the second appearance of $k(f)$ seems to inherit the non-reference of the first precisely through this syntactic relationship between them.

Expansion of domain information follows a conservative policy in domain management (first try to 'refer' to constants in EX or NEX) analogous to the conservative policy followed in abduction for the management of a logical theory.

7.5 Blocking Correference: Indefinites

The present version of the framework lends itself easily to the consideration of definite and indefinite constructions at a very basic level. Indefinite constructions can be represented by a similar set of terms especially marked to distinguish them. There should be a special correspondence between the definite and the indefinite constructions, so for every definite term there is an indefinite term that is mapped onto the same object by the semantics. Therefore the definite and indefinite terms are semantically equivalent, but they behave differently in the proof theory.

In the present framework an indefinite term will be represented by the same term as its definite counterpart but underlined. The following table gives a few examples.

Expression	Definite term	Indefinite term
John	j	$\underline{j}$

Heim argues that the contrast between definite and indefinite constructions lies in the fact that indefinites are assumed to point to a new referent and definites are assumed to point to referents that were already present. In the previous section it has been shown that the terms used in a proposition may 'refer' to terms used previously or they may introduce new domain information. The similarity between these two descriptions suggest that in the present framework the definite-indefinite contrast can be represented by assuming that definite constructions are expected to be used in the 'referring' role and indefinites are expected to be used in the 'introducing' role.

However, definite constructions are sometimes used to introduce domain information (by presupposition, as shown in the previous chapters). An adequate modelling of the definite-indefinite contrast is better obtained in this case by assuming that indefinite constructions cannot be used in a 'referring' role. Definite constructions operate as had been described earlier for generic terms. This approach implies that an indefinite construction cannot be interpreted as 'referring' even when there is a previous available 'referent'.

The difference in behaviour between definites and indefinites becomes apparent in the following examples:

(135.a) A boy walked in. The boy shouted.

(135.b) The boy walked in. A boy shouted.

For example (135.a):

$$\overbrace{\left|\ \underline{b1}\ \right| \quad \left|\ I(\underline{b1})\ \right|}^{\boldsymbol{I(\underline{b1})}}$$
$$\overbrace{\left|\ \Uparrow\ \right| \quad \left|\ S(b)\ \right|}^{\boldsymbol{S(b)}}$$

where the definite term b in $S(b)$ has been interpreted correctly as a reference to the indefinite $\underline{b1}$ introduced earlier.

For example (135.b):

$$\overbrace{\left|\, b \,\right| \;\; \left|\, I(b)) \,\right|}^{\boldsymbol{I(b)}}$$
$$\overbrace{\left|\, \underline{b1} \,\right| \;\; \left|\, S(\underline{b1}) \,\right|}^{\boldsymbol{S(\underline{b1})}}$$

where the indefinite term $\underline{b1}$ in $S(\underline{b1})$ has been interpreted correctly as a reference to a new boy, different from the definite b introduced earlier.

7.6 Anaphora

Terms have so far been treated as names, being mapped into constants of a given sort. A rough treatment of anaphora can be achieved if anaphors are considered as free variables of a given sort, to be bound during interpretation to a single unique term of the same sort.

An anaphoric term inherits the reference of the term that it is bound to.

In the present formalism anaphoric terms will be represented generically by an additional auxiliary term of the language, represented as $\Box$. In order to represent anaphoric binding, each instance of $\Box$ is given as a subscript a copy of the term that is chosen as its antecedent.

Binding of an anaphor is different from instantiation of a free variable introduced by the expansion of a quantifier. The basic difference is an anaphor has one (unique) intended binding, whereas a free variable x resulting from the expansion of a quantifier has many instantiations (in fact, all the possible ones) .

Additional complication is introduced by the fact that there are several types of term that an anaphor may be bound to. Take for instance, the following examples:

(136.a) *John came in. He was crying.*

(136.b) *Mary's husband came in. He was crying.*

(136.c) *Some farmer has a car. He drives it.*

(136.d) *Every farmer has a donkey. He beats it.*

(136.e) **Every farmer has a donkey. He plays golf.*

Each one of them contains a pronominal reference that may be represented in terms of anaphora in the framework.

The corresponding discourses can be represented as follows.

For example (136.a)

(136.a) *John came in. He was crying.*

the representation would be:

$$\overbrace{\left| \; j \; \right| \; \left| \; I(j) \; \right|}^{\boldsymbol{I(j)}} \atop \boldsymbol{C(\Box_j)}$$

where $\Box_j$ inherits the reference of the constant j.

For example (136.b):

(136.b) *Mary's husband came in. He was crying.*:

the representation would be:

$$\overbrace{\left| \; h(m) \; \right| \; \left| \; I(h(m)) \; \right|}^{\boldsymbol{I(h(m))}} \atop \boldsymbol{C(\Box_{h(m)})}$$

where $\Box_{h(m)}$ inherits the reference of the function constant $h(m)$.

For example (136.c)

(136.c) *Some farmer has a car. He drives it.*

the representation would be:

$$\begin{array}{c}
\boldsymbol{\exists f H(f, d(f))} \\
\overbrace{\left| \begin{array}{c} f_0 \\ d(f_0) \end{array} \right| \; \left| \begin{array}{l} H(f_0, d(f_0)) \\ \end{array} \right.}^{H(f_0, d(f_0))} \\
\boldsymbol{B(\Box_{f_0}, \Box_{d(f_0)})}
\end{array}$$

where $\Box_{f_0}$ inherits the reference of the Skolem function constant f_0 and $\Box_{d(f_0)}$ of the function constant $d(f_0)$.

For example (136.d)

(136.d) *Every farmer has a donkey. He beats it.*

the representation would be:

$$\begin{array}{c}
\boldsymbol{\forall f H(f, d(f))} \\
\overbrace{\left| \begin{array}{c} [f] \\ d([f]) \end{array} \right| \; \left| \begin{array}{l} H([f], d([f])) \\ \end{array} \right.}^{H([f], d([f]))} \\
\boldsymbol{B(\Box_1, \Box_2)}
\end{array}$$

where $\Box_1$ inherits the reference of the free variable $[f]$ and $\Box_2$ of the function constant $d([f])$.

For example (136.e)

(136.e) **Every farmer has a donkey. He plays golf.*

the representation would be:

$$\begin{array}{c}
\boldsymbol{\forall f H(f, d(f))} \\
\overbrace{\left| \begin{array}{c} [f] \\ d([f]) \end{array} \right| \; \left| \begin{array}{l} H([f], d([f])) \\ \end{array} \right.}^{H([f], d([f]))} \\
\boldsymbol{G(\Box)} \\
\overbrace{\left| \; \Box \; \right| \; \left| \; G(\Box) \; \right|}
\end{array}$$

where $\Box$ cannot inherit the reference of the free variable $[f]$. This is related in some way to the fact that in $G(\Box)$, $\Box$ is not being related to the function constant $d([f])$.

So the same pronoun is represented as binding anaphorically to the following types of syntactic term:

- a constant a
- a term $f(t_1, \ldots, t_n)$ such that $[f(t_1, \ldots, t_n)]^I = a^I$
- a free variable x such that $[x]^A = a^I$ for some assignment A
- a Skolem function $f(x_1, \ldots, x_n)$ such that $[f(x_1, \ldots, x_n)]^{I,A} = a^I$

The problem here is that an anaphor can be bound not only to a previous constant, but also to a free variable introduced by a quantifier. The binding takes place independently of the possible instantiations of the free variable, and it consists of tying up the anaphor to whatever instantiation is later given to the free variable. Therefore, when an anaphor has become bound to a free variable, the proposition that the anaphor appeared in automatically inherits the instantiation constraints (and therefore the quantification) of the referent. This issue only plays a role if the sentence were to be paraphrased later (the information represented in the tableau transcribed back into natural language and communicated again). If binding of anaphors takes place from EX and NEX, then all that is required is to specify how quantifiers affect EX and NEX. If a term is in EX and is picked out by binding of an anaphor , it does not mater whether it was put in EX by a quantifier or not.

When an anaphor is bound to a term in the EX column, it refers to the object that the term refers to.

When an anaphor is bound to a free variable in the EX column, for each possible instantiation it refers (indirectly) to the object that the term that the free variable is instantiated with refers to.

The behaviour of $\Box$ can be compared to that of $\Uparrow$. The main difference in the present system is that $\Uparrow$ appears in EX whenever an expansion is redundant, whereas $\Box$ appears directly as an argument in the corresponding predicate. The similarities in behaviour and function over the representation suggest that both representational devices correspond to phenomena that are very similar in nature. This is indeed the position defended by Sandt. The similarities may be brought out even more clearly if redundant expansion into EX are avoided altogether and replaced by a substitution of the redundant argument by a conveniently subscripted $\Uparrow$.

Chapter 8

Comparison with Other Work

8.1 Update Accounts

The great advantage of context change accounts of presupposition is that, for those connectives that are defined as primitive, the context change potentials needed to obtain the accepted truth conditional behaviour of the connectives also have the property of always requiring evaluation of the presuppositions in contexts where they have been added immediately before. This is the basis for their claim of descriptive adequacy. However, their predictions fail for disjunction, which they do not define as primitive but, in contrast with conditionals and conjunction, is symmetric in its behaviour towards presupposition. Because of this difference in properties of symmetry, disjunction cannot be defined in terms of the other two connectives.

8.1.1 Quantification

The predicate tableau presented here relate closely to Heim's idea that the representation of the context as possible worlds be replaced with one where context is represented by pairs $< g, w >$ of sequences of referents g and possible worlds w. In the present framework the sets $EX(\Delta_i)$ play a role in the proof theory akin to that played by Heim's sequences of referents in the semantics, in as much as they determine the domains of quantification. A closer comparison will have to wait until the semantic implications of the proposed proof theory for quantification are considered in detail.

8.1.2 Local and Global Accommodation

The addition of presupposition to the context as extra information is justified and explained in the present formalism in a way that accounts in terms of accommodation cannot match. First of all, because accommodation is defined as a repair operation once a transgression of the original rules occurs. But most of all because accommodation is at a loss when trying to set out appropriate criteria for deciding when to use local and when global accommodation. In the present framework the answers to both questions follow simply from the basic assumptions.

Aside from this major difference, the final version of the framework with presupposition expansion rules is equivalent to Heim's account of cancelation in terms of local accommodation for presuppositions that originate under the scope of negation.

In Heim's framework, when processing the negation of a sentence, the context can be divided into two local subcontexts: one where the incoming sentence is true and one where it is false. Global accommodation corresponds to accommodating any presuppositions of the sentence in the whole of the original context. Local accommodation corresponds to accommodating any presuppositions of the sentence only in the local subcontext where the sentence is true and not accommodating it where it is false. Heim postulates that global accommodation is preferred in general terms, but local accommodation occurs when the context holds the negation of the presupposition.

In the present framework, presuppositions that originate under the scope of negation are governed by the expansion rule for presuppositions of negative sentences. But the same sentence may appear unnegated in a different branch. The presuppositions of the unnegated sentence will always be added to the corresponding branch. The presuppositions of the negated sentence will be added to the corresponding branch only if that branch does not contain already the negation of the presuppositions. Heim's local accommodation corresponds to the case where the presupposition of the negated sentence is blocked and the presupposition of the unnegated sentence goes through. Heim's global accommodation corresponds to the case where both go through, and therefore go on to become presuppositions of the whole.

This is easily apparent from the representation for the sentence

(71) *If there is a typewriter then the typewriter is blue*

as given earlier:

$\boldsymbol{t \to b^t}$		
$\neg t$	$\neg t$	t
b^t	$\neg b^t$	b^t
$\underline{t}$		t

Conceptually, the operations that have taken place over the different branches of this tableau are analogous to the process of local accommodation as described by Heim. In this case, the presupposition t could be said to have been accommodated locally in the last branch of the tableau. It would have been accommodated globally if it had also been added to the neighbouring branch. Because the addition has been blocked in this one, the final result is that the presupposition has an effect on a local context.

The CCP approach is correct in as much as the context for the interpretation of the components of a sentence must be different from the context for interpreting the sentences itself. However, it fails on two counts. On one hand, by not accepting that the CCPs of disjunctions (and also, to a certain extent, conditionals) must include the possibility of a branching representation of context. On the other hand, by not accepting that accommodation (or whatever equivalent operation is used to model how presupposition contributes to a context) must be defeasible (as Gazdar and Mercer suggest) or *de jure* as well as *de facto* (as Soames suggests). The concept of local accommodation is very close to solving the problem because it relies on the natural branching of a context that negation produces (worlds where ϕ holds versus worlds where $\neg\phi$ holds) and allows accommodation in one of these branches to be waived. However, because it fails to acknowledge these basic ingredients (branching context, defeasibility) underlying the explanation, the CCP account cannot give full justification for local accommodation nor adequate criteria for governing its use.

8.1.3 An Example in Detail

The interpretation of the discourse

> (75) (*If Mary has had a bath, then there is no hot water left*) $\circ$ (*If Mary has had a bath, then Bill regrets that there is no hot water left*)

can provide a case study over which to compare the methods and predictions of the several methods.

For ease of comparison, wherever possible the references to the other frameworks have been written the same notation for the language used for the present one.

In terms of update semantics, the discourse above corresponds to the sequence of updates $[m \rightarrow \neg w]$, $[m \rightarrow (\partial(\neg w) \wedge r)]$.

In order to consider the interpretation process in intuitive terms it is better to resort (as Beaver does in his explanations) to a more syntactic representation of the context. For the example above, the possible worlds used to represent the context can be classed into the following categories depending on the values they assign to the atomic propositions involved in this example.

		Value assigned to proposition		
Possible world	W_1	m	w	r
	W_2	m	w	$\neg r$
	W_3	m	$\neg w$	r
	W_4	$\neg m$	w	r
	W_5	$\neg m$	$\neg w$	r
	W_6	$\neg m$	w	$\neg r$
	W_7	m	$\neg w$	$\neg r$
	W_8	$\neg m$	$\neg w$	$\neg r$

Let the context σ then be the set $\{W_1, W_2, W_3, W_4, W_5, W_6, W_7, W_8\}$.

The interpretation of the first sentence operates as follows.

$$\begin{aligned} \sigma[m \rightarrow \neg w] \quad &= \sigma[\neg(m \wedge \neg\neg w)] \\ &= \sigma \setminus \sigma[m \wedge \neg\neg w] \end{aligned}$$

this eliminates from the context all those worlds where m and w are true together. This leaves worlds W_3 to W_8. Let this be the context σ_1.

Against this context the second sentence must be interpreted. This is the sentence that involves presuppositions, and it pays to carry out the whole process in more detail.

$$
\begin{aligned}
\sigma_1[m \rightarrow (\partial(\neg w) \wedge r)] \quad &= \sigma_1[\neg(m \wedge \neg(\partial(\neg w) \wedge r))] \\
&= \sigma_1 \setminus \sigma_1[m \wedge \neg(\partial(\neg w) \wedge r)] \\
&= \sigma_1 \setminus \sigma_1[m][\neg(\partial(\neg w) \wedge r)]
\end{aligned}
$$

It is easier to process the intermediate context $\sigma_1[m][\neg(\partial(\neg w) \wedge r)]$ on its own first. The proposition m eliminates all worlds but W_3 and W_7. From these, those worlds where the presuppositional sentence holds must be substracted. In both of these $\neg w$ is satisfied, so the presuppositional proposition is admitted. The proposition r is satisfied in world W_3, so this is eliminated. World W_7 is left as the only world contained in the intermediate context $\sigma_1[m][\neg(\partial(\neg w) \wedge r)]$. The full context $\sigma_1 \setminus \sigma_1[m][\neg(\partial(\neg w) \wedge r)]$ then results in the worlds W_3, W_4, W_5, W_6, W_8.

Of these, W_3, W_5, W_6 and W_8 correspond to the truth value assignments found in the branches of the UAP tableau for this discourse.

World W_4 is given by Beaver's framework as a valid information state in the framework. It corresponds to the truth value assignment $\neg m, w, r$, that is a situation where Mary has not had a bath, there is hot water, and Bill regrets that there is no hot water.

This seems to be related with the fact that the presupposition is only tested in the very last of the local contexts, so it may not get tested in contexts in levels closer to the surface that are then allowed to survive as valid information states even though they fail to satisfy the presupposition.

8.2 Default Accounts

The method followed by the compositional rules is quite close to Mercer's and Gazdar's method for computing presuppositions simply in terms of consistency with the context. However, it presents two major innovations: 1) it allows both satisfaction and cancellation (presuppositions disappear when inconsistent and/or when already present), and 2) it applies the procedure locally. The combination of these two innovations allows adequate treatment of hybrid cases and modelling of reasoning by case analysis.

8.2.1 Presuppositional Tableaux and Order of Application of Rules

The major criticism against Gazdar's solution to the projection problem was that it relied on a given order of processing the implicatures of a sentence in order to operate correctly, and no justification could be provided for this specific order. Mercer's default logic solution avoids a similar criticism by using the definition of default extension in terms of a fix point operator, thus avoiding the need to consider any specific order of processing. A similar abstraction from the order of application is obtained in the present framework by the 'apply all and retract some' approach to expansion rules.

8.2.2 Implicatures and Reasoning by Cases

In the present framework the information that Mercer must introduce as pragmatic implicatures can be read off the representation. An expansion rule like:

$$\frac{\beta_1 \vee \beta_2}{\begin{array}{ccc} \beta_1 & \neg\beta_1 & \beta_1 \\ \beta_2 & \beta_2 & \neg\beta_2 \end{array}}$$

captures the idea that each of β_1, $\neg\beta_1$, β_2 and $\neg\beta_2$ is valid in at least one of the possible alternatives given. This property of the present framework is a direct result of the insistence on taking the semantics – as given by the internal structure of the sentence in terms of connectives – into account.

In the present framework the selection of cases is done automatically by the expansion rules. Whenever a tableau branches out, every open branch that results is treated as a possible case. The cases that result from the expansion rules fulfill Mercer's criteria.

Mercer's choice of case is smaller than that resulting from the tableau expansion rules. According to his own analysis, this suggests that the choice of cases that results from the tableau expansion rules is more restrictive than Mercer's. Since this does not prevent the tableau expansion framework from giving the correct predictions, the automatization in the choice of cases is an advantage of the presuppositional tableaux framework.

Mercer gives the following choice of cases for disjunctions $\alpha \vee \beta$:

$$\begin{array}{cc} \alpha & \neg\alpha \\ \neg\beta & \beta \end{array}$$

and for conditionals $\alpha \to \beta$:

$$\begin{array}{ll} \neg\alpha & \alpha \\ \neg\beta & \beta \end{array}$$

These choices can be seen as partial views of the β expansion rules.

For specific examples, the difference in choice does not result in differing predictions. This is because the case that Mercer rejects happens to be closed in the problematic examples for which predictions have been tried. This is so for sentence like

(71) *If there is a typewriter then the typewriter is blue*:

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{t \to b^t}} \\ \hline \neg t & \neg t & t \\ b^t & \neg b^t & b^t \\ \underline{t} & & t \end{array}$$

and

(73) *Either Bill has started smoking or Bill has stopped smoking*

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{e^{\neg s} \vee p^s}} \\ \hline e^{\neg s} & e^{\neg s} & \neg e^{\neg s} \\ p^s & \neg p^s & p^s \\ \neg s & \neg s & \\ \underline{s} & & s \end{array}$$

but not for

(69) *If the typewriter is blue then Sue will be happy*

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{b^t \to h}} \\ \hline \neg b^t & \neg b^t & b^t \\ h & \neg h & h \\ t & t & t \end{array}$$

or

(70.c) *If Mary has had a bath, then Bill regrets that there is no hot water left*

$$\begin{array}{ccc} \multicolumn{3}{c}{\boldsymbol{m \to r^{\neg w}}} \\ \hline \neg m & \neg m & m \\ r^{\neg w} & \neg r^{\neg w} & r^{\neg w} \\ \neg w & \neg w & \neg w \end{array}$$

In this last two examples there are cases that show a significant contribution and yet are not considered by Mercer's method. Their presuppositional contribution happens to be in accordance with those of the other cases, but this need not always be the case.

This problem, and the general gain in simplicity presented by the presuppositional tableaux method is more evident the more complex the sentence is. In [30] Mercer attempts to extend his method to complex conditionals. He applies a process of selection of 'appropriate' cases in order to reduce the complexity based on two different methods to justify his selection. He considers none of them totally satisfactory.

These methods give him the following case selections:

For sentences *If* α *or* β *then* δ:

$$\begin{array}{ccc} \alpha & \neg\alpha & \neg\alpha \\ \neg\beta & \beta & \neg\beta \\ \delta & \delta & \neg\delta \end{array}$$

For sentences *If* α *then if* β *then* δ:

$$\begin{array}{cccc} \alpha & \neg\alpha & \alpha & \neg\alpha \\ \beta & \beta & \neg\beta & \neg\beta \\ \delta & \neg\delta & \neg\delta & \neg\delta \end{array}$$

For sentences *If* α *and* β *then* δ:

$$\begin{array}{ccc} \alpha & \neg\alpha & \alpha \\ \neg\beta & \beta & \beta \\ \neg\delta & \neg\delta & \delta \end{array}$$

This last selection of cases can be compared with those used implicitly by the presuppositional tableaux method in the case of the tableau for sentence

(74) *If John is married and he has children, then his children are at school*

given earlier (the tableau is rephrased here using the same metavariables as in the cases above to make the comparison more transparent):

$$
\begin{array}{ccccccc}
\multicolumn{7}{c}{\boldsymbol{(\alpha \wedge \beta) \to \delta^{\beta}}} \\
\hline
\multicolumn{3}{c}{\neg(\alpha \wedge \beta)} & \multicolumn{3}{c}{\neg(\alpha \wedge \beta)} & \alpha \wedge \beta \\
\multicolumn{3}{c}{\delta^{\beta}} & \multicolumn{3}{c}{\neg\delta^{\beta}} & \delta^{\beta} \\
\hline
\neg\alpha & \neg\alpha & \alpha & \neg\alpha & \neg\alpha & \alpha & \alpha \\
\neg\beta & \beta & \neg\beta & \neg\beta & \beta & \neg\beta & \beta \\
\underline{\beta} & \beta & \underline{\beta} & & \beta & & \beta
\end{array}
$$

Excluding the inconsistent cases that correspond to closed branches, the presuppositional tableau representation gives two cases that Mercer does not consider. The predictions are still the same because the cases that Mercer selects are a representative subset of all the possible cases. However, there is no guarantee that this will always be the case, and the fact the selection of cases is done automatically for presuppositional tableau and with no need for an additional justification argument is a definite advantage.

8.2.3 Comparisons in Predictions

Mercer's definitions of presupposition are only intended to cover the presuppositions of negative sentences. However, his proof method combines these with presuppositions of positive sentences as follows: whenever a case analysis results in a certain presupposition being 'presupposed' (Mercer's terminology for it arising from a negated presuppositional sentence) in one case and 'entailed' (it arises from a positive presuppositional sentence) in another, the 'weaker of the two relationships' prevails. This means that the sentence is presupposed even if in some of the cases it works as an entailment. Given this mechanism, the predictions of the two frameworks can be compared.

The predictions of the two methods will differ only in the case of presuppositions of positive presuppositional sentences asserted on their own: Mercer considers them entailments but not presuppositions, the presuppositional tableaux framework considers them presuppositions.

8.2.4 Monotonicity of Discourses

The fact that presuppositional consequence is monotonic over discourses does not introduce differences with Mercer's framework. Mercer considers the same behaviour when he says that

> Default logic itself does not have the facilities to retract a conjectural inference in a dynamic world when contradictory information is presented. [...] The addition of a Truth Maintenance System [...] would be required to retract conjectural inferences when contradictory non-conjectural information is acquired.
>
> Mercer [27, page71]

This is the same sort of solution that would be required for the cases of refutation of the preferred interpretation considered above.

8.2.5 An Example in Detail

For the same example (71) as above, Mercer's framework operates as follows.

The first proposition of the sequence, $m \rightarrow \neg w$, results in the addition to the context of the implicatures that m, $\neg m$, w, $\neg w$ are possibly true in the context.

The interpretation of the second sentence $m \rightarrow r^{\neg w}$, results in the addition to the context of the implicatures that m, $\neg m$, $r^{\neg w}$, $\neg r^{\neg w}$ are possibly true in the context. When it comes to processing the presupposition case analysis would have to be invoked. However, in this particular case, the implicatures that Mercer suggests using in order to select the appropriate cases are the ones of second sentence, whereas the ones that might have affected the decision procedure for the presuppositions are the implicatures of the first sentence. These have been added to the context as modal statements of possibility, but are no longer available as criteria for the case selection procedure. All the cases that result by applying the selection criteria to the implicatures that m, $\neg m$, $r^{\neg w}$, $\neg r^{\neg w}$ are possibly true in the context come out as non-committal on whether w is true or not. The defaults cannot be blocked by the implicatures of the sentence itself. The implicatures of the previous sentence (that m, $\neg m$, w, $\neg w$ are possibly true in the context) are not enough to block the presupposition, because in the modal form in which they are updated into the context they are compatible with it.

As a result, Mercer's framework gives the wrong prediction that the sentence presupposes that there is no hot water left.

8.3 Anaphora Accounts

8.3.1 DRS Method and Tableaux

DRS sets out to describe the relation between form and meaning. It is not concerned with how this relation is put to work in actual language use.

It is presented as a rewrite of standard predicate logic that can be drawn algorithmically from the syntactic form of natural language sentences.

A DRS K is true in a model M if there exists a function f that maps the discourse referents in U_K to elements of U_M and verifies the conditions in Con_K.

The procedure for obtaining a standard predicate logic formulation of a DRS is given:

- provide an order for discourse referents and an order for conditions
- start translating from the inside out,
- provide an existential quantifier quantifier for each discourse referent

The DRS translation does not allow constant terms directly, but introduces them in translating DRS conditions *Mary(x)* as x =*Mary.*

On the issue of domains, the DRS translation is evaluated over the domain that its quantifiers create. An initial empty domain is assumed, and objects in the domain are added to it on encountering a quantifier.

Considering that a DRS is actually a representation of the logical structure of a natural language sentence in terms of the natural language connectives that appear in it, there is a certain analogy between DRT's definition of discourse markers of a particular DRS and my definition of a local domain for a tableau branch.

For sentence

(50) *John's cat purrs*

the corresponding DRS and tableau representation would be:

y, x
John(y)
cat(x)
poss(y,x)
purrs(x)

$$\overbrace{\begin{array}{|c|c|c|} c(j) & & P(c(j)) \\ j & & \end{array}}^{\boldsymbol{P(c(j))}}$$

Because of the criteria followed for the choice of evaluation for domains, the information on domains contained in a DRS in terms of discourse referents corresponds closely to that represented in a tableau. The set of discourse referents { x, y } is equivalent to the set $EX = \{c(j), j\}$ for the corresponding tableau. The information captured in the conditions of the DRS appears in the tableau in different forms. The basic conditions that identify the linguistic nouns with the discourse referents are implicit in the tableau where each noun is represented by a specific constant term or a function ($y =$ John $= j$,$x =$ cat $= c(j)$; as given by the translation mentioned above). The function $c(j)$ also captures the relation of possession that exists between the cat and John. The condition concerning the verb is explicit as the proposition that is being interpreted in the tableau.

The use of the equality in a DRS to relate two referents that refer to the same object concerns the yet underdeveloped semantic side of the tableau representation, so comparison is difficult. On the one hand, the notation introduced in chapter 5 of adding a subscript to occurrences of $\Uparrow$ to indicate the instance of redundant proposition that they stand for could be extended to the use of $\Uparrow$ given in chapter 6. This or some similar mechanisms may be used eventually in the tableau framework in order to capture syntactically different terms that 'refer' to the same semantic object. On the other hand, DRS has to make extended use of the equality predicate because it uses it to make explicit the binding operations (for instance, in the case of anaphora) that are only left implicit in the tableau framework.

For a sentence like

(52) *Either John has no donkey or his donkey is in the stable*

each of the methods gives the following representations:

<table>
<tr><td colspan="4">x</td></tr>
<tr><td colspan="4">John(x)</td></tr>
<tr><td>¬</td><td><table><tr><td>y</td></tr><tr><td>donkey(y)
poss(x,y)</td></tr></table></td><td>∨</td><td><table><tr><td>z</td></tr><tr><td>donkey(z)
poss(x,z)
in-the-stable(z)</td></tr></table></td></tr>
</table>

$$
\begin{array}{ccc}
\multicolumn{3}{c}{\boldsymbol{\neg\varepsilon(d(j)) \vee I(s, d(j))}} \\
\hline
\begin{array}{r} \neg\varepsilon(d(j)) \\ I(s, d(j)) \end{array} &
\begin{array}{r} \neg\varepsilon(d(j)) \\ \neg I(s, d(j)) \end{array} &
\begin{array}{r} \varepsilon(d(j)) \\ I(s, d(j)) \end{array} \\
\overbrace{\begin{array}{|c|c|c|} j & d(j) & \neg\varepsilon(d(j)) \\ s & & I(s, d(j)) \\ \Uparrow & & \\ d(j) & & \\ \hline \hline \end{array}} &
\overbrace{\begin{array}{|c|c|c|} j & d(j) & \neg\varepsilon(d(j)) \\ s & \Uparrow & \neg I(s, d(j)) \\ \Uparrow & & \end{array}} &
\overbrace{\begin{array}{|c|c|c|} d(j) & & \varepsilon(d(j)) \\ j & & \\ s & & I(s, d(j)) \\ \Uparrow & & \end{array}}
\end{array}
$$

In this case, the relations between the two representations are not so clear. Mostly this is due to the fact that the DRS representation does not allow easy representation of the information equivalent to that apparent in the set NEX. The presupposition has had to be accommodated locally because otherwise the constraints would have been violated. This ensures that the sentence is predicted not to presuppose the existence of a donkey. But this leads to the appearance of two different discourse markers for *John's donkey* in the DRS (y and z). One of these discourse markers refers to the donkey that exists in one of the alternative interpretations of the sentence. The other one does not refer at all, but rather is only used to state the non-existence of the donkey in the other interpretation of the sentence. The information contained in each representation is still basically the same. The difference lies in that the tableau framework has made it explicit what the difference is between these two syntactic occurrences of the condition 'donkey' applied to discourse markers and has this information available explicitly to be used in inference. This allows this information to result in closure of a branch (as in the case of the first branch of the tableau above).

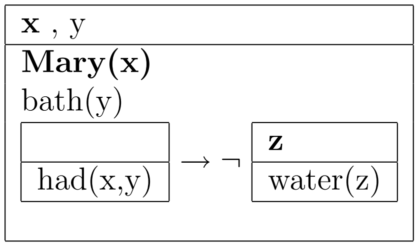
8.3.2 An Example in Detail

The interpretation of the discourse

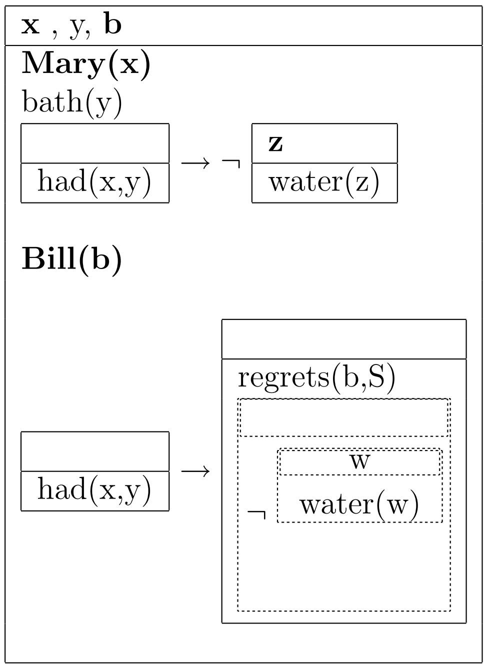
> (75) (*If Mary has had a bath, then there is no hot water left*) $\circ$ (*If Mary has had a bath, then Bill regrets that there is no hot water left*)

allows interesting comparisons.

The DRS representation for the first sentence would be



Adding the second sentence produces an initial representation (with the proper names already correctly placed but before presupposition resolution).



The only resolution that is not logically incompatible with the context is to accommodate the presupposition in its local context, resulting in a representation:

$$\begin{array}{|l|}\hline \mathbf{x}\ , y, \mathbf{b} \\ \hline \mathbf{Mary(x)} \\ \text{bath}(y) \\ \begin{array}{|c|}\hline \\ \hline \text{had}(x,y) \\ \hline\end{array} \rightarrow \neg \begin{array}{|c|}\hline \mathbf{z} \\ \hline \text{water}(z)\\ \hline\end{array} \\ \\ \mathbf{Bill(b)} \\ \\ \begin{array}{|c|}\hline \\ \hline \text{had}(x,y)\\ \hline\end{array} \rightarrow \begin{array}{|l|}\hline \\ \hline \text{regrets}(b,S) \\ \neg \begin{array}{|c|}\hline \mathbf{z}\\ \hline \text{water}(z)\\ \hline\end{array} \\ \\ \hline\end{array} \\ \\ \hline\end{array}$$

It is apparent from this representation that some important information is being lost. Given that both conditionals have the same antecedent, intuitively it seems that the presupposition might have been bound to the DRS that is the consequent of the first conditional, at least in some of the possible interpretations of the sentences (those were Bill regrets that there is no hot water left because Mary has had a bath).

Because it represents explicitly only certain logical interpretations of the discourse Sandt's framework looses the flexibility to represent these interactions between the logical structure of the discourse and the possible bindings of presuppositions.

Chapter 9

Conclusions

9.1 Critical Analysis

9.1.1 The Contributions

The presuppositional expansion rules constitute a step forward in the understanding of the interaction between presupposition and the logical semantics of discourse. This is apparent in the simplicity with which the projection problem finds a solution.

The incremental behaviour that drives the update semantics frameworks is captured without losing the semantics for disjunction. This is achieved by extending the use of tableau from their traditional role as a decision method to include an additional role as a representation for a context of interpretation. The traditional use of tableaux as a decision method allows this representation to be queried for information.

The behaviour of all the connectives (including disjunction) is described satisfactorily by the rules given. No specific rules for each connective are required. The compositionality rules rely on the semantics in general terms. As a result, the same compositionality rules may be applied to other connectives if their semantics can be represented in the same framework in a way that preserves the Coverage Property.

The present framework is designed to capture the incremental construction of a discourse, and therefore it cannot be used to provide an abstract view of the effect of a set of sentences without considering the sequence of updates. This agrees with the intuition that language understanding is an incremental process.

The incremental approach is related to the role of sentence boundaries in interpretation: a sentence boundary forces expansion of all information contained within it before adding any further information.

The defeasible nature of presuppositional inference is modelled in a solid infrastructure that allows a natural definition of reasoning by cases.

The definition of the extended proof language in the preliminary study of the first order case for presuppositions of definite descriptions shows promising avenues of research into the matter of interpretation of quantified statements in informal language use with no specified domain.

The basic structure that underlies the whole framework also shows how the behaviour of presupposition is closely related to that of abduction.

9.1.2 Expressive Power

The framework presented here is not intended to stand as a knowledge representation framework on its own, but rather as a case study of very specific issues carried out with the aim of casting light on the interaction between elementary phenomena that tend not to coexist in more complex frameworks.

For the sake of completeness, a list of the major shortcomings of the framework with respect to expressive power follows.

Linguistic Evaluation of Negation

The issue of representing the negation of complex propositions has been discussed in chapter 4.

Disjunction: Inclusive or Exclusive

The most common natural language use of disjunction carries an implicature that one or the other, but not both disjuncts are true. This is not captured in the framework as it stands. No attempt has been done to represent this feature in the model because it was preferred to retain the symmetry of the β rules in order to study in detail the relation between the semantics and the compositionality of presupposition. The framework may be developed further to account for implicatures of this type.

Quantification

Natural language quantification goes far beyond the quantification sketched here, even beyond quantification in full first order logic.

9.2 Further Work

The semantic implications of the first order case still need a lot of work. This should shed more light on the issues of definites and indefinites, and anaphoric relations in discourse.

The first order case can be studied even further by considering the effects of adding presupposition as defined in the propositional case.

The abductive solution must be explored further in the first order case. The tableaux framework has been shown (Cialdea et al [3]) to be well suited for the formalization of abduction. Further work along these lines could lead to interesting results.

The possible world semantics developed to account for worlds with differing domains can be taken a step further to handle sentences with lexical entries that carry an implicit modal operator. Factives and verbs of propositional attitude, well known presuppositional constructions, fall in this category.

The interaction with belief revision is another interesting avenue of research.

Bibliography

[1] Beaver, David, 'The Kynematics of Presupposition', in: H. Kamp (ed.), *Presupposition*, Dyana Report R2.2.A, Part II, Centre for Cognitive Science, University of Edimburgh, August 1993.

[2] Beaver, David, *What Comes First in Dynamic Semantics*, Institute for Logic, Language and Computation, Amsterdam, 1993.

[3] Cialdea Mayer, Marta and Fiora Pirri, 'First order abduction via tableau and sequent calculi', *Bulletin of the IGPL*, 1, July 1993, pp 99-117.

[4] Clark, H.H., 'Bridging', in: Johnson-Laird, P.N. and Wason, P.C. (eds.), *Thinking*, CUP, Cambridge, 1977.

[5] Clark, H.H. and Havilland, S.E., 'Comprehension and the Given-New Contract', in: Freedle, R. (ed.), *Discourse Production and Comprehension*, Erlbaum, Hillsdale, N.J., (in press 1977).

[6] Fitting, Melvin, *First Order Logic and Automated Theorem Proving*, Springer Verlag, New York, 1989.

[7] Frege, Gottlob, 'On Sense and Reference', in: P. Geach and M.Black (eds.) *Translations from the Philosophical Writings of Gottlob Frege*, Basil Blackwell, Oxford, 1952.

[8] Gazdar, Gerald, *Pragmatics: Implicature, Presupposition and Logical Form*, Academic Press, New York, 1979.

[9] Gazdar, Gerald, 'A Solution to the Projection Problem' in: *Syntax and Semantics 11. Presupposition*, 1979.

[10] Grice, H.P., 'Logic and Conversation', in: A.P. Martinich (ed.),*The Philosph of Language*, OUP, 1985.

[11] Hamblin, C.L., *Fallacies*, Methuen, London, 1970.

[12] Heim, Irene, *The Semantics of Definite and Indefinite Noun Phrases*, PhD Thesis, University of Massachusetts, 1982.

[13] Heim, Irene, 'On the projection problem for presuppositions', in: D. Flickinger et al (eds.), *Proceedings of the Second West Coast Conference on Formal Linguistics*, 1983. Reprinted in: S. Davis (ed.), *Pragmatics*, OUP, New York, 1991.

[14] Heim, Irene, 'Presupposition Projection and the Semantics of Attitude Verbs', *Journal of Semantics*, Volume 9, Number 3, 1992.

[15] Hirst, G. 'Existence Assumptions in Knowledge Representation', *Artificial Intelligence*, 49:199-242,1991.

[16] Hodges, Wilfrid, 'Elementary Predicate Logic', in: Gabbay and Guenther (eds.) *Handbook of Philosophical Logic*, 1989.

[17] Hughes, G.E. and Creswell, M.J., *An Introduction to Modal Logic*, Routledge, New York, 1968.

[18] Kamp, H. and U. Reyle, *From Discourse to Logic*, Kluwer, Dordrecht, 1990.

[19] Karttunen, Lauri, 'Presupposition of Compound Sentences', *Linguistic Inquiry*, Volume IV, Number 2 (Spring 1973).

[20] Karttunen, Lauri, 'Presupposition and Linguistic Context', *Theoretical Linguistics* I, 1974.

[21] Karttunnen, L. and S. Peters 'Conventional Implicature', in: *Syntax and Semantics 11. Presupposition*, 1979.

[22] Kripke, Saul, 'Semantic Considerations on Modal Logic', *Acta Philos.Fenn.* 16,1963. Reprinted in : L.Linsky (ed.) *Reference and Modality*, PUP, 1971.

[23] Langendoen and Savin, 'The Projection Problem for Presuppositions' in: Fillimore and Langendoen (eds.), *Studies in Linguistic Semantics*, New York, 1971.

[24] Lejewsky, C., 'Logic and Existence' *British Journal for the Philosophy of Science*, 5:104-119,1954.

[25] Lewis, David, 'Scorekeeping in a Language Game', *Journal of Philosophical Logic* 8 (1979), 339-359.

[26] Marcu, Daniel, *A Formalism and an Algorithm for Computing Pragamatic Inferences and Detecting Infelicities*, Technical Report CSRI-309, October 1994, Computer Systems Research Institute, University of Toronto.

[27] Mercer, Robert Ernest, *A Default Logic Approach to the Derivation of Natural Language Presuppositions*,Technical Report 87-35, Department of Computer Science, University of British Columbia, Vancouver, B.C.,Canada, 1987.

[28] Mercer, Robert, 'Solving Some Persistent Presupposition Problems', in: *Proceedings of the 12th International Conference on Computational Linguistics*, pp 420-425, 1988.

[29] Mercer, Robert, 'Using Default Logic to Derive Natural Language Presuppositions', in: *Proceedings of the Seventh Biennial Conference of the Canadian Society for the Computational Studies of Intelligence (CSCI/SCEIO)*, pp 14-21, 1988.

[30] Mercer, Robert, 'Deriving Natural Language Presuppositions from Complex Conditionals', in: *Proceedings of the Eighth Biennial Conference of the Canadian Society for the Computational Studies of Intelligence (CSCI/SCEIO)*, pp 114-120, 1990.

[31] Mercer, Robert, 'Presuppositions and Default Reasoning: A Study in Lexical Pragmatics', in: Pustejovsky, J. and Bergler, J. (eds.), *Lexical Semantics and Knowledge Representation*, (First SINGLEX Workshop, Berkeley, 1991), (627) Lecture Notes in AI, Springer Verlag, 1991.

[32] Mercer, Robert, 'Towards a Common Logical Semantics for Presuppositions and Entailment', *Journal of Semantics*, Volume 9, Number 3, 1992, pp 223-250.

[33] Montague, R., 'The Proper Treatment of Quantification in Ordinary English', in: R.H.Thomason (ed.) *Formal Philosphy. Selected Papers of Richard Montague*, Yale University Presss, New Haven, 1974.

[34] Morgan, J.L., 'On the Treatment of Presupposition in Transformational Grammar', in: R.Binnick et al. (eds.) *Papers from the Fifth Regional Meeting of the Chicago Linguistic Society*, University of Chicago, Chicago, Illinois, 1969.

[35] Peirce, Charles Sanders, excerpt from *Philosophical Lectures*, in: *Classics in Logic. Readings in Epistemology, Theory of Knowledge and Dialectics*, Peter Owen, London, 1962.

[36] Pople, Harry E. Jr., 'On the Mechanization of Abductive Logic', *IJCAI* 1973.

[37] Schöter, Andreas, *Evidential Bilattice Logic and Lexical Inference*, Research Paper EUCCS/RP-64, Centre for Cognitive Science, University of Edinburgh, 1994.

[38] Smullyan, Raymond, *First Order Logic*, Springer Verlag, New York, 1968.

[39] Soames, Scott, 'A Projection Problem for Speaker Presuppositions', *Linguistics Inquiry* 10, 1979.

[40] Soames, Scott, 'How Presuppositions are Inherited: A Solution to the Projection Problem', *Linguistic Inquiry* 13, 1982.

[41] Soames, Scott, 'Presupposition', in: Gabbay and Guenther (eds.) *Handbook of Philosophical Logic*, 1989.

[42] Stalnaker, Robert, 'Presuppositions', *Journal of Philosophical Logic* 2 (1973), 447-457.

[43] Stalnalker, Robert, 'Pragmatic Presuppositions', in: M. Munitz and P. Unger (eds.), *Semantics and Philosphy*, New York University Press, New York, 1974.

[44] Stalnalker, Robert, 'Assertion', in: P.Cole (ed.), *Syntax and Semantics 9. Pragmatics*, 1979.

[45] Strawson, P.F., 'On Referring', in: *Mind* 59, 1950.

[46] van der Sandt, *Context and Presuppositions*, Croom Helm, London, 1988.

[47] van der Sandt, Rob A., 'Presupposition Projection as Anaphora Resolution', *Journal of Semantics*, 9, 1992, pp333-377. Helm, London, 1988.

[48] van Fraasen, 'Presupposition, Implication, and Self Reference', *Journal of Philosophy* 65, 1968, pp132-152.

[49] van Fraasen, *Formal Semantics and Logic*, MacMillan, New York, 1971.

[50] Veltman, F. 'Defaults in Update Semantics', in: H. Kamp (ed.), *Conditionals and Belief Revision*, Dyana Report R2.5.A, Centre for Cognitive Science, University of Edimburgh, 1990.