A Definite Clause Grammatical Inversion of Extended Montague Semantics

ἀποθανων ἐτι λαλειται

VOLUME I

(Volume One of Two Volumes)

Roy Ivor Bainbridge

Thesis submitted for the degree of
Doctor of Philosophy
to
The Department of Computer Science,
University of York.

May 1987.

CONTENTS

A DCG INVERSION OF EXTENDED MONTAGUE SEMANTICS

VOLUME I

CONTENTS

A DCG INVERSION OF EXTENDED MONTAGUE SEMANTICS

VOLUME II

ACKNOWLEDGEMENTS

A proposed investigation within a cross disciplinary domain disjoint from other current research endeavours might be greeted with trepidation in many institutions. I am accordingly indebted to Professor Ian Pyle and Dr. Keith Mander for their initial forbearance in accepting my registration, originally as a part time student, at the University of York, and to the latter both for his courage in agreeing to supervise my research and for his encouragement throughout.

Beneficial influences upon my work have been legion and defy enumeration, but I should like to extend my thanks to Dorothy Edgington who first drew my attention to the writings of Richard Montague, while even a cursory reading of the thesis will reveal that my debts to Professor Barbara Partee and Professor Joyce Friedman, neither of whom have I ever met, and to Professor Friedman's collaborators, are incalculable.

Finally I dedicate this thesis to my wife, Marie, without whose financial support in my early days as a philosopher and unswerving faith, patience, tolerance and self sacrifice in time since it would never have seen the light of day.

DECLARATION

A Prototype inversion of the grammar of Montague's PTQ in which some of the tactics adopted in chapter 9 were initially explored has appeared in *Montagovian Definite Clause Grammar*, [B4]. Preliminary enquiries which form the basis of the natural syntax for tense and aspect together with the multi indexed tense logic appearing in chapter 6 were documented in *On the Recursive Generation of Intransitive Verb Phrases and Subordinate Time Relativisation*, [B5]. No other extracts from this thesis have previously appeared.

ABSTRACT

Montague Grammar synthesises complex expressions comprising a fragment of natural language by means of a simultaneous recursive definition of phrasal categories, the extremal clauses constituting a lexicon. A compositional semantics recapitulates each syntactic derivation so as to express the meaning of the complex derived in terms of the meanings of its parts, such meanings being formulated either directly in model theoretic terms or indirectly as translations into a language of higher order intensional logic for which prior model theoretic interpretation is available. Definite Clause Grammar combines an augmented Phrase Structure Grammar with a recursive descent parsing algorithm and provides a convenient medium in which to implement the inverses of Montagovian syntax rules with a view to computational investigation of their behaviour.

This thesis presents a definite clause grammatical inversion designed both to simulate and to assist in the development of an extended Montague Grammar **TMG**. The program suite comprises six modules, four concerned with syntax and two with semantics, the former including a parser **TMDCG**, a string editor **EDIT**, a topic neutral lexicon **GENLEX** and a domain specific lexicon **LEXTMG** while a language of intensional logic translator **LILT** and a thesaurus of primitive intensional logic assignments **TBASE** constitute the latter. Although **TMG** and **TMDCG/LILT** were developed in parallel, exposition will be sequential.

A survey of the philosophical background to Richard Montague's program followed by an exegesis of his best known fragmental analysis commences Volume I which thereafter concentrates on proposed corrections and extensions, culminating in a definition of both the target grammar **TMG** and the language of intensional logic **TIL** employed in its indirect interpretation. After a discussion of the essentials of Montague's general theory leading to a delimitation of the concept "computer implementation of Montague Grammar", Volume II surveys previous bona fide implementations and terminates with an exposition of the strategic features and tactical scope of the current inversion.

INTRODUCTION

¶ Having outlined the autobiographical ontogenesis of the endeavour, this author's preface to the work presents a brief overview of the project described in ensuing pages which is subsumed under the rubric "*Computation in Service to Linguistics*". A digest of the ground covered by each chapter, including an identification of appropriate related work in the area, is then presented seriatim. Details of documentation contained in appendices thereafter completes the apologetics.

Ontogenesis

My interest in Montague semantics was first awakened when, having recently graduated in philosophy, and in response to my enquiry regarding the state of the art in formal semanticist alternatives to the Wittgensteinian and "speech act" approaches to meaning with which I had been imbued, I was handed a barely legible photostat copy of "*Universal Grammar*" (UG), [M5], with the caveat that it would probably prove incomprehensible even if successfully deciphered.

Few topics could prove less tractable to an autodidact, especially when accessible only in the form of Montague's original uncompromising prose, thus comprehension might indeed have eluded me but for the timely publication of Thomason's edition of "*Formal Philosophy*", [T3], and a series of lucid papers by Partee, [P2, P3, P4], for whose limpid clarity as an expositor I shall be eternally grateful; for these were the days before the arrival of Dowty Wall and Peter's seminal introduction, [D9].

As my interest grew, it became apparent to me that so complex was the data to be processed in the design and verification of a Montague grammar, and so intricate the processing mechanism, that here was a subject area the investigation of which could not but benefit from computational assistance. Thoughts similar to these were occurring simultaneously and independently to Friedman and her collaborators. I thus became determined to construct a computer simulation of Montague semantics which would serve as a tool for the investigation and development of the theory, an enterprise which falls squarely under Thompson's rubric, [T5], "*Computation in Service to Linguistics*". Prior to embarking on the venture however I deemed it necessary to acquire a formal background in computer science and accordingly changed faculty with this end in view, recommencing studies in what to me was a novel field. Five years ago the envisaged voyage eventually got under way: this dissertation therefore represents the completion of a personal odyssey.

Overview

Montague's semantic theory is encapsulated in five papers, [M2] ... [M6], published between 1968 and 1972. The definitive characterisation of a Montague grammar appears in **UG**, [M5], while the paradigm exemplification occurs in "*The Proper Treatment of Quantification in Ordinary English*", (**PTQ**), [M6], with "*English as a Formal Language*" (**EFL**), [M4], serving as a prototypical experiment.

A Montague grammar offers a simultaneous recursive definition of the phrasal categories of a language, each syntax rule determining a mode of combination for its input categories and specifying a category for its output. Such a grammar in effect *synthesises* complex expressions from ultimate lexical elements, the synthetic history being represented by an *analysis tree*. To each lexical entry there is assigned an interpretation, and to each syntax rule there corresponds a semantic rule defining an interpretation of the syntactic output in terms of the interpretations of the inputs. In both **UG** and **PTQ** interpretation is allowed by an *indirect* method: basic assignments and the inputs and outputs of all semantic rules constitute expressions in a language **IL** of intensional logic for which a non circular interpretation in model theoretic terms is already available. Montague's papers "*Pragmatics*", [M2], and "*Pragmatics and Intensional Logic*" (**PIL**) represent early attempts to define such an intermediary language, a definitive version of which is given in **PTQ**. Whether interpretation be direct or indirect Montague semantics is *compositional*: the meaning of a compound is a function of the meanings of its parts.

Unless a computer implementation is to synthesise *random* sentences (as does Janssen's experimental generator, [J1]), Montague syntax must first be inverted so that a sentence *analyser* (ie. a parser) may be derived. Provided that the parser is able, on consumption of an input sentence, to synthesise analysis trees comparable to those generated by the original syntax, the compositional semantic rules may be applied during a post order traverse of each tree. Hence the need is for a parser which both *analyses* a sentence and *synthesises* a tree. It is my contention that the conflicting requirements of analysis and synthesis are best reconciled by recourse to the PROLOG *Definite Clause Grammar* (DCG) technique.

A DCG may be characterised as a Phrase Structure Grammar (PSG) having non terminal and pre terminal category symbols augmented by argument places, and containing in the right hand side of rules supplementary goals not limited in function to the consumption of the input string. Indeed, had the terminology not already been appropriated, a DCG might have been called a *generalised phrase structure grammar*. Logical variables in the argument places of a DCG provide for the accommodation of both inherited and synthesised attributes, [K9], or alternatively of both trickling and percolating features, [G5], with equal facility. When executed by the PROLOG interpreter, a DCG performs a left to right, depth first, recursive descent parse whilst simultaneously synthesising specified attributes to be returned as output parameters.

A DCG implementation of an extended Montague grammar is accordingly presented in the form of a suite comprising six modules, viz:

TMDCG: A definite clause grammatical inversion of the syntax rules of an extended Montague grammar.

EDIT: A string editor simulating the effects of Montagovian "structural operations" in the construction of nodal phrases to date.

GENLEX: A lexicon containing topic neutral vocabulary, ie. vocabulary not specific to a particular field of discourse.

LEXTMG: A lexicon containing a superset of the vocabulary particular to Montague's PTQ.

LILT: A language of intensional logic translator and reducer which postorders analysis trees generating equivalents for each nodal phrase in TIL, a tensed superset of IL.

TBASE: A compendium of basic semantic assignments to lexical items.

This suite is by no means the first computational implementation of Montague semantics, being predated by Friedman and Warren's ATN parser, [F3], Landsbergen's Rosetta project parser, [L1], and Janssen's experimental generator, [J1]. My earlier DCG inversion of Montague's PTQ, [B4], upon which the present implementation is based, was however the first published *logic programming* simulation of Montague semantics, and qua DCG the first to represent a Montague grammar in an alternative *grammatical* formalism: moreover it was the first simulation fully to reproduce the details of Montagovian analysis trees and thus wholly to obviate manual intervention in the reconstruction of nodal phrases. The suite may accordingly claim to be the first *exact* simulation of a Montague grammar.

The target grammar, TMG, which is simulated by TMDCG, is a fully tensed extension of the grammar of PTQ which includes published corrections together with earlier extensions such as Rodman's restricted relative clauses, [R4], and Karttunen and Peters' indirect interrogatives, [K4, K5]. The treatment of tense and aspect has its ancestry in the work of Bach, [B2, B3], and Dowty [D6, D8].

Whereas my earlier DCG, MDCG, [B4], did no more than *simulate* Montague's PTQ, the present DCG, TMDCG, was employed in the *development* of TMG, thus endorsing a contention of Ritchie's that reflection upon the exigencies of computer implementation may provide feedback for the tuning of the target grammar. Improvement and development of the underlying linguistic theory is indeed, in my view, the proper objective of a computational investigation. It is a matter of indifference to me whether or not an unadulterated Montague grammar could be incorporated within an autonomous "natural language understanding system" (NLUS); for it remains to be established whether or not such systems, despite their ingenuity and the preposterous extravagance of their claims, have any more to contribute to the serious development of cognitive psychology, or the systematically scientific investigation of the theory of performance, than does the software of the ubiquitous and equally ingenious penny arcade "space invaders" game to the implementation of the strategic defence initiative.

Three separate disciplines, philosophical logic, theoretical linguistics and computer science, contribute presuppositions towards my project, and some familiarity with all three is accordingly a prerequisite for an appreciation of the whole. It seems not unreasonable to presume familiarity with the subject area of the department under whose aegis the research has been undertaken, viz. computer science, but no such presumption would be justified with respect to the other crucial areas: accordingly all necessary background assumptions emanating from these sources will be explicitly introduced and discussed.

The reader is thus assumed to be familiar with the basic tenets of *logic programming*, [B12, C3, K8, P7], *recursive descent parsing*, [A2], and *typed lambda calculus*, [C2]. Likewise the proof theoretic aspects, although not the model theoretic semantics, of first order logic is assumed to be common ground. By contrast no prior knowledge of Montague semantics, nor of the philosophical problems which it attempts to solve, is taken for granted. Readers already acquainted with Montague's PTQ, [M6], and its philosophical precursors need not be detained by the introductory material in chapters 1 and 2 which are innovative only in their style of presentation.

Since an exhaustive chronicle of the evolution of TMG would prove inordinately lengthy, this thesis reflects the dynamics of the heuristic process only where these are germane to the conversion of previously published extensions of Montague grammar to TMG format. Otherwise the target grammar together with its computational representation is presented in its ultimate form.

Digest

Volume I of this thesis comprises six chapters in which the target grammar TMG is fully developed. Consideration of previous computational implementations of Montague grammars and the description of the present DCG inversion is reserved for volume II.

Chapter 1, "Prolegomena to Montague Semantics", introduces the philosophical motivation behind Montague's program and justifies his recourse to a language of higher order intensional logic as an intermediary in the design of a computational semantics for natural language. The classical problems of referential opacity, which undermine any compositional semantics formulated in purely extensional terms, are discussed and solutions to these problems in terms of a fundamental possible worlds semantics outlined. As a prelude to the formal introduction of Montague's higher order IL, the semantics of a language TAL of first order tensed alethic logic are developed on the basis of the standard semantics for the language FOL of first order logic. No novel solutions are proposed and the ground covered approximates to that explored in Thomason's introduction to "*Formal Philosophy*", [T3], chapters 3...5 of Dowty Wall and Peters' "*Introduction to Montague Semantics*", [D9], and the introductory sections of Gallin's "*Intensional and Higher order Modal Logic*", [G4].

In chapter 2, "Montague's PTQ", the language IL is formally defined and the syntactic and semantic rules of the fragment of English analysed in "*The Proper Treatment of Quantification in Ordinary English*" presented. This chapter is again introductory in nature and owes much to the influence of Thomason, [T3], Dowty et al, [D9], and in addition Partee, [P2, P4]. Illustrations are provided of *all* Montague's rules, and the practice is adopted of superimposing syntactic analysis and logical derivation trees so that the "rule by rule" hypothesis may the more easily be verified. Several examples of fully reduced IL translations in tabular form, as advocated by Partee, [P4], are also included. No attempt is made at this stage to discuss the subsumption of PTQ under the general theory of UG because analysis of the general theory is deferred until chapter 7.

Since no purpose would be served by implementing an incorrect grammar, chapter 3, "Corrections and Constraints", commences with a discussion of known inadequacies in PTQ and of suggested solutions, the crucial sources being Bennett, [B7], Friedman, [F5], Janssen, [J3], Partee, [P3, P6] and Thomasson, [T4]. All the solutions considered require Montagovian "structural operations" to access some form of structural description of their inputs and to maintain some form of structural analysis of their outputs, thus the discussion serves to introduce Partee's contention, [P6], that limitations on the legitimate forms of such operations should be imposed. Partee's proposed constraints and innovations are considered and her semi formal requirement that structural operations be formulable as subfunctions in a pseudo programming language is formalised by redefining the operations in terms of executable PROLOG predicates, thus giving rise to a PROLOG normal form (PNF) in which the rules of TMG may be expressed. The chapter ends with the suggestion of an alternative tree labelling convention which expedites translation in the computational analogue.

The proper treatment of restrictive relative clauses, as suggested by Rodman, [R4], and the accommodation of indirect interrogatives after the manner of Karttunen and Peters, [K4, K5], forms the subject matter of chapter 4, "Fundamental Extensions". Since the grammar rules of TMG are to be strictly binary, the published extensions here discussed are massaged into suitable forms and converted to PNF for incorporation in the target grammar.

Chapter 5, "Passivisation, Tense and Aspect", commences with a discussion of Bach's account of the eponymous topics, [B2, B3], together with a review of Dowty's treatment of tense and time adverbials, [D6, D8]. Although neither account is incorporated inviolate in TMG, both contribute significantly to the analysis finally adopted. Dowty's two dimensional tense logic and the multi dimensional tradition originating with Reichenbach, [R1], and developed by both Bull, [B16], and Bruce, [B15], are also considered in this context as precursors of the system of interpretation required for TMG.

A suitably restricted binary recursive mechanism for constructing intransitive verb phrases from auxiliaries and earlier (active or passive) intransitive verb phrases is introduced in chapter 6, "Verb Phrases in

TMG". Given this mechanism, the subject predicate rule must combine noun phrases with *finite* verb phrases themselves derived by combining tenses with *intransitive* verb phrases. A passive intransitive verb phrase results from the combination of a passive transitive verb phrase and an agentive phrase, while a passive transitive phrase combines a passive morpheme with a transitive verb phrase.

This mechanism, which is innovative, was adumbrated in an earlier paper, [B5], and provides for a uniform treatment of tense, aspect and passivisation. Semantic representations of the phrases generated are interpreted in a tense logic based upon that of Dowty, [D6], but modified so as more closely to reflect the intuitions of Reichenbach, [R1], Bull, [B16] and Bruce, [B15]. Although forming part of the final target grammar implemented by TMDCG, the innovations were in fact products of the development of the computational model, this being employed expeditiously to verify the implications of proposed rule formulations.

With the introduction of the tense and aspect rules, the target grammar TMG is complete and attention may be directed to the implementation issues which constitute the subject matter of volume II. Two chapters deal with previous computational implementations of Montague grammar. Chapter 7, "Orthodoxy, Apostacy and Utilisation", commences with an exposition of Montague's general theory as formulated in UG, [M5], in order to identify the sine qua non of a genuine Montague grammar and accordingly to provide criteria for determining whether or not an alleged computational implementation deserves to be so classified. Computational implementations are subdivided into *utilisations* and *investigations*, and it is argued that the orthogonal tradition of "computational compositional semantics", [H7, R7, S1, H6], whatever its intrinsic merits, should be excluded from the category of Montagovian implementations. As an example of a computational utilisation Landsbergen's "Rosetta" project, [L1, L2], which employs a Montague grammar in the context of machine translation, is discussed.

Those computational implementations which may be seen as ancestral to LILT or TMDCG are reserved for chapter 8, "Computational Investigations". Janssen's experimental generator, [J1, J2], is included here because of the affinities between his reduction rules and those of my own language of intensional logic translator, while Friedman and Warren's pioneering paper, [F3], constitutes by far the most significant influence upon the architecture of my syntactic processor: indeed a PROLOG implementation of the Friedman Warren algorithm featured in my earlier Montagovian DCG, [B4]. Modifications to the Friedman Warren algorithm which allow it to support both cataphora and interrogatives are introduced in this chapter, and finally the pros and cons of equivalence parsing, [W2] are discussed. In equivalence parsing the intensional logic translator must be called on a node by node basis as the analysis tree is constructed, thus parser and translator operate in parallel so simulating the mode of a single pass compiler.

In chapter 9, "Inverted Montague Grammar", we consider the design of the DCG analogue of TMG. Once the strategic decision to employ PROLOG has been defended discussion of the architectural details of the program suite commences. The present parser, TMDCG, has some affinity to my earlier prototype MDCG, [B4], but is faster, wider in scope, and significantly different in its handling of left recursion for which it employs the method of Brough and Hogger, [B14]. Syntactic analysis by TMDCG is accomplished with the assistance of its slave modules EDIT, GENLEX and LEXTMG, while LILT together with its slave TBASE undertakes the semantic processing.

A concluding assessment in "Postscript" terminates the thesis with some suggestions regarding future directions for development, following which come the appendices. Complete listings of TMG, TMDCG, EDIT, GENLEX, LEXTMG, LILT and TBASE are included as appendices A...G, while appendix H contains sundry sample analyses.

For the sake of brevity, literature citations are throughout given in the form of alpha-numeric pointers to the bibliography. In the author's opinion the sheer range and volume of such citations renders any alternative form of reference both impractical and unhelpful. Hence the bibliography contains a small residue of works to which no specific reference has been made, however their elimination in order to abstract a separate list of references would contribute nothing to the comprehensibility of the whole.

CHAPTER 1: PROLEGOMENA TO MONTAGUE SEMANTICS

¶ Montague's semantic theory evolved from earlier model theoretic attempts to provide solutions to known problems in philosophical logic. This chapter includes an overview of those problems which constituted the catalyst for earlier endeavours, and summarises those developments in type theory and model theoretic semantics deemed essential to an understanding of Montague's own algebraic approach to natural language definition.

1.1. The Goals of Semantic Theory

A semantic theory for any language must provide for the systematic mapping of sentences in that language to extra linguistic structures having a genuine explanatory value. The phenomena to be explained must at least include such semantically interesting concepts as "truth" "validity", "entailment", "synonymy", and "equivalence"; moreover the form of explanation must be non trivial. If the mapping is to *meta* linguistic rather than *extra* linguistic†1 structures then the latter condition remains unfulfilled unless our understanding of the proposed metalinguistic representation is both independent of and better founded than any intuitive understanding of the object language in question. Pretentious translation into an ad hoc semi formal notation comprehensible only by reference to accompanying or solicited object language redescription involves banal circularity:†2 while translation into a metalinguistic extension of the object language fares little better,serving merely to postpone the requirement for explanation.

To date only the mathematical constructions of the theory of sets have emerged as bona fide contenders for the explanatory role, with alternative modes of incorporation proposed. The direct correlation of set theoretic structures with natural language expressions is a possible tactic. Alternatively the natural language expressions may first be translated into an intermediary formal language which is itself interpretable in set theoretic terms by techniques already available, and already subjected to rigorous scrutiny.

The *indirect* method currently relies for interpretation of the formal language upon a version of the possible world semantics evolved by Kripke, [K11], from the original model theoretic apparatus introduced

†1. Only if ultimate *extra* linguistic interpretation is presumed can the goal of a semantic theory be defined as the expression of strings in an antecedently understood *meta* language.

†2. This objection is similar in substance to that raised by Halvorsen and Ladusaw, [H1].

by Tarski, [T1], which in turn was developed in order to provide a semantics for the language of first order logic. Accordingly this method adopts the possible world model theoretic analysis of semantic concepts developed in the context of artificial languages.

Conversely the *direct* method requires an independent analysis of semantic concepts which could, but need not, diverge from the standard model theoretic formulations, and may accordingly be attractive to those who like Bowers and Reichenbach, [B12], regard the known limitations of possible world semantics as insuperable, and who suspect that no revision oriented towards artificial languages could be appropriate for natural language interpretation. Plainly the onus for justifying divergent analyses rests on the dissenters. Montague adopts a conservative (i.e. non-divergent) version of direct correlation in EFL,[M4], but opts for the indirect alternative in both UG, [M5], and PTQ, [M6].

Whether direct or indirect correlation is preferred, the ultimate rigorously founded structures must prove impotent for explanatory purposes in default of an algorithm for mapping thereinto from the object language. Haphazard correlations tend to introduce corrupt parodies and result in the bogus pseudo logical notational devices rightly derided by Ritchie and Thompson, [R2], who cite the fatuous:

(1) before(leave(mary,the(house)),possible(achieve(mary,anything)))

as a putative rendering of:

(2) Mary left the house before she could achieve anything.

Such reflections suggest that a semantic theory should meet the following conditions:

Condition 1 The theory must introduce a well founded, antecedently understood metalinguistic apparatus.

Condition 2 The apparatus must provide a means for the definition of semantically significant concepts.

Condition 3 The theory must provide for an algorithmic mapping from object language sentences to metalinguistic structures.

The **principle of compositionality** first enunciated by Frege, [F2], requires the semantic interpretation of a compound to be a function of the semantic interpretations of its parts. Condition 3 may be met by a system which extends the compositionality principle to syntax. In such a system sets of basic (lexical) expressions are first determined whereafter compositional syntax rules are applied recursively to combinations of basic expressions and\or previous results in order to generate complex expressions. Semantic representations are assigned directly to the basic expressions, and each syntactic rule S*n* is correlated with a compositional semantic rule T*n* which takes as inputs the semantic representations of the inputs to S*n*. If the semantic representations chosen fulfill conditions 1 and 2, a satisfactory semantic theory emerges.

An important characteristic of this approach is that it rejects any notion of autonomous syntax. If a complete sentence is first generated by the syntax, and a semantic representation subsequently derived by recapitulating the order of rule application with the semantic correlates, the mode of operation is akin to that of a multi- pass compiler. The application of syntax rules is *temporally* prior but the syntax rules themselves are not *logically* prior, since the introduction of a syntax rule into the system is licenced only by its utility for semantic interpretation: the purpose of syntax, as several authors have stressed, (eg. [P4], [D8]), is to provide a basis for semantics . If the syntactic and semantic components of the system are run in parallel, simulating the behaviour of a single pass compiler, even the temporal priority becomes vestigial.

In both UG and PTQ Montague meets the above conditions with a three stage program answering the foregoing description:

Stage 1 A fragment of English, designed to include constructions of major philosophical interest and puzzlement, is defined by means of a compositional syntax.

Stage 2 Each syntax rule is correlated with a compositional semantic rule which maps the syntactic structure to an expression in a language IL of higher order intensional logic.

Stage 3 The language IL is given a model theoretic interpretation in terms of possible world semantics.

As has been intimated, it is the semantic considerations which in such a program determine the syntax; thus prolegomena to Montague semantics must explore the philosophical motivation for the adoption of IL.

1.2. Higher Order Abstraction and Type Theory

Montague's choice of a language of higher order rather than first order logic as the intermediary is based not upon any naive observation that English plainly contains higher order constructions involving quantification over predicates, as for example in the sentence:

(3) The offspring of a hermaphrodite inherit all the characteristics of the parent.

but rather on the recognition that abstraction over predicate variables is needed for a uniform compositional account of "terms" (noun phrases). As Warren, [W1], has remarked, first order methodology correlates a sentence such as:

(4) Every man walks.

with a formula of first order logic viz.

(5) $\forall X(\text{man}(X) \rightarrow \text{walk}(X))$.

but is unable to identify the contributions of the individual elements in the original sentence. The first order analysis reflects Russell's contention, [R10], that:

> "a denoting phrase is essentially *part* of a sentence, and does not like most single words have any significance on its own."

Commenting on this situation Cooper, [C6], suggests that first order logic relates formulae not to English, which *does* have compound noun phrases as constituents, but to an English like substitute in which determiners are operators on sentences. The absence of a constituent by constituent mapping guarantees that condition 3 is infringed.

By contrast, a simplified preview (ignoring intensions) of Montague's proposals would be as follows.

(6) "Every" translates as: $\lambda p \lambda q \forall X(p(X) \rightarrow q(X))$.†3

(7) "Every man" translates as: $\lambda p \lambda q \forall X(p(X) \rightarrow q(X))(\text{man}) = \lambda q \forall X(\text{man}(X) \rightarrow q(X))$.

(8) "Every man walks" translates as: $\lambda q \forall X(\text{man}(X) \rightarrow q(X))(\text{walk}) = \forall X(\text{man}(X) \rightarrow \text{walk}(X))$.

The term "every man" is here treated as denoting the set of properties which every man has. This analysis may be generalised to embrace proper names in a uniform manner: thus the (simplified) translation

†3 Familiarity with Church's calculus of λ-abstraction is assumed, but for quick reference the rules of λ-conversion may be stated as follows where $M^{(x/N)}$ is the result of replacing all free occurrences of x in M by N:

I. If y does not occur in M then λxM may be *renamed* $\lambda yM^{(x/y)}$.

II. If the bound variables in M are distinct from both x and the free variables in N then $\lambda xM(N)$ may be *reduced* to $M^{(x/N)}$.

III. The converse of II.

of "John" becomes:

(9) $\lambda pp(john)$.

denoting the set of all John's properties. A formal definition of "property" will emerge in due course, whereupon the above formulations will be suitably amended.

Notoriously, the availability of predicate variables, and the accompanying possibility of quantification thereover, introduces into a system of logic the potentiality for paradoxes, one of which will serve for illustrative purposes. Russell observes, [R9§80], that some predicates, for example *predicable*, are predicable of themselves and form a well defined sub class Φ such that $\forall P(P \in \Phi \leftrightarrow P(P))$. Predicates not so characterised, which we may describe as *impredicable*, form a disjoint sub class Ψ such that $\forall P(P \in \Psi \leftrightarrow \neg P(P))$. For convenience let "$\xi$" represent "impredicable": it transpires that if $\xi \in \Psi$ then impredicable must be impredicable so $\xi(\xi)$ is true and accordingly $\xi \in \Phi$. If however $\xi \notin \Psi$ then impredicable is not impredicable in which case $\neg\xi(\xi)$ must be true so $\xi \in \Psi$. Indeed as Copi observes, [C9], given the definition $\forall P(\xi(P) \leftrightarrow \neg P(P))$ we may derive the contradiction $\xi(\xi) \leftrightarrow \neg\xi(\xi)$ by universal instantiation.

Russell's own response to the paradoxes was to introduce a hierarchy of "types" where a type is defined, [R9§497], as a "range of significance". The simplest way to introduce such a hierarchy is to identify types with integers such that type $n+1$ indexes all classes having members of type n. A predicative expression such as P(x) then has significance only if the type of P is one higher than the type of x. If types are employed as indices to syntactic categories, and a syntax is defined so that syntactic combination is permitted only when type compatibility guarantees significance, semantically deviant formulae become ill formed, thus effectively eliminating the paradoxes.

Montague adopts not the integer system but a more sophisticated formulation due to Church, [C2], in which any function having an independent variable of type β and a dependent variable of type α must be of type $<\beta\alpha>$. On this account if M is of type α and x is of type β, then λxM is a function of type $<\beta\alpha>$, and $\lambda xM(y)$ is a well formed expression of type α only in case y is of type β. Once again the types are available as indices for syntactic categories. An expression of type $<\beta\alpha>$ may combine with another of type β to generate a resulting expression of type α.

1.3. Extensional Semantics

The fragment of English investigated in PTQ is chosen to include sample sentences known to resist purely extensional analysis: hence the language of IL includes *intensional* features. For reference and comparison purposes it may prove helpful to precede further discussion with a résumé of the extensional semantics of the language FOL of first order logic.

In the ensuing exposition the variables *j, k, m,* and *n* range over the natural numbers, and as usual, the conventions are adopted that Y^X represents the set of all functions having domain X and range Y, while X^n represents the set of all ordered *n*-tuples of members of set X. With these conventions in mind FOL may be defined as follows:

1.3.1. Lexicon for FOL

(Fs1) $Lvar = \{v_n : n \geq 0\}$.

(Fs2) $Lcon = \{f^0_m : m \geq 0\}$.

(Fs3) $Lfun = \{f^n_m : n > 0, m \geq 0\}$.

(Fs4) $Lprop = \{P^0_m : m \geq 0\}$.

(Fs5) $Lpred = \{P^n_m : n > 0, m \geq 0\}$.

Lvar is the set of *individual variables*, *Lcon* the set of *individual constants*, *Lfun* the set of *n-ary functors*, *Lprop* the set of *sentence constants*, and *Lpred* the set of *n-ary predicates*.

1.3.2. Syntax for FOL

(Fs6) If $\tau \in Lvar$ then τ is a term.

(Fs7) If $\tau \in Lcon$ then τ is a term.

(Fs8) If $\tau_1,...,\tau_n$ are terms then $\ulcorner f^n_m(\tau_1,...,\tau_n) \urcorner$ is a term.†4

There are no terms other than those defined.

†4. The signs "⌜" and "⌝" are employed as "Quine corners" which act as selective quotation marks to *mention* both lexical and syncategorematically introduced object language elements while *using* (Greek) metasymbolism with which the former items may be interspersed. Without such a device many of the definienda would be ill formed.

(Fs9) If $\tau_1,...,\tau_n$ are terms then $\ulcorner P^n_m(\tau_1,...,\tau_n) \urcorner$ is an atomic formula.

(Fs10) If τ_j and τ_k are terms then $\ulcorner \tau_j = \tau_k \urcorner$ is an atomic formula.

(Fs11) If $\Phi \in$ *Lprop* then Φ is an atomic formula.

Every atomic formula so defined is also a wff (well formed formula).

(Fs12) If Φ and Ψ are wffs then:

$\ulcorner \neg\Phi \urcorner$ is a wff.

$\ulcorner (\Phi \wedge \Psi) \urcorner$ is a wff.

$\ulcorner (\Phi \vee \Psi) \urcorner$ is a wff.

$\ulcorner (\Phi \rightarrow \Psi) \urcorner$ is a wff.

$\ulcorner (\Phi \leftrightarrow \Psi) \urcorner$ is a wff.

(Fs13) If Φ is a wff and $\upsilon \in$ *Lvar* then:

$\ulcorner \forall\upsilon\Phi \urcorner$ is a wff with Φ the scope of $\forall\upsilon$.

$\ulcorner \exists\upsilon\Phi \urcorner$ is a wff with Φ the scope of $\exists\upsilon$.

There are no other wffs besides those defined.

Since the publication of Tarski's original semantics for FOL, [T1], there have been various equivalent formulations. The one now adopted is in essence that of Kanger, [K2], which I choose both for its perspicuity and for its adaptability in forming a bridge between conventional extensional semantics and the semantics of Montague's IL.

1.3.3. Lexical Semantics for FOL

A primary valuation structure is a pair <*M*, *G*>, where *M* is a model and *G* is a sequence-set.[†5] The model *M* is itself a pair <*D*, *I*> where *D*, the domain of the model, is any non empty set, and *I* is an interpretation function defined over lexical constants.

†5 Tarski introduces *sequences* as functions from the set ω of natural numbers to the domain of individuals, and allows sequences to induce values for both lexical items and terms: induced values thus correspond to the primary valuation and the first three clauses of the secondary. He then defines *satisfaction* of a formulae by a sequence σ such that σ satisfies Φ in those circumstances where the secondary valuation gives the value 1. His definition of truth in a model as satisfaction by all sequences is thus equivalent to ours.

The sequence-set G is defined as follows:

$$G = D^{Lvar}.$$

Thus each $g \in G$ is a function assigning variables to members of the domain hence:

(Ft1) If $\upsilon \in Lvar$ then $g(\upsilon) \in D$.

Moreover, for all $g, g' \in G$, g' is a v_k-variant of g iff for all $j \neq k$ $g'(v_j) = g(v_j)$.

Interpretation of the lexical constants is as follows:

(Ft2) If $\alpha \in Lcon$ then $I(\alpha) \in D$.

(Ft3) If $\zeta \in Lfun$ and ζ has superscript n then $I(\zeta) \in D^{D^n}$.

(Ft4) If $\Phi \in Lprop$ then $I(\Phi) \in \{0,1\}$.

(Ft5) If $\Phi \in Lpred$ and Φ has superscript n then $I(\Phi) \subseteq D^n$.

1.3.4. Expression Semantics for FOL

A secondary valuation structure for expressions defined by the syntax is a pair $<V,<M,G>>$ where V is a valuation function defined as follows:

(Ft6) If $\upsilon \in Lvar$ then $V(\upsilon,M,g) = g(\upsilon)$.

(Ft7) If $\alpha \in Lcon$ then $V(\alpha,M,g) = I(\alpha)$.

(Ft8) If $\zeta \in Lfun$ with superscript n and $\tau_1,\ldots,\tau_n$ are terms then

$$V(\zeta(\tau_1,\ldots,\tau_n),M,g) = I(\zeta)(<V(\tau_1,M,g),\ldots,V(\tau_n,M,g)>).$$

(Ft9) If $\Phi \in Lpred$ with superscript n and $\tau_1,\ldots,\tau_n$ are terms then $V(\Phi(\tau_1,\ldots,\tau_n),M,g) = 1$ iff

$$<V(\tau_1,M,g),\ldots,V(\tau_n,M,g)> \in I(\Phi), \text{ 0 otherwise.}$$

(Ft10) If τ_j and τ_k are terms then $V(\ulcorner\tau_j = \tau_k\urcorner, M,g) = 1$ iff $V(\tau_j, M,g) = V(\tau_k, M,g)$, 0 otherwise.

(Ft11) If $\Phi \in Lprop$ then $V(\Phi,M,g) = I(\Phi)$.

(Ft12) If Φ and Ψ are wffs then:

$$V(\ulcorner\neg\Phi\urcorner, M,g) = 1 \text{ iff } V(\Phi,M,g) = 0.$$

$$V(\ulcorner(\Phi\wedge\Psi)\urcorner, M,g) = 1 \text{ iff both } V(\Phi,M,g) = 1 \text{ and } V(\Psi,M,g) = 1.$$

$V(\ulcorner(\Phi\vee\Psi)\urcorner, M,g) = 1$ iff either $V(\Phi,M,g) = 1$ or $V(\Psi,M,g) = 1$.

$V(\ulcorner(\Phi\rightarrow\Psi)\urcorner, M,g) = 1$ iff either $V(\Phi,M,g) = 0$ or $V(\Psi,M,g) = 1$.

$V(\ulcorner(\Phi\leftrightarrow\Psi)\urcorner, M,g) = 1$ iff $V(\Phi,M,g) = V(\Psi,M,g)$.

(Ft13) If Φ is a wff and υ a variable then:

$V(\ulcorner\forall\upsilon\Phi\urcorner, M,g) = 1$ iff $V(\Phi,M,g') = 1$ for all g' that are υ-variant to g.

$V(\ulcorner\exists\upsilon\Phi\urcorner, M,g) = 1$ iff $V(\Phi,M,g') = 1$ for some g' that is υ-variant to g.

The secondary valuation function is plainly a function of three arguments, the first of which is an expression of FOL, while the second and third cite a model and a "sequence". There is however a convention for abbreviating the functional notation which will be adopted hereafter. For any expression Θ:

$V(\Theta,M,g)$ may be expressed $[\![\Theta]\!]^{M,g}$.

Semantic concepts which may be defined using this extensional model theory include the following:

Φ is true under M

$M \models \Phi$ iff, for all $g \in G$, $[\![\Phi]\!]^{M,g} = 1$.

M is a model of a set Γ of wffs

$M \models \Gamma$ iff, for all $g \in G$, and for all $\Phi \in \Gamma$, $[\![\Phi]\!]^{M,g} = 1$.

Set Γ has a model

$Sat(\Gamma)$ iff, for some M, $M \models \Gamma$.

Γ semantically entails Φ

$\Gamma \models \Phi$ iff, for every M such that $M \models \Gamma$, it is the case that $M \models \Phi$.

Φ is valid

$\models \Phi$ iff, for all M, $M \models \Phi$.

Φ is unsatisfiable

$Unsat(\Phi)$ iff $\models \ulcorner\neg\Phi\urcorner$.

1.4. Extensional Compositionality

The interpretation function *I* assigns *extensions* to the logical constants of FOL relative to a model. Inspection reveals that the extension of an *individual constant* (a member of *Lcon*) is an *individual* ie. a member of the domain *D*, while the extension of the *m*th. *one place predicate* P^1_m (which corresponds roughly to an intransitive verb phrase or common noun phrase in English) is a subset of *D*, that is to say a *set of individuals*. Likewise any *two place predicate* of form P^2_m has as extension a *binary relation* on *D*, ie. a set of ordered pairs of individual members of *D*. Adopting the usual convention whereby {0,1} is the set of truth values, with 0 representing "false" and 1 representing "true", we see that the extension of a *sentence constant* (a member of *Lprop*) is a truth value.

If the semantic interpretation of an expression is identified with its extension, then the principle of compositionality becomes the **principle of extensional compositionality**: the *extension* of a compound must be a function of the *extensions* of its parts. This principle in turn entails a **principle of transparency**: the *extension* of a compound should not vary with the interchange of *coextensive* parts.†6

Since the extension of a sentence is to be a truth value, the truth value of a sentence should not be effected if any component phrase be replaced by another having the same extension. Coextensive *terms* may without controversy be equated in true identity statements, and plainly for all X, Y, p if X=Y then p(X)↔p(Y): hence the principle of transparency for sentences subsumes Quine's **principle of the indiscernibility of identicals**, [Q2]:†7

> "given a true statement of identity, one of its two terms may be substituted for the other in any true statement and the result will be true"

A construction for which the principle of extensional compositionality holds may be described as

†6. Quine, [Q4], distinguishes *codesignative* = referring to the same object from *coextensive* = true of the same object, and implicitly contrasts both with *coveridical* = having a common truth value,in order to isolate *terms* having a common extension, which alone on his view may appear in true identity statements. However these distinctions obscure the generalisation that whether terms, predicates, or sentences are under consideration, commonality of extension is the issue; hence I adopt *coextensive* to cover all cases.

†7. It is often claimed that the principle of transparency for sentences derives from "Leibniz law", or the "*salva veritate*" principle. This seems to me to be a mistake. The most lucid statement of Leibniz principle reads, [L3§7]:

> "*Eadem sunt quorum unum in alterius locum substitui potest, salva veritate, ut triangulum et trilaterum, quadrangulum et quadrilaterum.*"

The import of Quine's principle is that given we have established that two terms are coextensive (codesignative in his usage), we may predict their substitutivity in extensional contexts. By contrast Leibniz principle is one of *identity of indiscernibles*: If we have established that the terms are everywhere substitutable *salva veritate*, then we may conclude *eadem sunt*. Moreover it is far from clear that *eadem sunt* signifies merely that the terms are coextensive. Leibniz own examples quoted above involve terms which on *extensional*: and a language may be classified as extensional if it consists solely of extensional constructions. Although it may be possible, by truncating the vocabulary and restricting the allowable constructions, to identify extensional subsets of English, the English language itself is not an extensional language. Accordingly semantic interpretation cannot be identified with the assignment of extension unless the principle of compositionality is to be abandoned.

Most of the classic examples of failure of extensionality for English involve *referential opacity*, ie. failures of the principle of transparency. The earliest examples identified involved the attempted replacement of coextensive terms within the complements of verbs of *propositional attitude*. In none of the following pairs is the truth of the second sentence guaranteed by the truth of the first.

(10a) An ancient astronomer discovered that the evening star was the morning star.

(10b) An ancient astronomer discovered that the evening star was the evening star. (Frege, [F2])

(11a) George IV learned that Scott was the author of Waverley.

(11b) George IV learned that Scott was Scott. (Russell, [R10])

Examples (10b) and (11b) are presumed false because the complements are trivially true; and given that a trivial truth is one form of necessary truth it is but a short step to discovering that substitutivity of coextensive terms fails in modal contexts.

(12a) Necessarily Hesperus is Hesperus.

(12b) Necessarily Hesperus is Phosphorus. (Apocryphal)†8

(13a) Necessarily if there is life on the evening star there is life on the evening star.

(13b) Necessarily if there is life on the evening star there is life on the morning star. (Quine, [Q2])

Occasionally the direct object of a transitive verb may occupy a referentially opaque position, in which case the verb itself may be classified as an *intensional verb*. Thus, on the assumptions both that the

---

Montague's analysis turn out to be *intensionally* equivalent, ie. they have the same extension in all possible worlds. These examples serve incidentally to highlight the danger in everywhere equating *intension* with *meaning*: "triangle" and "trilateral" do not *mean* the same.

†8. That Frege first noticed the problem of referential opacity in modal contexts by comparing these two sentences has entered into folklore: this appears to be an anachronism. These cases are derived by extracting the complements of the propositional attitudes in Frege's original examples, making a non trivial exchange of proper names for definite descriptions, and subsuming the results within the scope of a modal operator.

commissioner does not know that the dean is also chairman of the hospital board and that we are dealing with the *de dicto* interpretation where the description of the object is understood to be supplied by the subject, we have:

(14a) The commissioner is looking for the dean.

(14b) The commissioner is looking for the chairman of the hospital board. (Quine, [Q4])

There is of course a *de re* interpretation of this pair wherein the *utterer*, not the subject, supplies the description, and for which substitutivity is unproblematic.

Certain *intensional adjectives* likewise introduce opaque contexts.†9 Given that Jones is presently a member of the United States Senate, so that the extension of "colleague of Jones" and "senator" are presently the same, we may generate:

(15a) Smith is visiting a former colleague of Jones.

(15b) Smith is visiting a former senator.

Finally failures of extensionality which do not involve referential opacity are typified by tensed constructions, on the assumption that in a tensed sentence an operator is applied to a corresponding sentence in the simple present. That the extension of the whole is not a function of the extension of the parts is attested by the fact that (16b) is true while (17b) is false despite the fact that (16a) and (17a) are both true.

(16a) Iceland is covered with a glacier.

(16b) Iceland was once covered with a glacier.

(17a) Africa is covered with a glacier.

(17b) Africa was once covered with a glacier. (Thomason, [T3§Introduction])

The only reason why these last examples cannot be formulated as failures of transparency is simply that in English the surface structure of a sentence in a past tense does not contain a component in the simple present, accordingly there is no such component for which substitutivity might fail.

†9 I dissent from the view of Dowty, [D9], that these cases do not involve opacity.

1.5. Possible World Semantics

Frege's solution, [F2], to the problem of referential opacity was to introduce a distinction between the *bedeutung* (reference) and the *sinn* (sense) of an expression, maintaining that in opaque contexts an expression denotes not its normal *bedeutung* but its *sinn*. On this account the extension of a compound may indeed remain a function of the extensions of the elements, with the proviso that in certain syntactically identifiable, (*ungerade*), contexts the latter extensions may prove abnormal. Transparency would then require that in such contexts there should be immunity only to the inter substitution of expressions with equivalent senses. Unfortunately Frege's thesis cannot be further formalised in default of an adequate analysis of *sinn*.

Carnap, [C1], was among the first to attempt a formal analysis of senses, which now take on the guise of *intensions*. Intensions are defined as functions from *possible states of affairs* to extensions: the extension is accordingly the intension valued at the pertaining state of affairs. Carnap compares his possible states of affairs both to Leibnizian possible worlds†10 and to Wittgensteinian *sachverhalten*, [W5], but refrains from offering a concrete definition. This analysis has the distinct advantage of insuring that extensions are determined by intensions, whereas the connection on Frege's original account remains gratuitous, but it is vitiated by the nebulous nature of possible states of affairs, which approximate to complete models of the language in question.†11

In his semantics for modal logic Kripke, [K11], treats the set of possible worlds as a primitive set K of *indices*: thus intensions become functions from *indices* to extensions. The simplest modification to the apparatus for extensional semantics which would reflect this innovation would be to define an *intensional* model M†12 as a triple $\langle D, K, I\rangle$†13 and to redefine I so that it assigned functions from K to previously

†10 References are legion, but Leibniz here includes the head of an audit trail to previous occurrences.

†11 The Leibnizian conception of discarded blueprints in the safekeeping of the Deity is picturesque but equally unhelpful.

†12 Montague preserves a nice distinction in terminology between a *model* which assigns extensions and an *intensional model* or *interpretation* which assigns intensions. Thus model = <intensional-model,specific-index>.

†13 Kripke's formulation is in fact more complex. His definition amounts to $M = \langle\psi, K, R, I\rangle$, where for all $\kappa \in K$, $\psi(\kappa)$ = the domain of individuals existing in world κ, and I is a two place function from worlds and expressions to extensions defined in terms of $\psi(\kappa)$. For all $\kappa, \kappa' \in K$, $\kappa R \kappa' \leftrightarrow \kappa'$ is possible relative to κ, and $[\![\Box\Phi]\!]^{M,\kappa,g} = 1$ iff $[\![\Phi]\!]^{M,\kappa',g} = 1$ for all κ' such that $\kappa R \kappa'$. The accessibility relation R is introduced so that the various Lewis systems, [L5], of modal logic may be simulated: to obtain *S5* R must be reflexive, transitive, and symmetric.

The localisation of domains to worlds allows questions of radical reference failure to be raised: should a sentence containing a referring expression with no extension in a given world be false, [R10], or lacking in truth value, [F2], [S3] in that world? Localised domains are also used to generate counter examples to the "Barcan formula": $\forall x\Box\Phi \leftrightarrow \Box\forall x\Phi$ which postulates equivalence between "everything that *actually* exists in *this* world is Φ in all worlds where it exists" and "in all worlds whatever happens to exist is Φ".

identified extensions. When such a function were valued at the index representing the *status quo*, covert reference would be made to relevant factors upon which the current extension depended; but should the relevant factors be treated as monolithic? The opacity problems in examples (10)...(14) may indeed involve covert references to extensions in this world which may differ in *alternative worlds*, but the covert references in examples (15), (16), and (17) are to *alternative times* in the present world.

Both Montague, [M2], [M3], and Scott, [S2], insist that modality is but one aspect of context sensitivity, and that in general the extension of an expression may depend on *complexes* of relevant factors of which "possible world" is but the one germane to alethic distinctions: Scott dubs such complexes *points of reference*. The set ***K*** should on this view represent the Cartesian product of distinct index sets, and a point of reference should be an ordered *n*-tuple of indices. Opinions regarding the requisite index sets differ†14 depending on the attitude taken to the integration of *pragmatics* with semantics, but at least a set ***W*** of possible worlds and a linearly ordered set ***T*** of moments of time to handle alethic and temporal phenomena will be necessary. Thomason observes, [T3§Introduction], that these two sets are privileged in so far that although extension in a given context may depend additionally on *pragmatic* factors, only ***W*** and ***T*** enter into the assignment of *possible* extensions. Thus we may define:

$$\textit{intensional}\ \text{model}\ M = \langle D, W, T, \leq, I \rangle.$$

$$\text{model} = \langle M, w \in W, t \in T \rangle.$$

Domain ***D*** is now a set of "possible individuals", ie. individuals existing in some world at some time, and convert **FOL** into a language **TAL** of tensed alethic logic.

1.5.1. Lexicon for TAL

The lexicon required is identical to that for **FOL**, ie. (Ts1)...(Ts5) repeats (Fs1)...(Fs5).

Scott, [S2], argues forcefully for a *single* domain of "possible individuals", rather than such localised domains, and this "advice on modal logic" is heeded by Montague.

†14 Scott, (*op cit*), recognises *world, time, position*, and *agent*. Lewis, [L6], adds *audience, indicated object*, and *previous discourse*. The proliferation of index sets is lampooned by Cresswell, [C10], who suggests a *previous drinks* index to handle sentences such as:

"Just fetch your Jim another quart"

In his earlier works Montague himself advocated a uniform treatment of unequivocally semantic issues such as tense and overtly pragmatic issues such as exophoric pronominal reference, and sought to identify *pragmatics* with *indexical semantics*, [M2], [M3]. This identification has not won general support.

1.5.2. Syntax for TAL

The opportunity will first be taken to modify (Fs9) as (Ts9) so as to force *n*-ary predicates to consume their arguments one at a time rather than in *n*-tuples. This mode of combination permits greater uniformity in the statement of the semantics because the translation rule (Tt9) can now be formulated as a case of *functional application.* Although strictly an optional variation in TAL, the mode becomes mandatory for languages having predicates formed by abstraction. With this in mind, the rules may be stated as follows:

(Ts6)...(Ts8) = (Fs6)...(Fs8)

(Ts9a) If τ is a term and Φ an *n*-ary predicate then $\Phi(\tau)$ is an *n*-1 ary predicate.

(Ts9b) If τ is a term and Φ a unary predicate then $\Phi(\tau)$ is an atomic formula.

(Ts10)...(Ts13) = (Fs10)...(Fs13)

(Ts14) If Φ is a wff then :

$\Box\Phi$ is a wff.

fut(Φ) is a wff.

past(Φ) is a wff.

1.5.3. Lexical Semantics for TAL

A primary valuation structure continues to have the form *<M, G>*, and the assignment of extensions to variables by means of some sequence $g \in G$ remains unaffected by the introduction of indices. However the interpretation function I now assigns *intensions* to lexical constants as follows:

(Tt1) = (Ft1)

(Tt2) If $\alpha \in Lcon$ then $I(\alpha) \in D^{W \times T}$.

(Tt3) If $\zeta \in Lfun$ and ζ has superscript n then $I(\zeta) \in {D^{D^n}}^{W \times T}$.

(Tt4) If $\Phi \in Lprop$ then $I(\Phi) \in \{0,1\}^{W \times T}$.

(Tt5) If $\Phi \in Lpred$ and Φ has superscript n then $I(\Phi) \in \{0,1\}^{D^{\cdot^{\cdot^{D^{W \times T}}}}}$ $\}$ *n times* .

The assignments in (Tt2)...(Tt4) are as predicted functions from $W \times T$ to previously defined extensions, but (Tt5) deserves comment. If we consider the case where Φ has superscript 1 then on the basis of the semantics for FOL we might expect:

$$I(\Phi) = \xi \in PD^{W \times T} \text{ such that } \xi(w,t) \subseteq D.$$

But let $\xi(w,t) = E$, then E may be supplanted by its *characteristic function*, ie. that function $\eta \rangle D \rightarrow \{0,1\}$ such that $\eta(\delta) = 1$ iff $\delta \in E$. Thus $\xi(w,t) = \eta \in \{0,1\}^D$ and accordingly $I(\Phi) \in \{0,1\}^{D^{W \times T}}$.

Generalising the argument, we might expect on the basis of the semantics for FOL that where Φ has superscript n then:

$$I(\Phi) = \xi \in PD^{n^{W \times T}} \text{ such that } \xi(w,t) = E \subseteq D^n.$$

By parity of reasoning we may establish that $\xi(w,t) = \eta \in \{0,1\}^{D^n}$; but now we may define for each $\delta \in D$ a function $\theta_{\delta n} \rangle \{\delta\} \rightarrow \{0,1\}^{D^{n-1}}$ such that $\theta_{\delta n}(\delta)(\delta_1,...,\delta_{n-1}) = 1$ iff $\eta(\delta_1,...,\delta_{n-1},\delta) = 1$. $\theta_{\delta n}(\delta)$ is the characteristic function of the set $\{<x_1,...,x_{n-1}>:<x_1,...,x_{n-1},\delta> \in E\}$.

Next we define $\psi_n \rangle D \rightarrow \{0,1\}^{D^{n-1}}$ such that for all $\delta \in D$, $\psi_n(\delta) = \theta_{\delta n}(\delta)$. The range of ψ_n is a set of functions which may be reformulated in like manner as $\{\psi_{n-1}:\psi_{n-1} \rangle D \rightarrow \{0,1\}^{D^{n-2}}\}$. Such reformulation may proceed recursively until we derive:

$$\psi_n \rangle D \rightarrow \{\psi_{n-1}:\psi_{n-1} \rangle D \rightarrow \{\psi_{n-2}:\psi_{n-2} \rangle D \rightarrow \{.....\{\psi_1:\psi_1 \rangle D \rightarrow \{0,1\}\}.....\}\}.$$

Accordingly $\eta(\delta_1,...,\delta_n) = \psi_n(\delta_n)(\delta_{n-1})....(\delta_1)$, and $I(\Phi)$ is a function from $W \times T$ to the set of which ψ_n is a member.

In particular the extension of a two placed predicate is a function from individuals to the extension of a one place predicate: this accords with Montague's treatment of transitive verbs as functions which take terms to make intransitive verbs.

1.5.4. Expression Semantics for TAL

The valuation function V in the secondary valuation structure $<V, <M, G>>$ must now take five arguments such that for any expression Θ, $V(\Theta,M,w,t,g)$ gives the *extension* of Θ: accordingly it becomes necessary to modify the convention previously introduced for assigning significance to the "semantic interpretation" brackets " $[\![$ " and " $]\!]$". We now stipulate that:

$$V(\Theta,M,w,t,g) = [\![\Theta]\!]^{M,w,t,g}.$$

signifies *the extension of* Θ *at* $<w,t>$ *with respect to* ***M***, and also that where κ is a lexical constant:

$$I(\kappa) = [\![\kappa]\!]^{M,g}.$$

signifies *the intension of* κ *with respect to* ***M***. Thus we may establish the following identities:

$$I(\kappa)(w,t) = [\![\kappa]\!]^{M,g}(w,t) = [\![\kappa]\!]^{M,w,t,g}.$$

The revised clauses for expression semantics then become as follows:

(Tt6) If $\upsilon \in Lvar$ then $[\![\upsilon]\!]^{M,w,t,g} = g(\upsilon)$.

(Tt7) If $\alpha \in Lcon$ then $[\![\alpha]\!]^{M,w,t,g} = I(\alpha)(w,t) = [\![\alpha]\!]^{M,g}(w,t)$.

(Tt8) If $\zeta \in Lfun$ with superscript n and $\tau_1,...,\tau_n$ are terms then

$$[\![\zeta(\tau_1,...,\tau_n)]\!]^{M,w,t,g} = [\![\zeta]\!]^{M,g}(w,t)(<[\![\tau_1]\!]^{M,w,t,g},..., [\![\tau_n]\!]^{M,w,t,g}>).$$

(Tt9) If τ is a term and $\Phi \in Lpred$ then $[\![\Phi(\tau)]\!]^{M,w,t,g} = [\![\Phi]\!]^{M,g}(w,t)([\![\tau]\!]^{M,w,t,g})$.

(Tt10) If τ_j and τ_k are terms then $[\![\ulcorner\tau_j=\tau_k\urcorner]\!]^{M,w,t,g} = 1$ iff

$$[\![\tau_j]\!]^{M,w,t,g} = [\![\tau_k]\!]^{M,w,t,g}.$$

(Tt11) If $\Phi \in Lprop$ then $[\![\Phi]\!]^{M,w,t,g} = I(\Phi)(w,t) = [\![\Phi]\!]^{M,g}(w,t)$.

(Tt12) If Φ and Ψ are wffs then:

$$[\![\ulcorner\neg\Phi\urcorner]\!]^{M,w,t,g} = 1 \text{ iff } [\![\Phi]\!]^{M,w,t,g} = 0.$$

$$[\![\ulcorner(\Phi\wedge\Psi)\urcorner]\!]^{M,w,t,g} = 1 \text{ iff both } [\![\Phi]\!]^{M,w,t,g} = 1 \text{ and } [\![\psi]\!]^{M,w,t,g} = 1.$$

$$[\![\ulcorner(\Phi\vee\Psi)\urcorner]\!]^{M,w,t,g} = 1 \text{ iff either } [\![\Phi]\!]^{M,w,t,g} = 1 \text{ or } [\![\psi]\!]^{M,w,t,g} = 1.$$

$$[\![\ulcorner(\Phi\rightarrow\Psi)\urcorner]\!]^{M,w,t,g} = 1 \text{ iff either } [\![\Phi]\!]^{M,w,t,g} = 0 \text{ or } [\![\psi]\!]^{M,w,t,g} = 1.$$

$$[\![\ulcorner(\Phi\leftrightarrow\Psi)\urcorner]\!]^{M,w,t,g} = 1 \text{ iff } [\![\Phi]\!]^{M,w,t,g} = [\![\Psi]\!]^{M,w,t,g}.$$

(Tt13) If Φ is a wff and υ a variable then:

$$[\![\ulcorner\forall\upsilon\Phi\urcorner]\!]^{M,w,t,g} = 1 \text{ iff } [\![\Phi]\!]^{M,w,t,g'} = 1 \text{ for all } g' \text{ that are } \upsilon\text{-variant to } g.$$

$$[\![\ulcorner\exists\upsilon\Phi\urcorner]\!]^{M,w,t,g} = 1 \text{ iff } [\![\Phi]\!]^{M,w,t,g'} = 1 \text{ for some } g' \text{ that is } \upsilon\text{-variant to } g.$$

(Tt14) If Φ is a wff then:

$[\![\ulcorner \Box\Phi \urcorner]\!]^{M,w,t,g} = 1$ iff

$[\![\Phi]\!]^{M,w',t',g} = 1$ for all $w' \in W$ and all $t' \in T$.

$[\![\ulcorner fut(\Phi) \urcorner]\!]^{M,w,t,g} = 1$ iff

$[\![\Phi]\!]^{M,w,t',g} = 1$ for some $t' \in T$ such that $t < t'$.

$[\![\ulcorner past(\Phi) \urcorner]\!]^{M,w,t,g} = 1$ iff

$[\![\Phi]\!]^{M,w,t',g} = 1$ for some $t' \in T$ such that $t > t'$.

The semantic concepts previously defined require relativisation to a pair $<w,t>$ as follows:

Φ is true under $<M,w,t>$

$<M,w,t> \models \Phi$ iff, for all $g \in G$, $[\![\Phi]\!]^{M,w,t,g} = 1$.

$<M,w,t>$ is a model of a set Γ of wffs

$<M,w,t> \models \Gamma$ iff, for all $g \in G$, and for all $\Phi \in \Gamma$, $[\![\Phi]\!]^{M,w,t,g} = 1$.

Set Γ has a model

$Sat(\Gamma)$ iff for some $<M,w,t>$ it is the case that $<M,w,t> \models \Gamma$.

Γ semantically entails Φ

$\Gamma \models \Phi$ iff, for every $<M,w,t>$ such that $<M,w,t> \models \Gamma$, it is the case that $<M,w,t> \models \Phi$.

Φ is valid

$\models \Phi$ iff for all $<M,w,t>$ it is the case that $<M,w,t> \models \Phi$.

Φ is unsatisfiable

$Unsat(\Phi)$ iff $\models \ulcorner \neg\Phi \urcorner$.

1.6. Intensions in Opaque Contexts

The intension assigned by I to an *individual constant* is a function from indices to individual members of the domain D. Following Carnap's suggestion, [C1], such an intension is termed an *individual concept*, and selects a specific individual for each argument pair $<w \in W, t \in T>$. A *one place predicate* is assigned as intension a *property of individuals* ie. a function from indices to the characteristic functions of sets of individuals. In the special case where the value for a given $<w,t>$ is the characteristic function of a *singleton* set, we may consider the intension a *uniquely individuating property* relative to that argument.

Any function from indices to the characteristic functions of n-tuples of individuals (or to their equivalent reformulations) is an *n-ary relation in intension between individuals*, thus the intension assigned to a *two place predicate* is a *binary relation in intension between individuals.* Finally the intension assigned to a *sentence constant* is a function from indices to truth values, conventionally termed a *proposition.*

One glaring anomaly appears in the semantics for **TAL**: although *sentence constants* have intensions, the intension of a structured wff is undefined. A further deficiency is the omission of any mechanism for *referring* to intensions in the object language as required by Frege: the distinction between *intension* and *extension* is formulated only in the metalanguage used for defining semantic rules. In view of these shortcomings, together with the absence of any algorithm for formal translation of English to **TAL**, it would be premature to expect a satisfactory solution to the problems of opacity yet to be available.

Nevertheless exposition of **TAL** makes a useful contribution towards an adequate solution for two reasons. Firstly the semantics for **TAL** are sufficiently close to those for **FOL** to render the modifications immediately comprehensible; and secondly, although not adequate to handle the problems without amendment, the semantics for **TAL** offer sufficient facilities to make the isolated lacunae readily identifiable. Informal augmentation therefore paves the way for a formal exposition of **IL**.

That strongly typed lambda abstraction is to be incorporated has already been intimated, but provisionally we disregard the typing and introduce both abstraction and the mechanism for handling intensions in the object language as ad hoc modifications to **TAL**. Furthermore we assume that "tr(english-expression)" is a well defined function from English expressions to their representations in the language of logic.

As regards lambda expressions it will suffice for the time being to record that if **M** is of type α, x is of type β, and $den(\kappa,M)$ indicates the *possible denotations* of expressions of type κ then:

$$[\![\ulcorner \lambda x M \urcorner]\!]^{M,w,t,g} \in den(\alpha,M)^{den(\beta M)}.$$

The device Montague adopts in order to refer to *intensions* is the operator " ^ " which is defined[†15] in such a way that if Θ is any expression, then for all $<w',t'> \in W \times T$:

$$[\![\ulcorner {}^\wedge\Theta \urcorner]\!]^{M,w,t,g}(w',t') = [\![\Theta]\!]^{M,w',t',g}.$$

that is to say:

$$[\![\ulcorner {}^\wedge\Theta \urcorner]\!]^{M,w,t,g} = [\![\Theta]\!]^{M,g} = \textit{the intension of } \Theta \textit{ with respect to } M.$$

A converse operator " ˅ " is likewise defined over intension denoting expressions such that if Θ has the form ^Ξ then for all $<w,t> \in W \times T$:

$$[\![\ulcorner {}^\vee\Theta \urcorner]\!]^{M,w,t,g} = [\![\Theta]\!]^{M,g}(w,t).$$

Inspection reveals that for all M,w,t,g the principle of "down-up" cancellation holds:

$$[\![\ulcorner {}^{\vee\wedge}\Theta \urcorner]\!]^{M,w,t,g} = [\![\Theta]\!]^{M,w,t,g}.$$

It will be observed that the expression semantics for **TAL** continue to be formulated in terms of the **extensional principle of compositionality**: the extensions of compounds are defined in terms of the extensions of components. With the advent of intension denoting expressions in the object language this principle can be salvaged on the assumption that any expression Θ is replaced in opaque contexts by ^Θ, hence the extension in opaque contexts will be the normal intension as Frege initially suggested.

For preliminary discussion, complications may be avoided if we assume counterfactually both that a definite description is to be handled as an individual constant and that the "is" of identity is to be translated directly as "=". The propositions expressed by the complements of (10) then become those functions from indices to truth values defined as η and η′ in (18).

(18a) $\eta(w,t) = 1$ iff $[\![$ tr(the evening star) $]\!]^{M,g}(w,t) = [\![$ tr(the morning star) $]\!]^{M,g}(w,t)$.

(18b) $\eta'(w,t) = 1$ iff $[\![$ tr(the evening star) $]\!]^{M,g}(w,t) = [\![$ tr(the evening star) $]\!]^{M,g}(w,t)$.

Plainly the primed function returns the value 1 for all $<w,t>$, accordingly it cannot be equated with the unprimed correlate, and whatever other modifications become necessary, we may assume that this ine-

†15. This definition is in fact Thomason's: *vide* [T3], page 259, footnote. Variables are included in the definition as stated, but since the extension of a variable is assigned by a sequence without reference to indices the intension becomes a constant function. If υ is a variable then for all $<w,t>$ $[\![\upsilon]\!]^{M,g}(w,t) = g(\upsilon)$.

quality of η and η' will survive the abandonment of the counterfactual assumptions. An analysis is now required of the effect of placing the intensions identified in (18) within the scope of a verb of propositional attitude. Given the facility for referring to intensions, the propositions expressed in (10) may be identified by the equations in (19) which incorporate definitions in accordance with the **principle of compositionality.**

(19a) $[\![$ tr(An ancient astronomer discovered that the evening star was the morning star) $]\!]^{M,g}(w,t)$ =

$[\![$ tr(discover) $]\!]^{M,g}(w,t')$($[\![$ tr(an ancient astronomer) $]\!]^{M,g}(w,t')$,

$[\![$ tr(the evening star is the morning star) $]\!]^{M,g}$).

(19b) $[\![$ tr(An ancient astronomer discovered that the evening star was the evening star) $]\!]^{M,g}(w,t)$ =

$[\![$ tr(discover) $]\!]^{M,g}(w,t')$($[\![$ tr(an ancient astronomer) $]\!]^{M,g}(w,t')$,

$[\![$ tr(the evening star is the evening star) $]\!]^{M,g}$).

Transparency now requires no more than the substitutivity of *intensionally* equivalent complements. According to this analysis, where Φ translates a verb of propositional attitude we require:

(20) $I(\Phi) \in \left(\{0,1\}^{D \times \{0,1\}^{W \times T}} \right)^{W \times T}$.

and this indeed becomes Montague's proposal.

A similar analysis†16 is available for *intensional adjectives*. If we stipulate that:

(21) $I(\text{tr(former)}) \in (\{0,1\}^{D})^{\left((\{0,1\}^{D})^{W \times T} \right)^{W \times T}}$.

then:

(22) $[\![$ tr(former colleague of jones) $]\!]^{M,g}(w,t)$ =

$[\![$ tr(former) $]\!]^{M,g}(w,t)$($[\![$ tr(colleague of Jones) $]\!]^{M,g}$) =

η such that $\eta(\delta) = 1$ iff $[\![$ tr(colleague of jones) $]\!]^{M,g}{w,t'} = 1$ for some $t' < t$.

Since "former" now requires an intension as argument, substitutivity cannot derive (15b) from (15a).

With regard to modal contexts, we define:

†16. Due simultaneously to Kamp and Parsons, *vide* [G4], [D9].

(23) $I(\text{tr(necessarily)}) \in \{0,1\}^{(\{0,1\}^{W\times T})}$.

By schematising the embedded propositions in (13a) and (13b) as "^Φ" and "^Ψ" we may identify the propositions expressed by the wholes using the equations:

(24) $[\![\text{tr(necessarily)}(^\wedge\Phi)]\!]^{M,g}(w,t) = [\![\text{tr(necessarily)}]\!]^{M,g}(w,t)([\![\Phi]\!]^{M,g})$.

(25) $[\![\text{tr(necessarily)}(^\wedge\Psi)]\!]^{M,g}(w,t) = [\![\text{tr(necessarily)}]\!]^{M,g}(w,t)([\![\Psi]\!]^{M,g})$.†17

These propositions plainly differ since, for reasons already discussed, $[\![\Phi]\!]^{M,g} \neq [\![\Psi]\!]^{M,g}$ even though at the current world w $[\![\Phi]\!]^{M,w,t,g} = [\![\Psi]\!]^{M,w,t,g}$.

Before correcting the counterfactual assumptions made earlier we observe first that a definite description is but one form of noun phrase, and that the denotations of noun phrases were provisionally introduced at (7) as denoting *sets of properties*, where "property" was undefined. A property of individuals has now been identified with a function from indices to the characteristic functions of subsets of individuals, and should accordingly be denoted by an expression of form "$^\wedge P^1_m$": but such a function does not accept *term* denotations as arguments, although its extensionalisation, referred to by "$^{\vee\wedge}P^1_m$", does. If the variables of abstraction in (6)...(9) are to range over *properties*, then their values must be extensionalised in order to licence syntactic combination with terms. Hence (6)...(9) must be reformulated:

(26) tr(every) = $\lambda p \lambda q \forall X(^\vee p(X) \rightarrow {}^\vee q(X))$.

(27) tr(every man) = $\lambda p \lambda q \forall X(^\vee p(X) \rightarrow {}^\vee q(X))(^\wedge man')$

$\lambda q \forall X(^{\vee\wedge} man'(X) \rightarrow {}^\vee q(X))$

$\lambda q \forall X(man'(X) \rightarrow {}^\vee q(X))$.

(28) tr(every man walks) = $\lambda q \forall X(man'(X) \rightarrow {}^\vee q(X))(^\wedge walk')$

$\forall X(man'(X) \rightarrow {}^{\vee\wedge} walk'(X))$

$\forall X(man'(X) \rightarrow walk'(X))$.

(29) tr(Hesperus) = $\lambda p\, ^\vee p(hesperus')$.

Perhaps it may be timely to mention explicitly a convention which has been adopted implicitly and

†17. As Dowty observes, [D9], Montague retains the conventional syncategorematic □ and in effect employs □Φ as an abbreviation for tr(necessarily)(^Φ)

which is employed by Montague and most logicians. To improve readability, and where only a small subset of available symbols is required, symbols with subscripts and/or superscripts may be supplanted by less forbidding colloquial forms, eg. X, Y, Z for v_2, v_3, v_4. When an English word translates directly into a lexical constant, the colloquial form for that constant is a primed variant of the word itself.

Note also that the assembly of components by functional application typically triggers a *reduction* process as a result of which the original elements cease to be separately identifiable.

For any Φ a *set of* Φ may be identified with a function $\zeta \in \{0,1\}^{\Phi}$, while a *property of* Φ is a function $\eta \in (\{0,1\}^{\Phi})^{W\times T}$. A *set of properties of individuals* should accordingly be a function:

(30) $\theta \in \{0,1\}^{\left((\{0,1\}^{D})^{W\times T} \right)}$.

and such a function indeed becomes the *extension* of a noun phrase on Montague's analysis. It follows moreover that the *intension* of a noun phrase should be a *property of properties of individuals*, ie. a function:

(31) $\xi \in \left(\{0,1\}^{\left((\{0,1\}^{D})^{W\times T} \right)} \right)^{W\times T}$.

An *intensional transitive verb* must acquire an *extension* capable of accepting such a function as its argument, consequently Montague concludes that where Φ translates a transitive verb:†18

(32) $I(\Phi) \in (\{0,1\}^{D})^{\left(\left(\{0,1\}^{\left((\{0,1\}^{D})^{W\times T} \right)} \right)^{W\times T} \right)^{W\times T}}$.

so that the analysis which emerges for (14a) becomes:

(33) $[\![$ tr(the commissioner) $]\!]^{M,g}(w,t)($ $[\![$ tr(look for the dean) $]\!]^{M,g})$

where $[\![$ tr(look for the dean) $]\!]^{M,g}(w,t) =$ $[\![$ tr(look for) $]\!]^{M,g}(w,t)($ $[\![$ tr(the dean) $]\!]^{M,g})$.

There need now be no expectation that tr(the dean) and tr(the chairman of the hospital board) be substitut-

†18. A uniform analysis is given of *all* transitive verbs, although the motivation is to block the substitutivity of objects having merely extensional equivalence, and so accommodate the *de dicto* interpretation, in the case of *intensional* transitive verbs. As already mentioned, even these verbs have an alternative *de re* reading wherein substitutivity of coextensive object phrases is unexceptionable, while with *extensional* transitive verbs the *de re* reading is the *only* one. Exegesis of Montague's method for allowing alternative readings in the first case, and eliminating unwanted *de dicto* readings by means of a "meaning postulate" in the second must be deferred.

able.

Once we have established an appropriate translation for the (singular) definite article we will be in a position fully to develop this translation. Montague's analysis conforms at this point to Russell's theory of definite descriptions, [R10].

(34) tr(the)(sing.) = $\lambda p \lambda q \exists Y(\forall X({}^{\vee}p(X) \leftrightarrow X=Y) \wedge {}^{\vee}q(Y))$.

thus:

(35) tr(the dean) = $\lambda p \lambda q \exists Y(\forall X({}^{\vee}p(X) \leftrightarrow X=Y) \wedge {}^{\vee}q(Y))[{}^{\wedge}dean']$

$= \lambda q \exists Y(\forall X(dean'(X) \leftrightarrow X=Y) \wedge {}^{\vee}q(Y))$.

Since the definite description in subject position has no bearing on the opacity in (14) we may for simplicity assume that the commissioner is named Henry. A complete reduced translation of (14a) then becomes:

(36) [1] $\lambda p\, {}^{\vee}p(henry')[{}^{\wedge}look\text{-}for'({}^{\wedge}\lambda q \exists Y(\forall X(dean'(X) \leftrightarrow X=Y) \wedge {}^{\vee}q(Y)))]$

[2] $look\text{-}for'({}^{\wedge}\lambda q \exists Y(\forall X(dean'(X) \leftrightarrow X=Y) \wedge {}^{\vee}q(Y)))(henry')$

[3] $look\text{-}for'(henry', {}^{\wedge}\lambda q \exists Y(\forall X(dean'(X) \leftrightarrow X=Y) \wedge {}^{\vee}q(Y)))$.

The last stage in this reduction merely expresses the functional notation $f(y)(x)$ in the more familiar relational form $f(x,y)$.†19

Ever since the time of Plato it has been customary to distinguish between the "is" of predication and the "is" of identity. Montague rejects this distinction and treats "to be" uniformly as a transitive verb with the translation:

(37) tr(be) = $\lambda n \lambda e\, {}^{\vee}n({}^{\wedge}\lambda Z(e=Z))$†20

If once more for simplicity we assume a primitive predicate "author of Waverley"†21 we may

†19. It transpires that the crucial characteristic of a *de dicto* reading of (14a) is the occurrence of "∃Y" within the scope of "^", thus if Montague is to allow an alternative interpretation there must be a means to generate:

$\exists Y(\forall X(dean'(X) \leftrightarrow X=Y) \wedge look\text{-}for'(henry', {}^{\wedge}\lambda p\, {}^{\vee}p(Y)))$.

†20. Those familiar with Montague's original works will notice an iconic change in the variables used, eg. "n" in place of italic capital P ranging over properties of properties of individual concepts. The reason for the change, and the significance of the distinction between lower case and capitalised variables, will become apparent once PROLOG implementation has been considered.

†21. Without this assumption we require the introduction of a relative clause, ie. "Scott was the author of Waverley" would be paraphrased "Scott was the man who wrote Waverley". A Montagovian translation of (11) would then be:

(i) $\exists X \forall Y(man'(Y) \wedge past(write'(waverley')(Y)) \leftrightarrow Y=X) \wedge scott'=X)$.

The definite description "the man who wrote Waverley" which is a constituent in this sentence may, (contra Russell), be isolated as:

(ii) $\lambda q \exists X \forall Y(man'(Y) \wedge past(write'(waverley')(Y)) \leftrightarrow Y=X) \wedge q(X))$.

reconstruct Montague's treatment of the complement of (11a) as follows:

(38) $[\![\text{tr(Scott was the author of Waverley)}]\!]^{M,g}(w,t) =$

$[\![\text{tr(Scott)}]\!]^{M,g}(w,t)([\![\text{tr(is the author of Waverley)}]\!]^{M,g})$ where

$[\![\text{tr(is the author of Waverley)}]\!]^{M,g}(w,t) =$

$[\![\text{tr(be)}]\!]^{M,g}(w,t)([\![\text{tr(the author of Waverley)}]\!]^{M,g})$.

Given the translations of elements already introduced, we may identify the corresponding reduced translation for the whole thus:

(39) [1] $\lambda p {}^{\vee}p(\text{scott}')[{}^{\wedge}\lambda n \lambda e[{}^{\vee}n({}^{\wedge}\lambda Z(e=Z))][{}^{\wedge}\lambda p \lambda q \exists Y(\forall X({}^{\vee}p(X) \leftrightarrow X=Y) \wedge {}^{\vee}q(Y))[{}^{\wedge}\text{a-o-w}]]]$

[2] $\lambda p {}^{\vee}p(\text{scott}')[{}^{\wedge}\lambda n \lambda e[{}^{\vee}n({}^{\wedge}\lambda Z(e=Z))][{}^{\wedge}\lambda q \exists Y(\forall X(\text{a-o-w}(X) \leftrightarrow X=Y) \wedge {}^{\vee}q(Y))]]$

[3] $\lambda p {}^{\vee}p(\text{scott}')[{}^{\wedge}\lambda e[\lambda q \exists Y(\forall X(\text{a-o-w}(X) \leftrightarrow X=Y) \wedge {}^{\vee}q(Y))[{}^{\wedge}\lambda Z(e=Z)]]]$

[4] $\lambda p {}^{\vee}p(\text{scott}')[{}^{\wedge}\lambda e[\exists Y(\forall X(\text{a-o-w}(X) \leftrightarrow X=Y) \wedge \lambda Z(e=Z)[Y])]]$

[5] $\lambda p {}^{\vee}p(\text{scott}')[{}^{\wedge}\lambda e \exists Y(\forall X(\text{a-o-w}(X) \leftrightarrow X=Y) \wedge e=Y)]$

[6] $\lambda e \exists Y(\forall X(\text{a-o-w}(X) \leftrightarrow X=Y) \wedge e=Y)[\text{scott}']$

[7] $\exists Y(\forall X(\text{a-o-w}(X) \leftrightarrow X=Y) \wedge \text{scott}'=Y)$.

1.7. Residual Problems

What now of the apocryphal example (12)? A reduced translation for (12a) may be provided directly using the facilities already available:

(40) [1] $\lambda p {}^{\vee}p(\text{phosphorus}')[{}^{\wedge}\lambda n \lambda e {}^{\vee}n({}^{\wedge}\lambda Z(e=Z))[{}^{\wedge}\lambda p {}^{\vee}p(\text{hesperus}')]]$

[2] $\lambda p {}^{\vee}p(\text{phosphorus}')[{}^{\wedge}\lambda e \lambda p {}^{\vee}p(\text{hesperus}')[{}^{\wedge}\lambda Z(e=Z)]]$

---

from which it is apparent that only the final conjunct in (i) varies with alternative predications.

Since both the semantics for English and the semantics for the intermediary language of logic are to be treated compositionally, it is interesting to compare the semantics of definite descriptions at both levels.

The *proposition* expressed by (i) is that function η from indices to truth values such that for all $\langle w,t \rangle \in W \times T$ it is the case that $\eta(w,t) = 1$ iff for some $t' < t$, some sequence g, and all y-variants g' such that $g(x)=g'(y)$:

(iii)

(a) $[\![\text{write}]\!]^{M,g'}(w,t')([\![\text{Waverley}]\!]^{M,g'}(w,t'))(g')(y)) = 1$ iff

(b) $[\![\text{write}]\!]^{M,g}(w,t')([\![\text{Waverley}]\!]^{M,g}(w,t'))(g)(x)) = 1$ and

(c) $[\![\text{Scott}]\!]^{M,g}(w,t) = g(x)$.

Since (c) embodies the particular predication, the essence of the definite description is encapsulated in (b). Apparently the intension of a definite description must embody a function from indices to the characteristic functions of sets of members of *D* which for appropriate arguments returns the characteristic function of a *singleton*. This should come as no surprise since a definite description identifies its referent by recourse to a uniquely satisfied predication. In functional terms, where the unique predication is Φ we require that $\Phi(w,t)\eta(w,t) = 1$.

[3] $\lambda p\check{}p(\text{phosphorus}')[\hat{}\lambda e\lambda Z(e{=}Z)[\text{hesperus}']]$

[4] $\lambda p\check{}p(\text{phosphorus}')[\hat{}\lambda e(e{=}\text{hesperus}')]$

[5] $\lambda e(e{=}\text{hesperus}')[\text{phosphorus}']$

[6] $\text{phosphorus}'{=}\text{hesperus}'$.

At first sight the apocryphal case seems to be no longer problematic since the intensions of the individual constants in the fully reduced translation are *individual concepts*, ie. functions from indices to individuals. The contingency of the embedded sentence in (12b), and hence the falsehood of the whole, might be presumed demonstrable from the fact that distinct functions may generate distinct values at many worlds, albeit coinciding at some.

There is however a complication. Kripke, [K12], has argued that cross world reidentification requires that proper names be treated as *rigid designators*, ie. constant functions which identify the same individual in all possible worlds.†22 Hence if the embedded sentence in (12b) is true at all it must be *necessarily* true, and so (12b) cannot be false! Indeed this sentence exemplifies the controversial *necessary a posteriori*: the necessary truth which can be discovered only by experience.

It would be no embarrassment to discover that any true identity statement involving rigid designators constitutes a *necessary* truth were it not for the fact that in possible world semantics *all* necessarily true propositions are intensionally equivalent: a necessary truth is the characteristic function of $W{\times}T$. Intensionally equivalent expressions are however substitutable *salva veritate* in all opaque contexts, including propositional attitudes, hence the following pairs should be mutually derivable:

(41a) An ancient astronomer discovered that Hesperus was Phosphorus.

(41b) An ancient astronomer discovered that Hesperus was Hesperus.

(42a) The first astronomer was unaware that Hesperus was Phosphorus.

(42b) The first astronomer was unaware that Hesperus was Hesperus.

Since coextensive proper names are to be treated as cointensive, their substitutivity should not be limited to appearances within necessary truths. Consequently the interderivability of the following

†22. To be precise Kripke differentiates between *strongly rigid* and *rigid designators* , where the former denote necessary existents (numbers?), while the latter denote the same individual in all worlds wherein the individual exists.

becomes predictable:

(43a) Philip was unaware that Tully denounced Cataline.

(43b) Philip was unaware that Cicero denounced Cataline. (Quine, [Q2])

Relativisation of extensions to indices proves effective provided that it serves to characterise non empty proper subsets of indices, but the ploy becomes vacuous at the limits where a constant value (0 or 1) is returned for all points of reference. Considerable effort has been devoted to the resolution of such problems, one tactic being the introduction of *impossible* worlds, [C10]. If we recognise $U = W \cup H$, where H is a non empty set of *impossible* worlds, and regard the set of indices as notionally $U \times T$, then $W \times T$ is indeed a proper subset of indices. A survey of tentative solutions is not a prerequisite for further exposition of Montague semantics, and accordingly will not be presented: it is sufficient to record that at present possible world semantics is no panacea.

Faced with the residual problems the faint hearted are tempted to abandon the possible worlds approach; but had Carnap lost faith in model theory at the time when the problems of alethic logic were proving insuperable the dramatic advances of the last two decades would never have been achieved. There is accordingly a sound precedent for pressing on in the possible world semantics tradition, believing that in due course the residual problems will not prove intractable.

CHAPTER 2: MONTAGUE'S PTQ

¶ The essential goal of Montague semantics is to provide for the interpretation of all the well formed phrases of a language in model theoretic terms. Montague's PTQ grammar achieves this goal indirectly for a fragment of English by mapping phrases onto expressions in the language IL of intensional logic for which a model theoretic interpretation is already available. In PTQ grammar, compositional syntax rules for constructing phrases from constituents correspond 1:1 with compositional translation rules which construct IL representations of wholes from the IL representations of parts. After describing both the language IL and the grammar of the PTQ fragment, the power and sensitivity of the method is illustrated by the superimposition of analysis and translation trees for several classic examples.

2.1. Montague's IL

Before turning attention to the grammar of the PTQ fragment it will be prudent first to establish the credentials of Montague's language IL of strongly typed higher order intensional logic, since its availability as an intermediary for semantic interpretation depends upon the rigour of its own formalisation. It should be stressed that the use of IL as an intermediate vehicle in Montague semantics, even given its integrity, is not mandatory but serves only as a convenience to obviate direct mapping from English into model theoretic constructs. We commence exposition with an account of the system of types used as indices for syntactic categories.

2.1.1. Semantic Types for IL

Let **e**, **t** and **s** be any three distinct objects other than ordered pairs or triples. The set *Type* of semantic types is the smallest set satisfying the recursive definition:

e ∈ *Type*.

t ∈ *Type*.

If **a**, **b** ∈ *Type* then <**ab**> ∈ *Type*.

If **a** ∈ *Type* then <**sa**> ∈ *Type*.

Objects of type **e** are to be *possible individuals*, ie. members of the domain of an intensional model. *Truth* *values* will be objects of type **t**, while objects of type <ab> will be functions from objects of type a to objects of type **b**. Since s $\notin$ *Type* there can be no objects of type s, but objects of type <sa> will be functions from indices to objects of type **a**. Given the availability of types to serve as indices to syntactic categories, the syntax of IL can now be stated with elegant economy.

2.1.2. Lexicon for IL

For each type a $\in$ *Type*:

(ILs1) $Var_{\mathbf{a}} = \{v_{n,\mathbf{a}} : n \geq 0\}$

(ILs2) $Con_{\mathbf{a}} = \{c_{n,\mathbf{a}} : n \geq 0\}$.

where $Var_{\mathbf{a}}$ is the set of variables of type a, and $Con_{\mathbf{a}}$ is the set of (non logical) constants of type a.

2.1.3. Syntax for IL

If $ME_{\mathbf{a}}$ is understood to be the set of *meaningful expressions* of type **a**, then the meaningful expressions of IL are the members of:

$$\cup_{\mathbf{a} \in Type} ME_{\mathbf{a}}.$$

The set is defined recursively as follows:

(ILs3) If $\alpha \in Var_{\mathbf{a}}$ then $\alpha \in ME_{\mathbf{a}}$.

(ILs4) If $\alpha \in Con_{\mathbf{a}}$ then $\alpha \in ME_{\mathbf{a}}$.

(ILs5) If $\alpha \in ME_{\mathbf{a}}$ and $\upsilon \in Var_{\mathbf{b}}$ then $\lambda\upsilon\alpha \in ME_{<\mathbf{ba}>}$.

(ILs6) If $\alpha \in ME_{<\mathbf{ab}>}$ and $\beta \in ME_{\mathbf{a}}$ then $\alpha(\beta) \in ME_{\mathbf{b}}$.

(ILs7) If $\alpha, \beta \in ME_{\mathbf{a}}$ then $\alpha = \beta \in ME_{\mathbf{t}}$.

(ILs8) If $\Phi, \Psi \in ME_{\mathbf{t}}$ then:

$$\ulcorner \neg\Phi \urcorner \in ME_{\mathbf{t}}.$$

$$\ulcorner (\Phi \wedge \Psi) \urcorner \in ME_{\mathbf{t}}.$$

$$\ulcorner (\Phi \vee \Psi) \urcorner \in ME_{\mathbf{t}}.$$

$$\ulcorner (\Phi \rightarrow \Psi) \urcorner \in ME_{\mathbf{t}}.$$

$$\ulcorner (\Phi \leftrightarrow \Psi) \urcorner \in ME_{\mathbf{t}}.$$

(ILs9) If $\Phi \in ME_t$ and for some a, $\upsilon \in Var_a$ then:

$\ulcorner \forall \upsilon \Phi \urcorner \in ME_t$.

$\ulcorner \exists \upsilon \Phi \urcorner \in ME_t$.

(ILs10) If $\Phi \in ME_t$ then:

$\ulcorner \Box \Phi \urcorner \in ME_t$.

$\ulcorner fut(\Phi) \urcorner \in ME_t$.

$\ulcorner past(\Phi) \urcorner \in ME_t$.

(ILs11) If $\alpha \in ME_a$ then $\ulcorner {}^{\wedge}\alpha \urcorner \in ME_{<sa>}$.

(ILs12) If $\alpha \in ME_{<sa>}$ then $\ulcorner {}^{\vee}\alpha \urcorner \in ME_a$.

There are no other members of ME_a besides those so defined.

2.1.4. Possible Denotations

Both primary valuation of lexical items and secondary valuation of expressions are to be governed by the semantic typing, thus given an intensional model:

$$M = <D, W, T, \leq, I>$$

wherein ***D*** is a domain of possible individuals, ***W*** a set of possible worlds, ***T*** a set of moments in time ordered by ≤, and ***I*** an interpretation function, we first define recursively for each type a a set ***den***(a,***M***) of *possible denotations* of type a relative to ***M***.

$den(e,M) = D$.

$den(t,M) = \{0,1\}$.

$den(<ab>,M) = den(b,M)^{den(a,M)}$.

$den(<sa>,M) = den(a,M)^{W \times T}$.

The set ***sen***(a,***M***) of *senses* of type a relative to a model ***M*** is defined as:

$sen(a,M) = den(<sa>,M)$

and a primary valuation structure ***<M, G>*** together with a secondary valuation structure ***<V, <M, G>>*** are defined as heretofore such that for any expression α:

$V(\alpha,M,w,t,g) = [\![\alpha]\!]^{M,w,t,g}$ = the extension of α at $<w,t>$ with respect to M.

2.1.5. Lexical Semantics for IL

(ILt1) If $\upsilon \in Var_a$ then $g(\upsilon) \in den(a,M)$.

(ILt2) If $\alpha \in Con_a$ then $I(\alpha) \in sen(a,M) = den(a,M)^{W\times T}$.

2.1.6. Expression Semantics for IL

(ILt3) If $\upsilon \in Var_a$ then $[\![\upsilon]\!]^{M,w,t,g} = g(\upsilon)$.

(ILt4) If $\alpha \in Con_a$ then $[\![\alpha]\!]^{M,w,t,g} = I(\alpha)(w,t)$.

(ILt5) If $\alpha \in ME_a$ and $\upsilon \in Var_b$ then $[\![\lambda\upsilon\alpha]\!]^{M,w,t,g} = \eta \in den(a,M)^{den(b,M)}$ such that for all $\beta \in den(b,M)$, $\eta(\beta) = [\![\alpha]\!]^{M,w,t,g'}$ where g and g' differ *at most* in that $g'(\upsilon) = \beta$.

(ILt6) If $\alpha \in ME_{<ab>}$ and $\beta \in M_a$ then $[\![\alpha(\beta)]\!]^{M,w,t,g} = [\![\alpha]\!]^{M,w,t,g}([\![\beta]\!]^{M,w,t,g})$.

(ILt7) If α and $\beta \in ME_a$ then $[\![\ulcorner\alpha{=}\beta\urcorner]\!]^{M,w,t,g} = 1$ iff $[\![\alpha]\!]^{M,w,t,g} = [\![\beta]\!]^{M,w,t,g}$, 0 otherwise.

(ILt8) If Φ and $\Psi \in ME_t$ then:

$[\![\ulcorner\neg\Phi\urcorner]\!]^{M,w,t,g} = 1$ iff $[\![\Phi]\!]^{M,w,t,g} = 0$.

$[\![\ulcorner(\Phi\wedge\Psi)\urcorner]\!]^{M,w,t,g} = 1$ iff both $[\![\Phi]\!]^{M,w,t,g} = 1$ and $[\![\psi]\!]^{M,w,t,g} = 1$.

$[\![\ulcorner(\Phi\vee\Psi)\urcorner]\!]^{M,w,t,g} = 1$ iff either $[\![\Phi]\!]^{M,w,t,g} = 1$ or $[\![\psi]\!]^{M,w,t,g} = 1$.

$[\![\ulcorner(\Phi\rightarrow\Psi)\urcorner]\!]^{M,w,t,g} = 1$ iff either $[\![\Phi]\!]^{M,w,t,g} = 0$ or $[\![\psi]\!]^{M,w,t,g} = 1$.

$[\![\ulcorner(\Phi\leftrightarrow\Psi)\urcorner]\!]^{M,w,t,g} = 1$ iff $[\![\Phi]\!]^{M,w,t,g} = [\![\Psi]\!]^{M,w,t,g}$.

(ILt9) If $\Phi \in ME_t$ and for some a, $\upsilon \in Var_a$ then:

$[\![\ulcorner\forall\upsilon\Phi\urcorner]\!]^{M,w,t,g} = 1$ iff $[\![\Phi]\!]^{M,w,t,g'} = 1$ for all g' that are υ-variant to g.

$[\![\ulcorner\exists\upsilon\Phi\urcorner]\!]^{M,w,t,g} = 1$ iff $[\![\Phi]\!]^{M,w,t,g'} = 1$ for some g' that is υ-variant to g.

(ILt10) If $\Phi \in ME_t$ then:

$[\![\ulcorner\Box\Phi\urcorner]\!]^{M,w,t,g} = 1$ iff

$[\![\Phi]\!]^{M,w',t',g} = 1$ for all $w' \in W$ and all $t' \in T$.

$[\![\ulcorner fut(\Phi)\urcorner]\!]^{M,w,t,g} = 1$ iff

$[\![\Phi]\!]^{M,w,t',g} = 1$ for some $t' \in T$ such that $t < t'$.

$[\![\ulcorner past(\Phi) \urcorner]\!]^{M,w,t,g} = 1$ iff

$[\![\Phi]\!]^{M,w,t',g} = 1$ for some $t' \in T$ such that $t > t'$.

(ILt11) If $\alpha \in ME_a$ then $[\![\ulcorner {}^{\wedge}\alpha \urcorner]\!]^{M,w,t,g} = \eta \in den(a,M)^{W \times T}$ such that for all $<w,t> \in W \times T$

$\eta(w,t) = [\![\alpha]\!]^{M,w,t,g}$.

(ILt12) If $\alpha \in ME_{<sa>}$ then $[\![\ulcorner {}^{\vee}\alpha \urcorner]\!]^{M,w,t,g} = [\![\alpha]\!]^{M,w,t,g}(w,t)$.

2.2. The Grammar of the Fragment

Syntactic categories (meaningful expressions) of IL are directly indexed by semantic types which serve to regulate the compositionality both of the syntax and the semantics. If a compositional grammar for a fragment of English is to be governed by similar constraints, then a correspondence between the syntactic categories of English and semantic types must first be established. Montague's original technique for guaranteeing such a correspondence was elegantly simplified by Bennett, [B8], and it is this simplification which is now presented.

2.2.1. Syntactic Categories and Semantic Types

The set *Cat* of available syntactic categories is defined by the following recursive definition:

$t \in Cat$.

$CN \in Cat$.

$IV \in Cat$.

If A and B $\in Cat$ then $A/_n B \in Cat$ $(n \geq 1)$.

These categories are mapped onto semantic types by the function f as follows:

$f(t) = t$.

$f(IV) = <et>$.

$f(CN) = <et>$.

$f(A/_n B) = <<s,f(B)>f(A)>$.

For Montague the Lexicon is a set which may be defined as:

$Lex = \cup_{A \in Cat} B_A$

where B_A is the set of *basic expressions* of type A. The set of *significant phrases of English* recognised by the PTQ fragment is:

$$\cup_{A \in Cat} P_A$$

where P_A is the set of *phrases* of type A. For all $A \in Cat$, $B_A \subseteq P_A$.

Where the symbol " ≈» " is to be read " translates into" an acceptable translation from English to IL may take the form:

$$\omega \approx\!\!\gg \omega'$$ †23

provided that $\omega \in P_A$, $f(A) = a$, and $\omega' \in ME_a$.

Since for all values of n the categories $A/_nB$ map to the same semantic type, the purpose of the subscript is plainly to discriminate between syntactically distinct categories serving a common semantic role.†24 Where convenient $A/_1B$, $A/_2B$, $A/_3B$ ..... may be written A/B, A//B, A///B ....., likewise <ab> may be written <a, b> if, as is unlikely, there is any ambiguity concerning the identity of the elements. Although an infinite set of categories is in principle available, only a small number are actualised in English. Those featuring in Montague's fragment are tabulated in fig 1.

2.2.2. Rule Forms in PTQ

Montague's notation for categories is derived from that introduced by Ajdukiewicz, [A1], in the development of categorial grammar; but as Partee has demonstrated, [P2], [P4], the grammar of PTQ is not itself a categorial grammar, although it subsumes the generalisation of one. A categorial grammar is a phrase structure grammar allowing rules only of the form:

$$A \rightarrow A/B\ B$$

ie. the *concatenation* of an expression of category A/B and another expression of category B gives a result in category A.

The grammar of PTQ may be formulated†25 in terms of 17 pairs of correlated syntactic and semantic

†23. It is important to distinguish ω' from $\lceil \omega' \rceil$. The former is an atomic metasymbol while the latter represents a primed variant of whatever the metasymbol ω denotes.

†24. In Montague's original formulation the primitive categories are e and t, with IV (intransitive verb phrase) and CN (common noun phrase) introduced as convenient mnemonics for $t/_1e$ and $t/_2e$ respectively. The semantic type of e was defined as: $f(e) = e$, thus both IV and CN mapped to <<se>t>.

†25. The format adopted for expository purposes is based on that of Janssen, [J3], rather than that appearing in the original.

****

Categories Used in the PTQ Fragment			
Category	Mnemonic	Type	Conventional Equivalent
t	t	t	Sentence
IV	IV	<et>	Intransitive verb phrase
CN	CN	<et>	Common noun phrase
t/IV	T	<<s<et>>t>	Term (noun phrase)
IV/IV	IAV	<<s<et>><et>>	Verb phrase adverb
IV/T [IV/(t/IV)]	TV	<<s<<s<et>>t>><et>>	Transitive verb
IV/t	SCVERB	<<st><et>>	Sentence complement verb
IV//IV	ICVERB	<<s<et>><et>>	Infinitival complement verb
IAV/T [(IV/IV)/T]	PREP	<<s<<s<et>>t>> <<s<et>><et>>	Preposition
t/t	SADV	<<st>t>	Sentential adverb

Fig 1

****

rules, of which the first pair <S1, T1> merely incorporates a predefined lexicon. As regards the remainder, in some cases the syntactic clauses may be seen as generalisations of the categorial apparatus in that they licence the combination of expressions of category $A/_n B$ with expressions of category B to form expressions of category A by means of *some structural operation* which may, but need not, exceed simple concatenation. These cases Montague classifies as *rules of functional application* because of the common form assumed by the semantic correlate. A rule of functional application may be exemplified by the following pattern:

(S*n*) (i) If δ is of category $A/_n B$ and β is of category B then $f_k(\delta,\beta)$ is of category A.

(ii) $f_k(\delta,\beta)$ = *definition of structural operation* f_k.

(T*n*) (i) If $\delta \twoheadrightarrow \delta'$ and $\beta \twoheadrightarrow \beta'$ then $f_k(\delta,\beta) \twoheadrightarrow g_j(\delta',\beta')$.

(ii) $g_j(\delta',\beta') = \delta'({}^\wedge\beta')$.

Provided that the type of δ' is $\langle\langle s,f(B)\rangle f(A)\rangle$ and ${}^\wedge\beta'$ has the type $\langle s,f(B)\rangle$ the translation preserves type compatibility and generates a result of type $f(A)$.[†26]

†26. Both syntactic categories and semantic types embody a cancellation operation, but curiously the directions of cancellation are reversed: $A/_n B$ together with B makes A, but <ab> together with a makes b.

Not all rules conform to this generalised categorial format. Those which do not exemplify the pattern:

(S*n*) (i) If δ is of category A and β is of category B then $f_k(\delta,\beta)$ is of category C.

(ii) $f_k(\delta,\beta)$ = *definition of structural operation* f_k.

(T*n*) (i) If $\delta \twoheadrightarrow \delta'$ and $\beta \twoheadrightarrow \beta'$ then $f_k(\delta,\beta) \twoheadrightarrow g_j(\delta',\beta')$.

(ii) $g_j(\delta',\beta')$ = *definition of semantic operation* g_j.

The semantic operation g_j may now specify any type compatible function of δ' and β' which legitimately generates a result of type *f*(C).

****

Colloquial Variables	
Colloquial Form	Pure Form
e	$v_{0,e}$
x_n	$v_{2*(n+1),e}$
X, Y, Z	$v_{j,e}$ where j>0 is the next unused odd subscript.
r, s	$v_{0,<st>}$ and $v_{1,<st>}$
p, q	$v_{0,<s<et>>}$ and $v_{1,<s<et>>}$
n	$v_{0,<s<<s<et>>t>>}$
S	$v_{0,<s<e<et>>>}$
G	$v_{0,<s<e,f(IAV)>>}$

Fig 2

****

Let *S** be the set {S*n*:S*n* is a syntactic rule}, *T** the set {T*n*:T*n* is a translation rule}, *F** the set {f_k:f_k is a structural operation} and *G** the set {g_j:g_j is a semantic operation}. There is a functional dependency of *F** upon *S**[†27] but not vice versa; likewise there is an asymmetrical functional dependency of *G** upon *T**. In UG Montague suggested that there should be a 1:1 correspondence between structural and semantic operations, but this condition plainly fails for PTQ since rules S14, S15, and S16 all employ structural

†27. A set Y is functionally dependent on a set X if there is a function from X to Y, ie. the specification of $x \in X$ serves uniquely to determine $y \in Y$.

operation f_{10} while the semantic operation for T14 differs from those of T15 and T16. There is however a 1:1 correspondence between S rules and T rules, thus G^* turns out to be functionally dependent on S^*.

2.2.3. Colloquial Variables

In order to improve readability, and because of the frequency of their use, Montague introduces a number of colloquial variables. The iconic form has been varied in this exposition, but the essential characteristics are preserved and are illustrated in fig 2.

2.2.4. The Lexicon for PTQ

For each realised $A \in Cat$, B_A is defined by extension:

B_{IV} = {change, rise, run, talk, walk}.

B_{CN} = {fish, man, park, pen, price, temperature, unicorn, woman}.

B_T = {Bill, John, Mary, ninety, he_n} ($n \geq 0$).

B_{IAV} = {allegedly, rapidly, slowly, voluntarily}.

B_{TV} = {be, conceive, date, eat, find, lose, love, seek}.

B_{SCVERB} = {believe that, assert that}.

B_{ICVERB} = {try to, wish to}.

B_{PREP} = {about, in}.

B_{SADV} = {necessarily}.

For all other $A \in Cat$, $B_A = \emptyset$.

2.2.5. The Grammar Rules

For each $A \in Cat$ the grammar rules define and interpret the members of P_A.

(S1) For all $A \in Cat$, $B_A \subseteq P_A$.

(T1) With the underlisted exceptions, if $\omega \in B_A$ then ω ⤳ $\ulcorner \omega' \urcorner$, ie. a primed variant of ω.[†28]

---

†28. The "primed variant" is a convenient colloquial form for an appropriate constant $c_{n,f(A)} \in Con_{f(A)}$. For example: John ⤳ john′. Montague introduces a translation function g from basic expressions other than "be", "necessarily" and members of B_T, such that if $\omega \in B_A$, and g is defined for ω, then $g(\omega) \in Con_{f(A)}$. Plainly ω ⤳ $g(\omega)$, but since "g" is already in use to denote sequences, I find this notation misleading.

be $\Rightarrow$ $\lambda n\lambda e\,{}^{\vee}n({}^{\wedge}\lambda Z(e=Z))$.

necessarily $\Rightarrow$ $\lambda r(\Box{}^{\vee}r)$.

he_n $\Rightarrow$ $\lambda p\,{}^{\vee}p(x_n)$.

For all other $\alpha \in B_T$, α $\Rightarrow$ $\lambda p\,{}^{\vee}p(\ulcorner\alpha\urcorner)$.

(S2) If $\zeta \in P_{CN}$ then $f_0(\zeta)$, $f_1(\zeta)$, and $f_2(\zeta) \in P_T$.

$f_0(\zeta) = \ulcorner$every $\zeta\urcorner$, $f_1(\zeta) = \ulcorner$the $\zeta\urcorner$, $f_2(\zeta) = \ulcorner$a $\zeta\urcorner$ or $\ulcorner$an $\zeta\urcorner$ as appropriate.†29

(T2) if ζ $\Rightarrow$ ζ' then:

$\ulcorner$every $\zeta\urcorner$ $\Rightarrow$ $\lambda q\forall X(\zeta'(X)\rightarrow{}^{\vee}q(X))$.

$\ulcorner$the $\zeta\urcorner$ $\Rightarrow$ $\lambda q\exists Y(\forall X(\zeta'(X)\leftrightarrow X=Y)\wedge{}^{\vee}q(Y))$.

$\ulcorner$a(n) $\zeta\urcorner$ $\Rightarrow$ $\lambda q\exists X(\zeta'(X)\wedge{}^{\vee}q(X))$.

(S3) If $\zeta \in P_{CN}$ and $\phi \in P_t$ then $f_{3,n}(\zeta,\phi) \in P_{CN}$.

$f_{3,n}(\zeta,\phi) = \ulcorner\zeta$ such that $\psi\urcorner$, where

ψ comes from ϕ by replacing each occurrence of he_n and him_n by a surface pronoun of like case and having the gender of the first member of $B_{CN} \in \zeta$.

(T3) If ζ $\Rightarrow$ ζ' and ϕ $\Rightarrow$ ϕ' then $f_{3,n}(\zeta,\phi)$ $\Rightarrow$ $\lambda x_n(\zeta'(x_n)\wedge\phi')$.

(S4) If $\alpha \in P_T$ and $\delta \in P_{IV}$ then $f_4(\alpha,\delta) \in P_t$.

$f_4(\alpha,\delta) = \alpha\ \gamma$, where γ comes from δ by replacing the *first verb* in δ by the 3rd. person singular present.

(T4) If α $\Rightarrow$ α' and δ $\Rightarrow$ δ' then $f_4(\alpha,\delta)$ $\Rightarrow$ $\alpha'({}^{\wedge}\delta')$.

(S5) If $\delta \in P_{TV}$ and $\beta \in P_T$ then $f_5(\delta,\beta) \in P_{IV}$.

$f_5(\delta,\beta) = \delta\ \beta$ if β does not have the form he_n, $\ulcorner\delta\ him_n\urcorner$ otherwise.

(T5) If δ $\Rightarrow$ δ' and β $\Rightarrow$ β' then $f_5(\delta,\beta)$ $\Rightarrow$ $\delta'({}^{\wedge}\beta')$.

(S6) If $\delta \in P_{PREP}$ and $\beta \in P_T$ then $f_5(\delta,\beta) \in P_{IAV}$.

(T6) If δ $\Rightarrow$ δ' and β $\Rightarrow$ β' then $f_5(\delta,\beta)$ $\Rightarrow$ $\delta'({}^{\wedge}\beta')$.

---

†29. There is no obvious reason for this syncategorematic introduction of determiners in preference to their inclusion in a set $B_{T/CN}$, but the outcome is the same either way. Montague's attempt to handle euphonics by introducing the *nu ephelkustikon* syntactically rather than within phonology is rather unfortunate.

(S7) If $\delta \in P_{SCVERB}$ and $\phi \in P_t$ then $f_6(\delta,\phi) \in P_{IV}$.

$f_6(\delta,\phi) = \delta\,\phi$.

(T7) If $\delta \twoheadrightarrow \delta'$ and $\phi \twoheadrightarrow \phi'$ then $f_6(\delta,\phi) \twoheadrightarrow \delta'(\hat{}\phi')$.

(S8) If $\delta \in P_{ICVERB}$ and $\gamma \in P_{IV}$ then $f_6(\delta,\gamma) \in P_{IV}$.

(T8) If $\delta \twoheadrightarrow \delta'$ and $\gamma \twoheadrightarrow \gamma'$ then $f_6(\delta,\gamma) \twoheadrightarrow \delta'(\hat{}\gamma')$.

(S9) If $\delta \in P_{SADV}$ and $\phi \in P_t$ then $f_6(\delta,\phi) \in P_t$.

(T9) If $\delta \twoheadrightarrow \delta'$ and $\phi \twoheadrightarrow \phi'$ then $f_6(\delta,\phi) \twoheadrightarrow \delta'(\hat{}\phi')$.

(S10) If $\delta \in P_{IAV}$ and $\gamma \in P_{IV}$ then $f_7(\delta,\gamma) \in P_{IV}$.

$f_7(\delta,\gamma) = \gamma\,\delta$.

(T10) If $\delta \twoheadrightarrow \delta'$ and $\gamma \twoheadrightarrow \gamma'$ then $f_7(\delta,\gamma) \twoheadrightarrow \delta'(\hat{}\gamma')$.

(S11) If ϕ and $\psi \in P_t$ then $f_8(\phi,\psi)$ and $f_9(\phi,\psi) \in P_t$.

$f_8(\phi,\psi) = \ulcorner \phi$ and $\psi \urcorner$.

$f_9(\phi,\psi) = \ulcorner \phi$ or $\psi \urcorner$.

(T11) If $\phi \twoheadrightarrow \phi'$ and $\psi \twoheadrightarrow \psi'$ then:

$$f_8(\phi,\psi) \twoheadrightarrow \ulcorner(\phi' \wedge \psi')\urcorner.$$

$$f_9(\phi,\psi) \twoheadrightarrow \ulcorner(\phi' \vee \psi')\urcorner.$$

(S12) If γ and $\delta \in P_{IV}$ then $f_8(\gamma,\delta)$ and $f_9(\gamma,\delta) \in P_{IV}$.

(T12) If $\gamma \twoheadrightarrow \gamma'$ and $\delta \twoheadrightarrow \delta'$ then:

$$f_8(\gamma,\delta) \twoheadrightarrow \lambda X(\gamma'(X) \wedge \delta'(X)).$$

$$f_9(\gamma,\delta) \twoheadrightarrow \lambda X(\gamma'(X) \vee \delta'(X)).$$

(S13) If α and $\beta \in P_T$ then $f_9(\alpha,\beta) \in P_T$.

(T13) If $\alpha \twoheadrightarrow \alpha'$ and $\beta \twoheadrightarrow \beta'$ then $f_9(\alpha,\beta) \twoheadrightarrow \lambda q(\alpha'(q) \vee \beta'(q))$.

(S14) If $\alpha \in P_T$ and $\phi \in P_t$ then $f_{10,n}(\alpha,\phi) \in P_t$.

$f_{10,n}(\alpha,\phi) = \psi$, where either:

(i) $\alpha = he_k$ and ψ comes from ϕ by replacing he_n with he_k and him_n with him_k.

(ii) $\alpha \neq he_k$ and ψ comes from ϕ by replacing the first occurrence of he_n or him_n by α and all other occurrences by a surface pronoun of corresponding case and matching in gender the first member of B_{CN} or B_T in α.

(T14) If $\alpha \Rightarrow \alpha'$ and $\phi \Rightarrow \phi'$ then $f_{10,n}(\alpha,\phi) \Rightarrow \alpha'(\hat{}\lambda x_n \phi')$.

(S15) If $\alpha \in P_T$ and $\zeta \in P_{CN}$ then $f_{10,n}(\alpha,\zeta) \in P_{CN}$.

(T15) If $\alpha \Rightarrow \alpha'$ and $\zeta \Rightarrow \zeta'$ then $f_{10,n}(\alpha,\zeta) \Rightarrow \lambda Y \alpha'(\hat{}\lambda x_n(\zeta'(Y)))$.

(S16) If $\alpha \in P_T$ and $\delta \in P_{IV}$ then $f_{10,n}(\alpha,\delta) \in P_{IV}$.

(T16) If $\alpha \Rightarrow \alpha'$ and $\delta \Rightarrow \delta'$ then $f_{10,n}(\alpha,\delta) \Rightarrow \lambda Y \alpha'(\hat{}\lambda x_n(\delta'(Y)))$.

(S17) If $\alpha \in P_T$ and $\delta \in P_{IV}$ then $f_{11}(\alpha,\delta)$, $f_{12}(\alpha,\delta)$, $f_{13}(\alpha,\delta)$, $f_{14}(\alpha,\delta)$ and $f_{15}(\alpha,\delta) \in P_t$.

$f_{11}(\alpha,\delta) = \alpha\ \gamma$ where γ comes from δ by replacing the first verb in δ by its negative third person singular present.

$f_{12}(\alpha,\delta) = \alpha\ \gamma$ where γ comes from δ by replacing the first verb in δ by its third person singular future.

$f_{13}(\alpha,\delta) = \alpha\ \gamma$ where γ comes from δ by replacing the first verb in δ by its negative third person singular future.

$f_{14}(\alpha,\delta) = \alpha\ \gamma$ where γ comes from δ by replacing the first verb in δ by its third person singular present perfect.

$f_{15}(\alpha,\delta) = \alpha\ \gamma$ where γ comes from δ by replacing the first verb in δ by its negative third person singular present perfect.

(T17) If $\alpha \Rightarrow \alpha'$ and $\delta \Rightarrow \delta'$ then:

$f_{11}(\alpha,\delta) \Rightarrow \neg\alpha'(\hat{}\delta')$.

$f_{12}(\alpha,\delta) \Rightarrow \mathit{fut}(\alpha'(\hat{}\delta'))$.

$f_{13}(\alpha,\delta) \Rightarrow \neg\mathit{fut}(\alpha'(\hat{}\delta'))$.

$f_{14}(\alpha,\delta) \Rightarrow \mathit{past}(\alpha'(\hat{}\delta'))$.

$f_{15}(\alpha,\delta) \Rightarrow \neg\mathit{past}(\alpha'(\hat{}\delta'))$.

Syntactic variables of the form "he_n", together with case marked variations which, curiously, are never specifically defined in the lexicon, are introduced into the grammar in order to deal with scope phenomena, coreferentiality and anaphora. Although not explicitly mentioned in PTQ, there appears to be a convention that the set of English sentences recognised in the fragment is:

$Esen = \{s : s \in P_t$ and s contains no syntactic variables$\}$.

ie. there must be a "surface filtering rule", [R4], [P4], [J3], [J4], which rejects sentences containing syntactic variables other than as intermediate stages in a derivation.†30

2.2.6. Meaning Postulates

On several occasions the powers introduced to handle *intensional* phenomena turn out to be excessive for the majority of cases. For example most transitive verbs do not admit a distinction between *de re* and *de dicto* readings although such a distinction is latent in the translation prescribed by T5. In order to limit the powers of the grammar in this and other appropriate situations Montague introduces *meaning postulates.* A total of nine postulates appear in PTQ, of which five become redundant given Bennett's simplified semantic typing. The remaining four, numbered as in the original, are as follows:

(MP1)Rigid designator postulate:

$\exists X \Box (X = \alpha)$ where $\alpha \in$ {bill′, john′, mary′, ninety′}.

(MP4)Extensional transitive verb postulate:

$\forall X \forall n \Box[\delta(X,n) \leftrightarrow {}^{\vee}n({}^{\wedge}\lambda Y(\delta_*(X,Y)))]$

where $\alpha \Rightarrow \delta$, $\alpha \in$ {date, eat, find, lose, love} and

$\delta_* =_{df} \lambda Y \lambda X[\delta({}^{\wedge}\lambda p[{}^{\vee}p(Y)])(X)]$.

(M8) Extensional preposition postulate:

$\forall n \forall p \forall e \Box \delta(e,p,n) \leftrightarrow {}^{\vee}n({}^{\wedge}\lambda Y(\delta_*(e,p,Y)))$

where $\alpha \Rightarrow \delta$, $\alpha \in$ {in}, and

$\delta_* =_{df} \lambda Y \lambda p \lambda e[\delta(e,p,{}^{\wedge}\lambda q\, {}^{\vee}q(Y))]$.†31

---

†30. Partee, [P4], makes the interesting suggestion that alternatively remaining syntactic variables could be converted to *indexical* surface pronouns; but as Janssen observes, [J3], without amendment to the valuation mechanism this ploy would invest the variable assignment function *g*, which is index independent, with the inappropriate onus of handling context sensitivity.

†31. For some reason Montague never reduces prepositional constructions to relational form, ie. he accepts $\delta(n)(p)(e)$ rather than $\delta(e,p,n)$. I can see no advantage in making reduction to relational notation an arbitrary option. The introduction of a constant δ_* in place of existentially quantified variables S and G in MP4 and MP8 respectively is due to Dowty [D9].

(MP9)Synonymy postulate:

$$\forall X \forall n \Box[\text{seek}'(X,n) \leftrightarrow \text{try}'(X,{}^\wedge[\text{find}'(n)])].$$

2.3. Analysis Trees

The derivational history of a sentence generated by Montague's fragment may be represented diagrammatically by an *analysis tree*. Each leaf position on the tree is occupied by a member of some B_A, while non terminal nodes are labelled by an ordered pair **<*l*,*m*>**, where ***m*** identifies the structural operation licenced by a rule for combining the daughters of the node, and ***l*** is the output of that operation. Normally ***m*** will be an integer but as Janssen points out, [J3], rules S3, S14, S15 and S16 are actually "hyperrules" introducing distinct operations for each instantiation of the syntactic variable subscript *k* involved: hence in these cases identification requires that ***m*** take the form <*f*,*k*> where *f* is the structural operation number, and *k* is the subscript. Any subtree having its root labelled by a pair <*l*,*m*> such that the output of $m \in P_t$ and $l \in$ *Esen* as defined above is the analysis tree of an English sentence.

Given the functional dependencies identified earlier, we may observe at once that the node labelling convention is singularly unhelpful, for ***m*** does not serve uniquely to individuate a specific rule S*n*, whereas had a node label included the *n* from S*n* then the structural operation information would have been deducible. As a compromise I propose provisionally to label non terminal nodes with the pair <*l*, <*n*:*m*>>.

****

```
rpo(tree);
    begin
        if tree ≠ nil then
            begin
                rpo(rightsubtree);
                rpo(leftsubtree);
                translate_reduce(node)
            end
    end;
```

Fig 3

****

By traversing the analysis tree in reverse post order according to the recursive algorithm of fig 3, where "translate_reduce" applies T*n* to a node created by S*n*, it is possible to derive a topographically identical *translation tree* having nodes labelled with the IL translations of corresponding proper expressions. Consequently for illustrative purposes analysis and translation trees may be superimposed.

2.3.1. Semantically Vacuous Discriminations

****

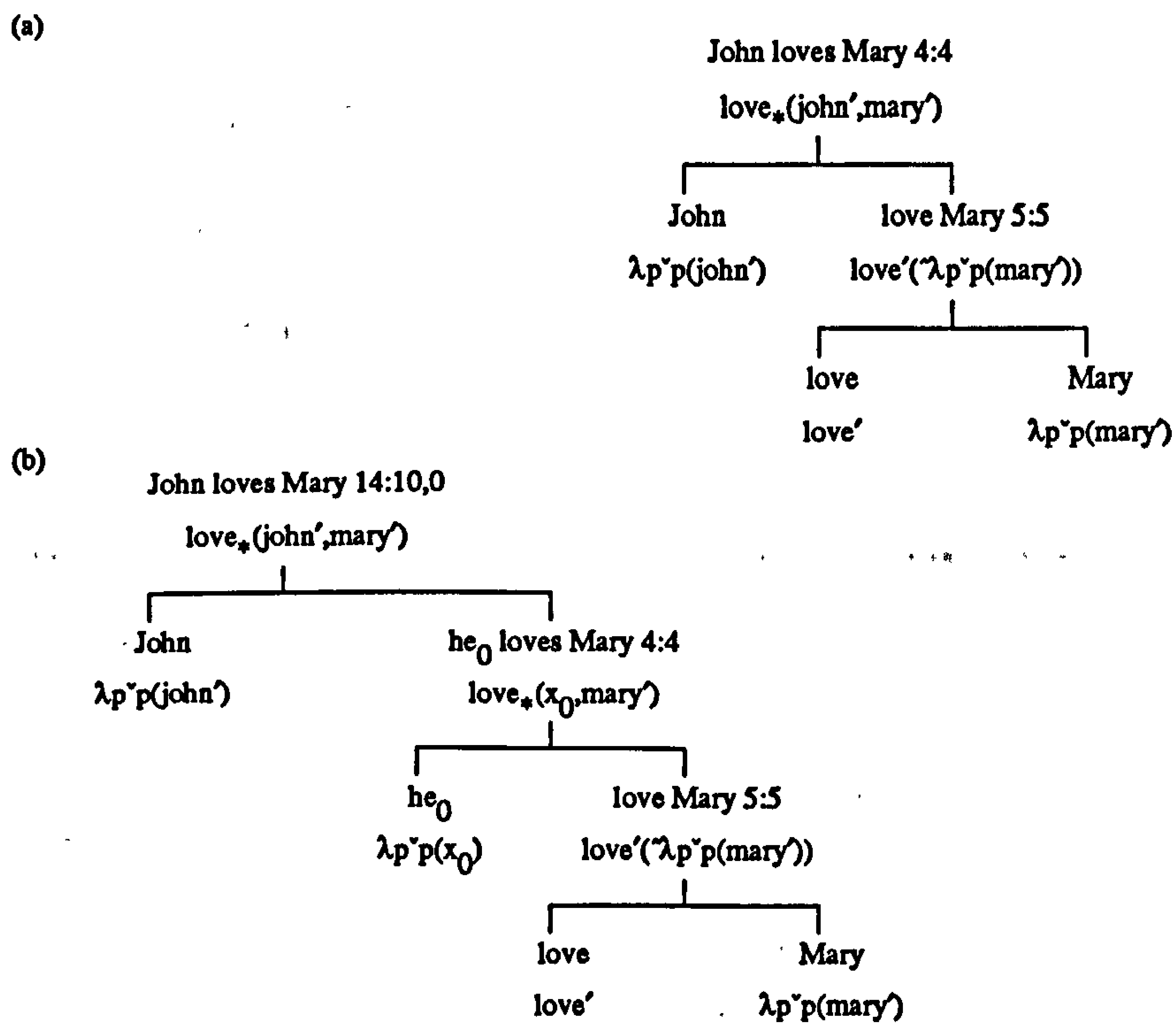


Fig 4

****

Different combinations of rule application will generate different analysis trees for the same sentence; but such distinctions in syntactic structure will entail a semantic distinction only in case the top nodes of the corresponding translation trees differ. It is an idiosyncracy of Montague grammar that this condition is not always fulfilled: ie. there may be semantically vacuous syntactic discriminations. For example noun phrases (*terms*) may be introduced *directly* into sentences by either the subject predicate rule S4 or by the sentence quantification rule S14. When the subject term of a *simple* sentence is introduced by these two rules alternatively, as in fig 4, a semantically vacuous distinction arises. The application of the pair <S4,T4> in both fig 4 (a) and fig 4 (b) triggers MP4, and accordingly it may be helpful on this first encounter to examine its effect. At the top node of the translation tree in fig 4 (a) the derivation proceeds as follows:

Construction by T4: $\lambda p\,{}^{\vee}p(john')[{}^{\wedge}love'({}^{\wedge}\lambda p\,{}^{\vee}p(mary'))]$

λ conversion: $love'({}^{\wedge}\lambda p\,{}^{\vee}p(mary'))(john')$

Relational notation: $love'(john', {}^{\wedge}\lambda p\,{}^{\vee}p(mary'))$

MP4: $\lambda p\,{}^{\vee}p(mary')[{}^{\wedge}\lambda Y(love_{*}(john', Y))]$

λ conversion: $\lambda Y(love_{*}(john', Y))[mary']$

λ conversion: $love_{*}(john', mary')$

2.3.2. Relative Clauses

****

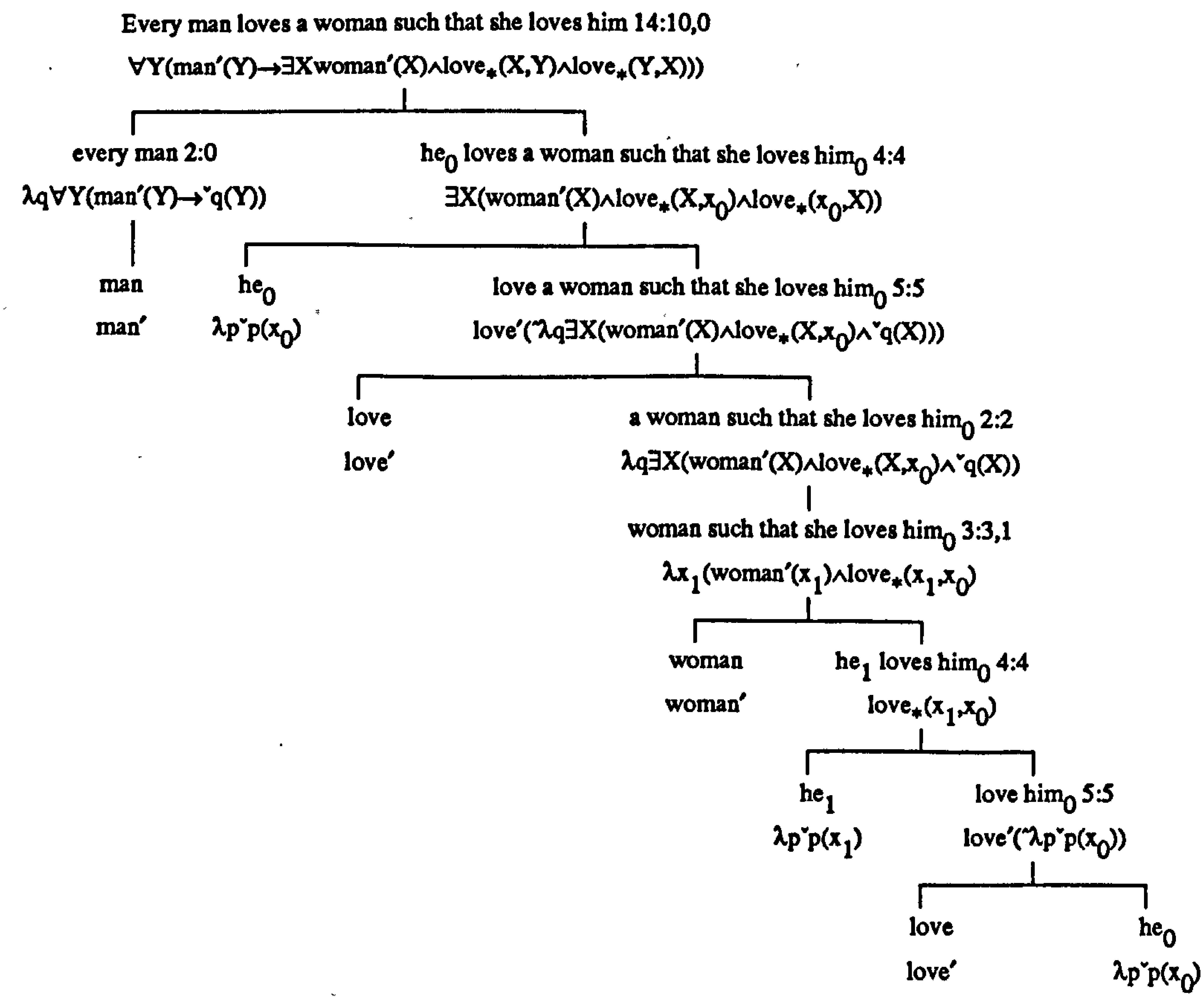


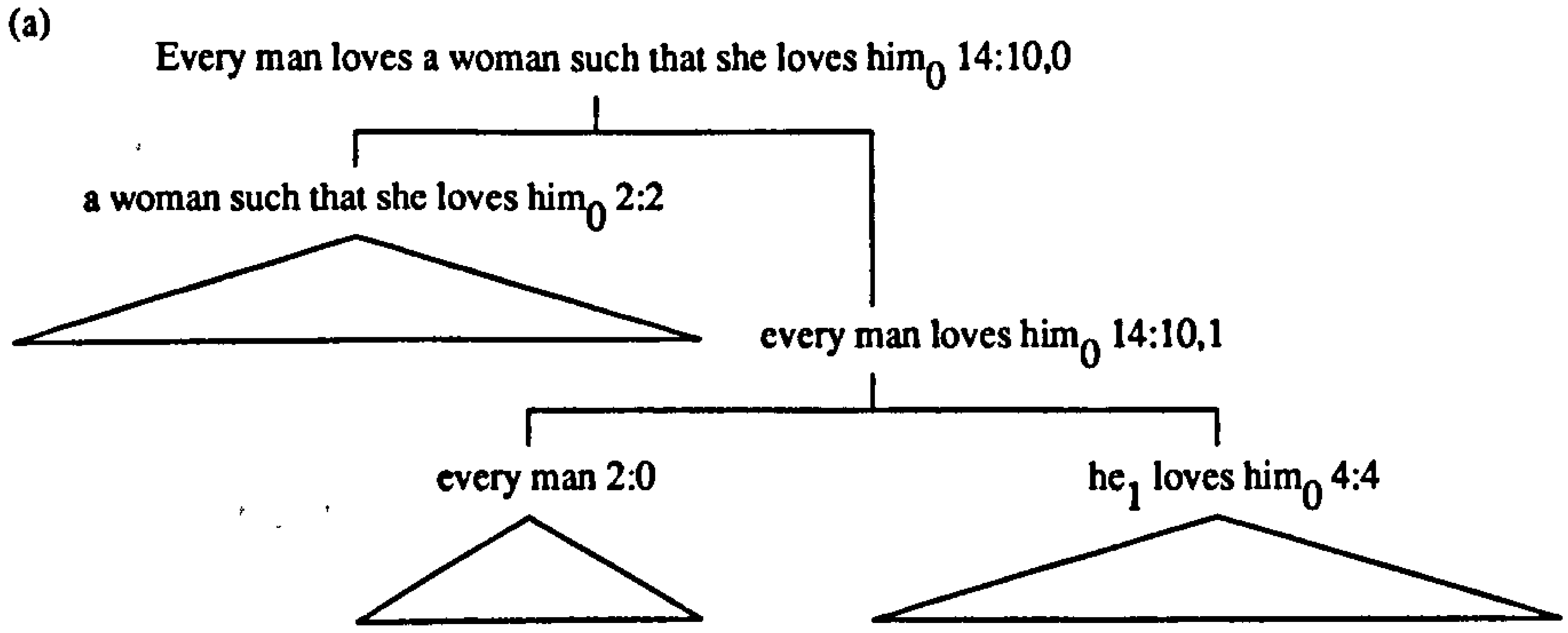
Fig 5

****

The first example of an analysis tree introduced by Montague himself in PTQ involves one of the less felicitous features of the fragment, viz. the rather contrived relative clause rule which becomes an

****

(a)



(b)

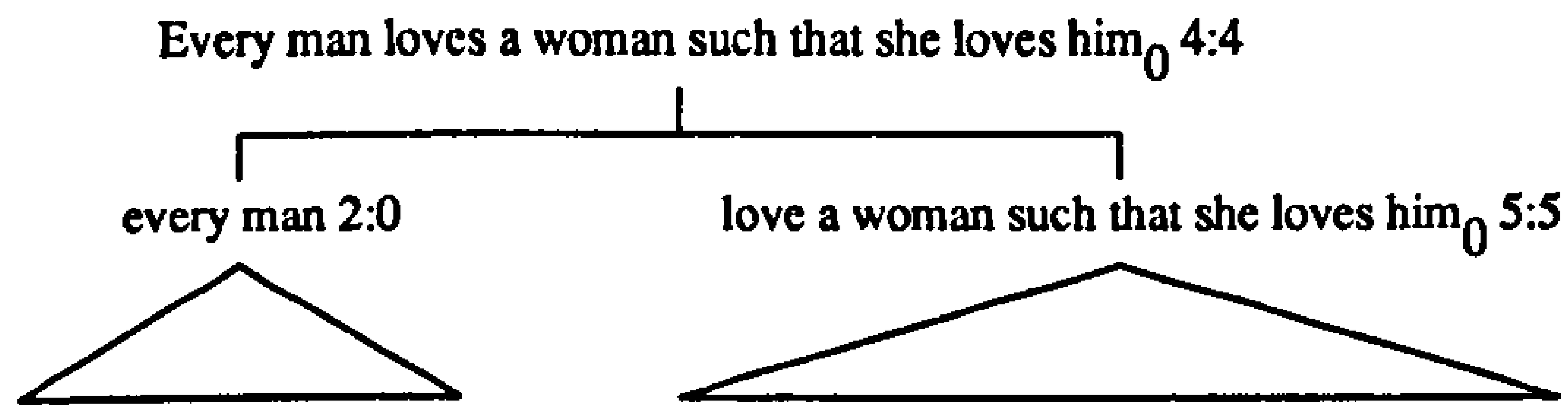


Fig 6

****

early target for revision. Relative clauses take the form illustrated by the sentence:

(44) Every man loves a woman such that she loves him.

the tree for which is illustrated in fig 5. This Montague uses to demonstrate rather cryptically the point already made that multiple trees may have a common translation. There are indeed infinite "alphabetical variants" of the tree in fig 5, ie. trees which differ trivially in the uniform substitution of alternative free syntactic variables. What is more interesting, although Montague draws no attention to it, is that alternative trees for the target sentence (44) cannot be constructed as previously by varying the mode of introduction of the noun phrases. If we attempt to introduce the subject by S4 we achieve the result in fig 6 (b), while if we endeavour to give the object noun phrase wide scope by quantifying it in at the latest opportunity using S14 we generate the tree in fig 6 (a). In neither case is the top node labelled with a member of *Esen*, since in both cases the syntactic variable him$_0$ is still present.

This illustrates a common phenomenon affecting grammars which use substitution rules to bind variables. If a pronoun in the main clause is coreferential with a noun phrase ζ within a relative clause, or if a pronoun within the relative clause is coreferential with a noun phrase ζ in the main clause, then in an analysis tree ζ must have wider scope than the noun phrase qualified by the relative clause, and its original position must be marked with the same syntactic variable as replaces the coreferential pronoun.[†32]

†32. Problematic cases arise where ζ should intuitively have narrow scope. Where ζ is in the main clause the problem is

No serious problems arise with (44) since intuitively there is no reading of the sentence in which "every man" has narrow scope, but the situation is rather different with Partee's example, [P4]:

(45) Every man such that he loves a woman loses her.

Here the preferred reading would give narrow scope to "a woman", but that is precisely the reading which PTQ fails to generate. Whereas there is a tree commencing as in fig 7 (a), any alternative beginning as in fig 7 (b) fails to represent an English sentence.

****

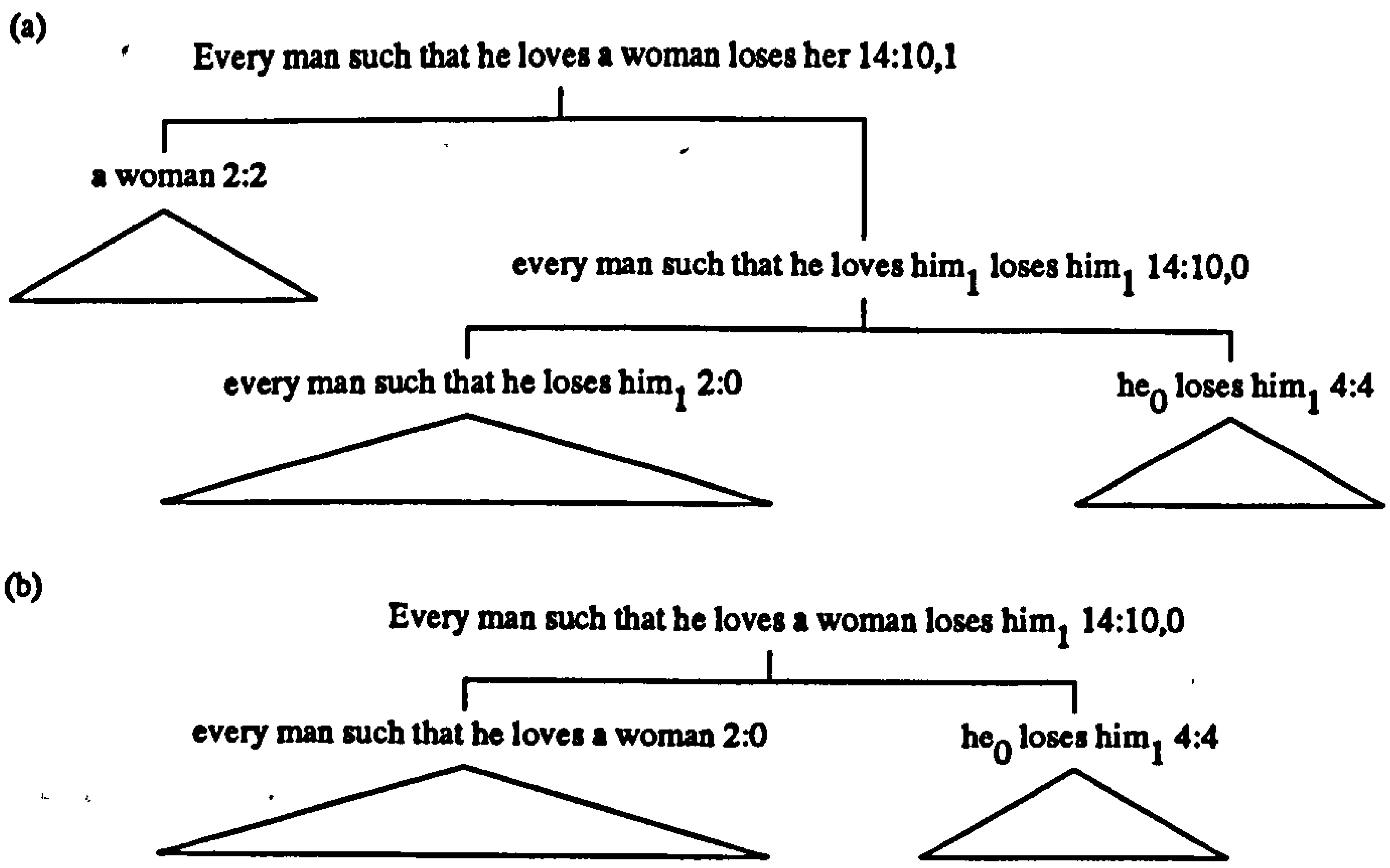


Fig 7

****

In order to protect Montague grammar from a charge of inadequacy Partee suggests that the grammar is satisfactory for genuine anaphoric references, but that the pronoun in (45) is in fact a "pronoun of laziness" as described by Geach, [G6]; but the evidence is tenuous. A pronoun of laziness behaves like a "macro" to copy across a text string with indifference to its previous referential use, thus saving repetition.†33 The pronouns in (45) on the other hand seem to perform a genuinely coreferential function. There

typified by the "Bach-Peters" sentence:

(i) The man who deserves it gets the price he wants.

The alternative with ζ in the relative clause is classically exemplified by the "donkey sentence":

(ii) Every man who owns a donkey beats it.

Partee's example (45) is a donkey sentence formulated in the vocabulary of PTQ.

†33. Nowhere is this better illustrated than in Karttunen's example, to which Partee alludes:
The man who gave his pay check to his wife was wiser than the man who gave *it* to his mistress.
where "it" is to be replaced by the *text string* "his paycheck".

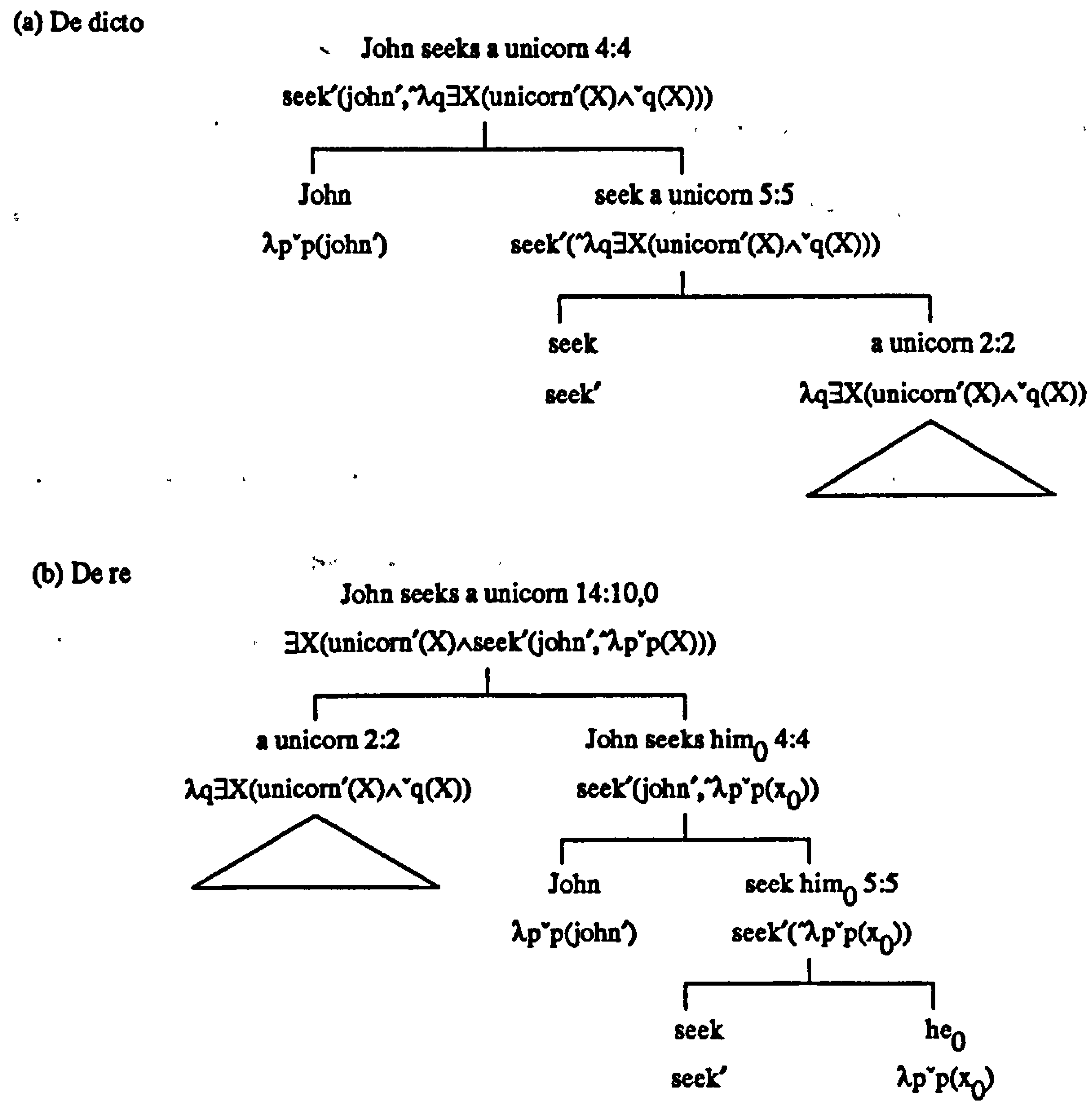


Fig 8

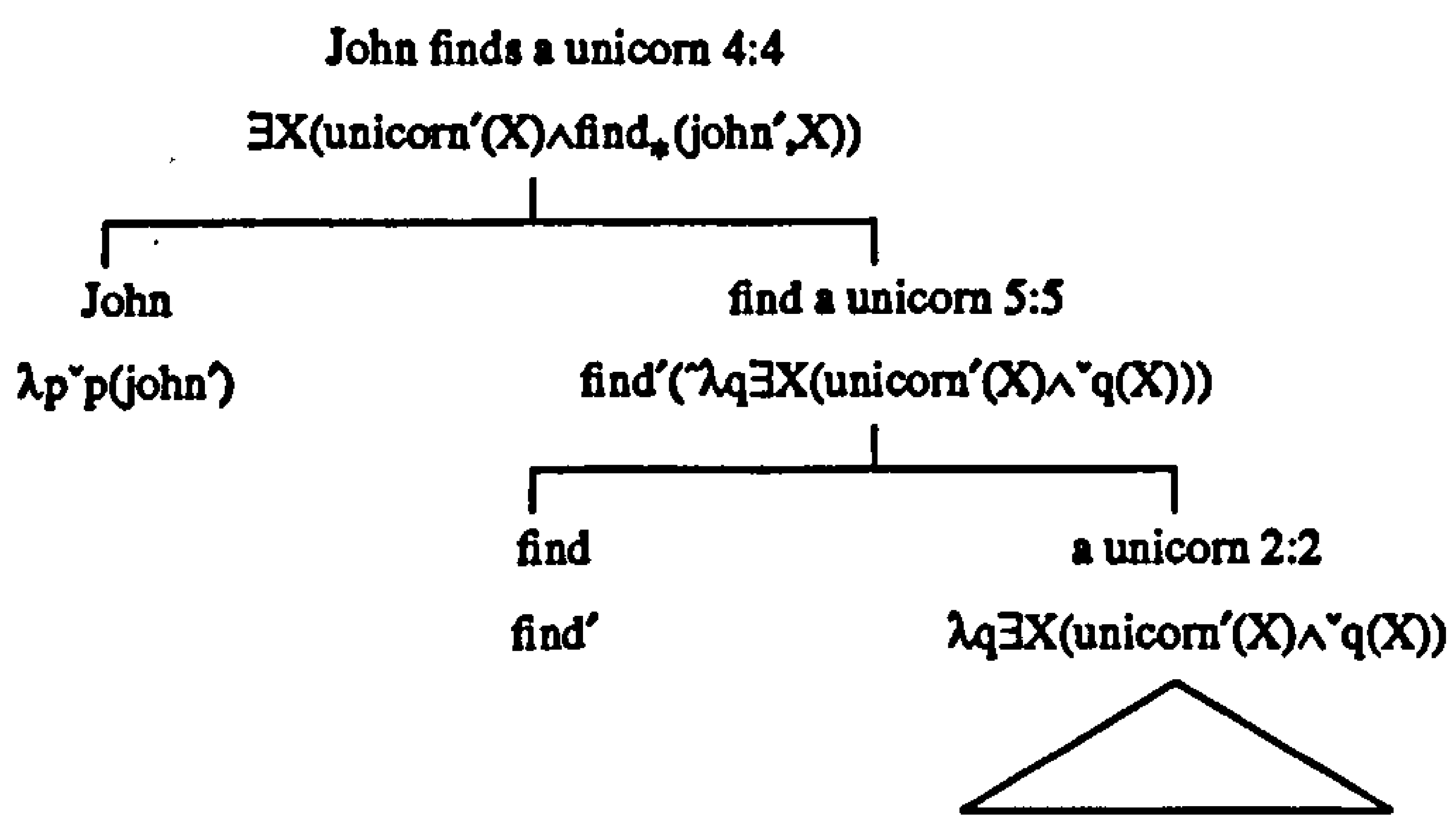


Fig 9

are accordingly certain relative clause constructions for which at present Montague grammar can offer no satisfactory analysis; we may however concede that the PTQ fragment is at least no less satisfactory than any other logic oriented grammar in finding "donkey sentences" intractable.

2.3.3. Intensional and Extensional Verbs

When considering the uniform analysis of transitive verbs discussed in the previous chapter we remarked not only that would it be necessary to provide alternative *de dicto* and *de re* readings for *intensional* verbs, but also that *de dicto* readings for *extensional* verbs should be eliminated. The distinction between the *de dicto* and *de re* readings of sentences containing intensional verbs like:

(46) John seeks a unicorn.

depends upon the alternative utilisation of S5 or S15 for the introduction of the object noun phrase. This is illustrated in fig 8, where the similarity between the IL translation of fig 8 (a) and the fully reduced translation of (36) should be noted. As promised the translation of fig 8 (b) realises the alternative IL formulation anticipated in footnote 19.

The extensional transitive verb postulate is designed to guarantee that only a *de re* reading be possible for the sentence:

(47) John finds a unicorn.

irrespective of the mode of introduction of the object term, and this turns out to be the case; for although the tree in fig 9 is topographically similar to that of fig 8 (a), the application of the pair <S4,T4> at the top node automatically triggers MP4 and "extensionalises" the verb.

****

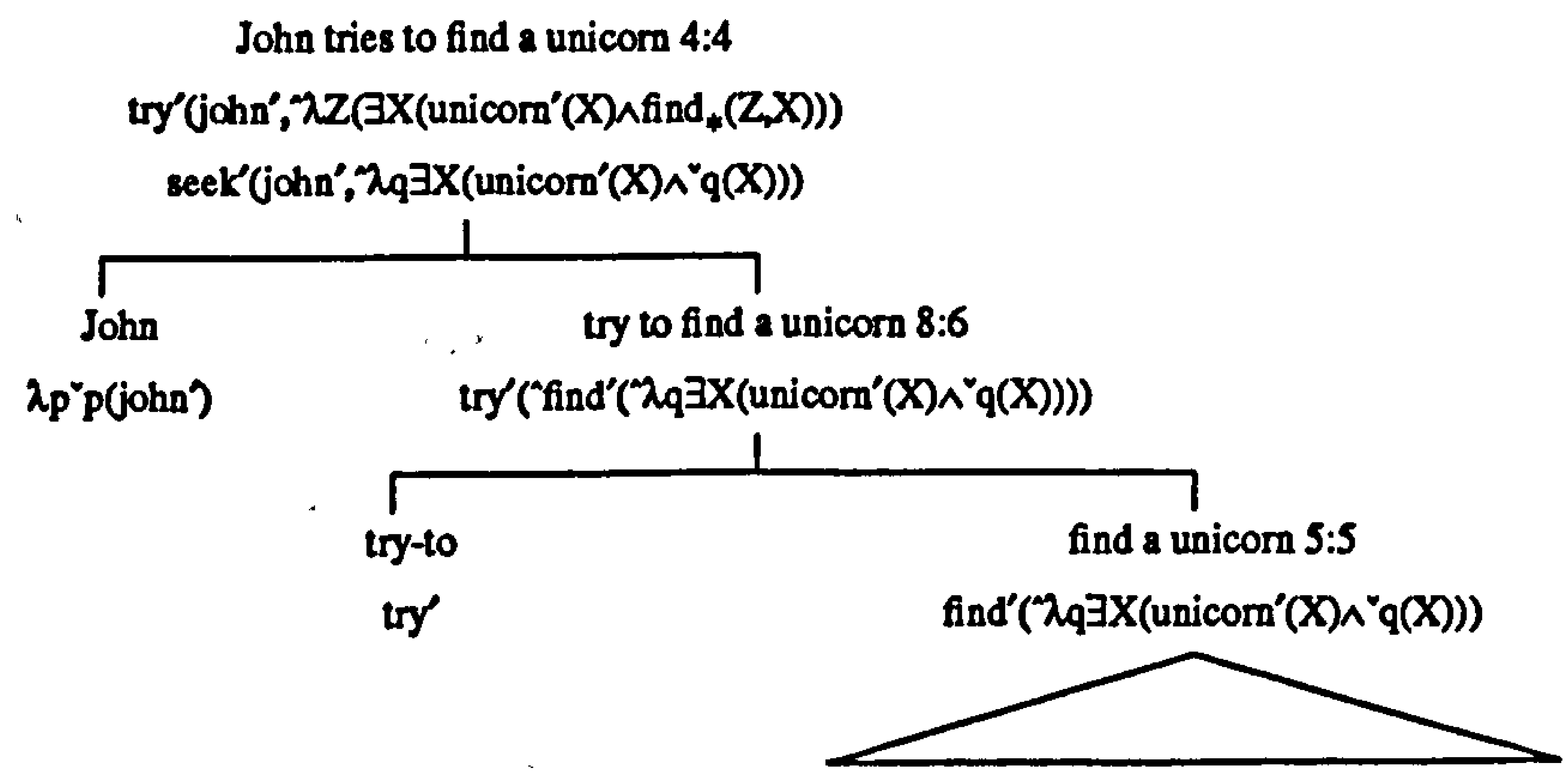


Fig 10

****

Since the order of application of reduction steps and meaning postulates is not mandatory, an interesting situation arises in the analysis of:

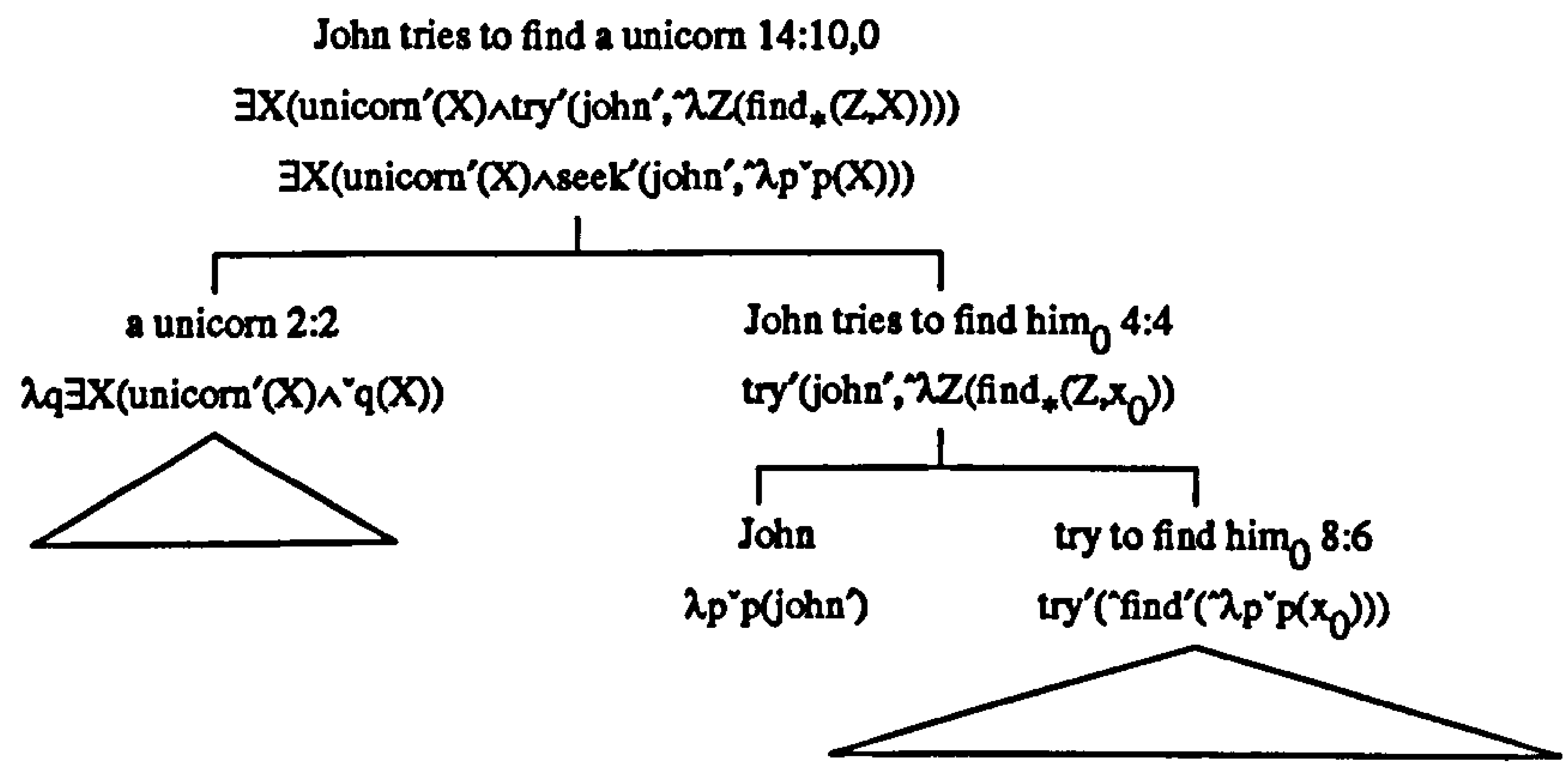


Fig 11

(48) John tries to find a unicorn.

where alternative but equivalent IL translations are available for the same tree depending on whether or not advantage is taken of MP9. The translation and reductions possible at the top node in fig 10 are as follows:

Construction by T4: $\lambda p\,{}^\vee p(john')[{}^\wedge try'({}^\wedge find'({}^\wedge\lambda q\exists X(unicorn'(X)\wedge{}^\vee q(X))))]$

λ conversion: $try'({}^\wedge find'({}^\wedge\lambda q\exists X(unicorn'(X)\wedge{}^\vee q(X))))(john')$

Relational notation: $try'(john',{}^\wedge find'({}^\wedge\lambda q\exists X(unicorn'(X)\wedge{}^\vee q(X))))$

then either:-

MP9: $seek'(john',{}^\wedge\lambda q\exists X(unicorn'(X)\wedge{}^\vee q(X)))$

or:-

λ expansion: $try'(john',{}^\wedge\lambda Z(find'(Z,{}^\wedge\lambda q\exists X(unicorn'(X)\wedge{}^\vee q(X)))))$

MP4: $try'(john',{}^\wedge\lambda Z(\lambda q\exists X(unicorn'(X)\wedge{}^\vee q(X))[{}^\wedge\lambda Y find_*(Z,Y)]))$

λ conversion: $try'(john',{}^\wedge\lambda Z(\exists X(unicorn'(X)\wedge\lambda Y find_*(Z,Y)[X])))$

λ conversion: $try'(john',{}^\wedge\lambda Z(\exists X(unicorn'(X)\wedge find_*(Z,X))))$

If “try to find” and “seek” are to count as equivalent then the sentence (48) should have an alternative *de re* reading, and this is indeed the case. Just as the translations in fig 8 (a) and fig 10 are equivalent, so too are those of fig 8 (b) and fig 11, where again the appearance of alternatives is dependent on the use or otherwise of MP9.

****

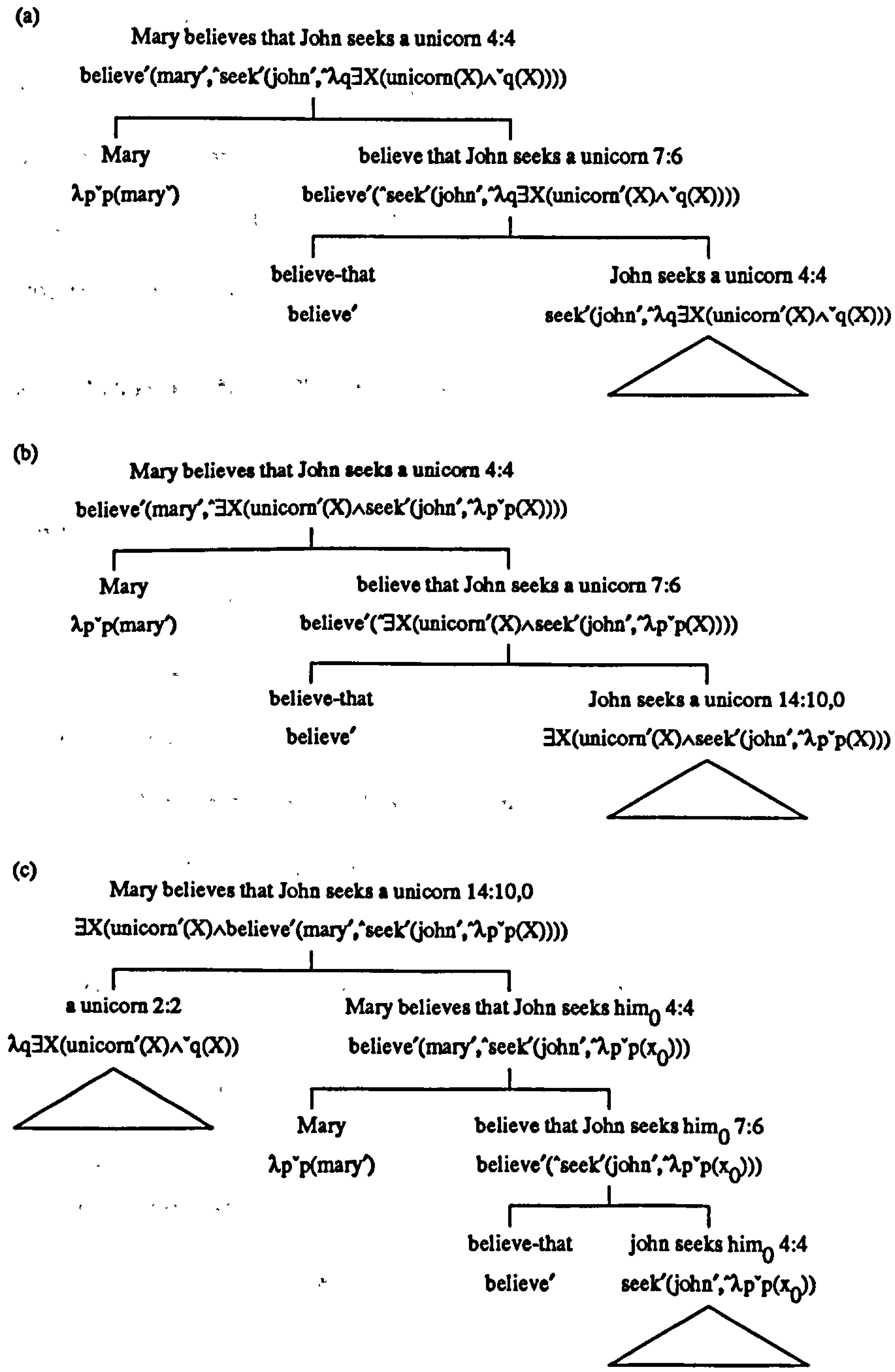


Fig 12

****

Some alternative effects of subsuming a sentence containing an intensional verb under a verb of propositional attitude as in:

(49) Mary believes that John seeks a unicorn.

may be demonstrated by the selection of possible analysis trees represented in fig 12, which illustrate alternative ontological commitments. The translations in fig 12 (a) and fig 12 (b) incorporate as complements the *de re* and *de dicto* analyses of fig 8, and these correspond to readings in which first John alone (in Mary's opinion), and then both Mary and John are zoologically naive. There is a third reading for sentence (49) wherein the *utterer* too is committed to the existence of mythical beasts and this is identified by the translation of fig 12 (c). Variations in the scope of the existential quantifier depend not only on whether but also upon when recourse is made to S14.

2.3.4. Verb Phrase and Common Noun Phrase Quantification

Certain nuances involving scope phenomena resist an analysis based entirely upon sentence quantification. For example the intuitive reading of:

(50) John wishes to find a unicorn and eat it.

is that John wishes to find some unicorn or other, and to eat which ever unicorn he happens to find. Fulfilment of the condition that the creatures found and eaten be one and the same is guaranteed if the object position of both embedded verbs is marked by a common syntactic variable. If however this variable is then bound by the sentence quantification rule S14 we derive the tree in fig 13, which not only enforces an unintended *de re* reading, but also commits the utterer to the existence of unicorns.

****

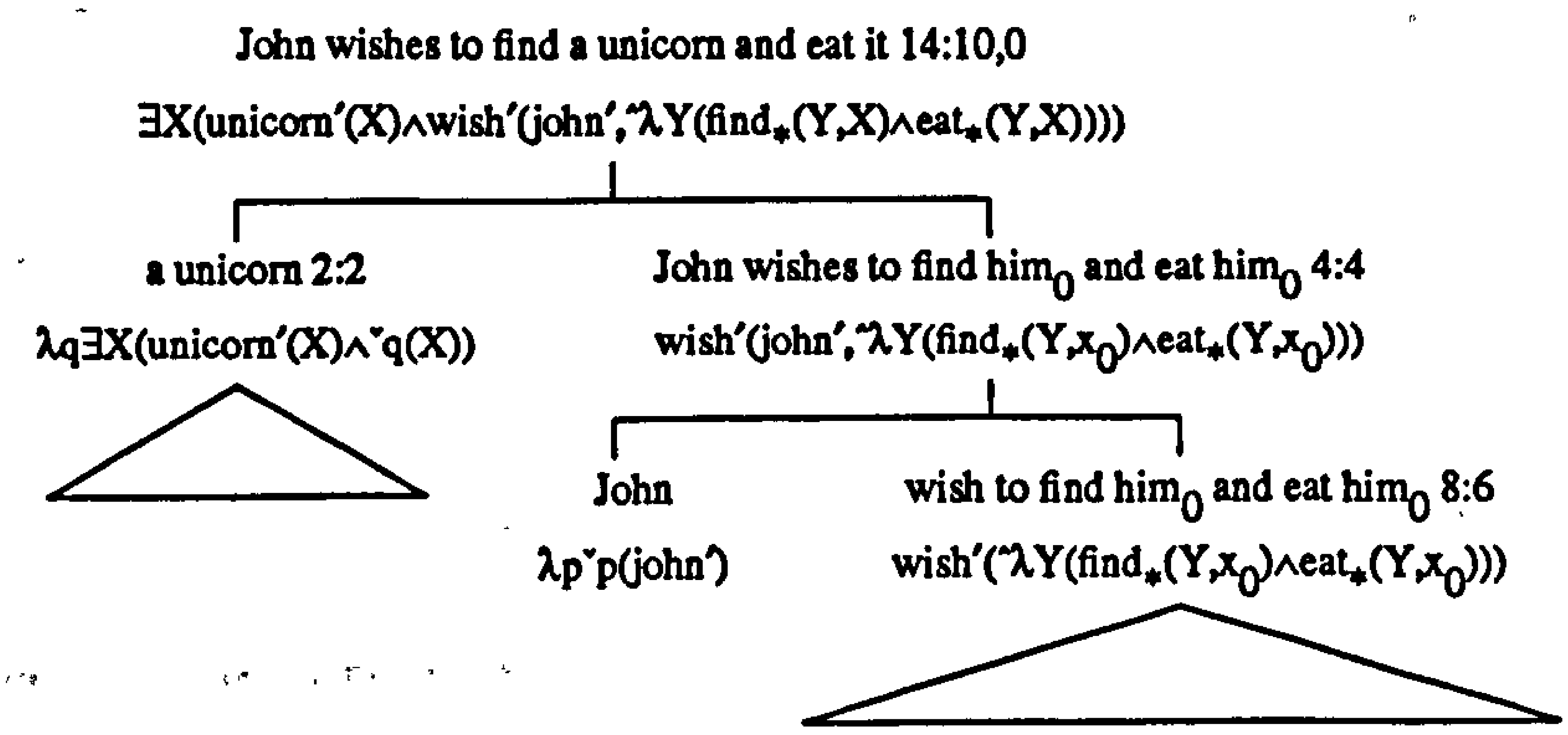


Fig 13

****

To obviate this difficulty Montague allows quantification into *verb phrases* by S16 as illustrated in fig 14, where the translation at the top node indicates a *de dicto* interpretation as required. Notice that this example serves also to justify the need for rule S12 which permits verb phrase conjunction (and disjunc-

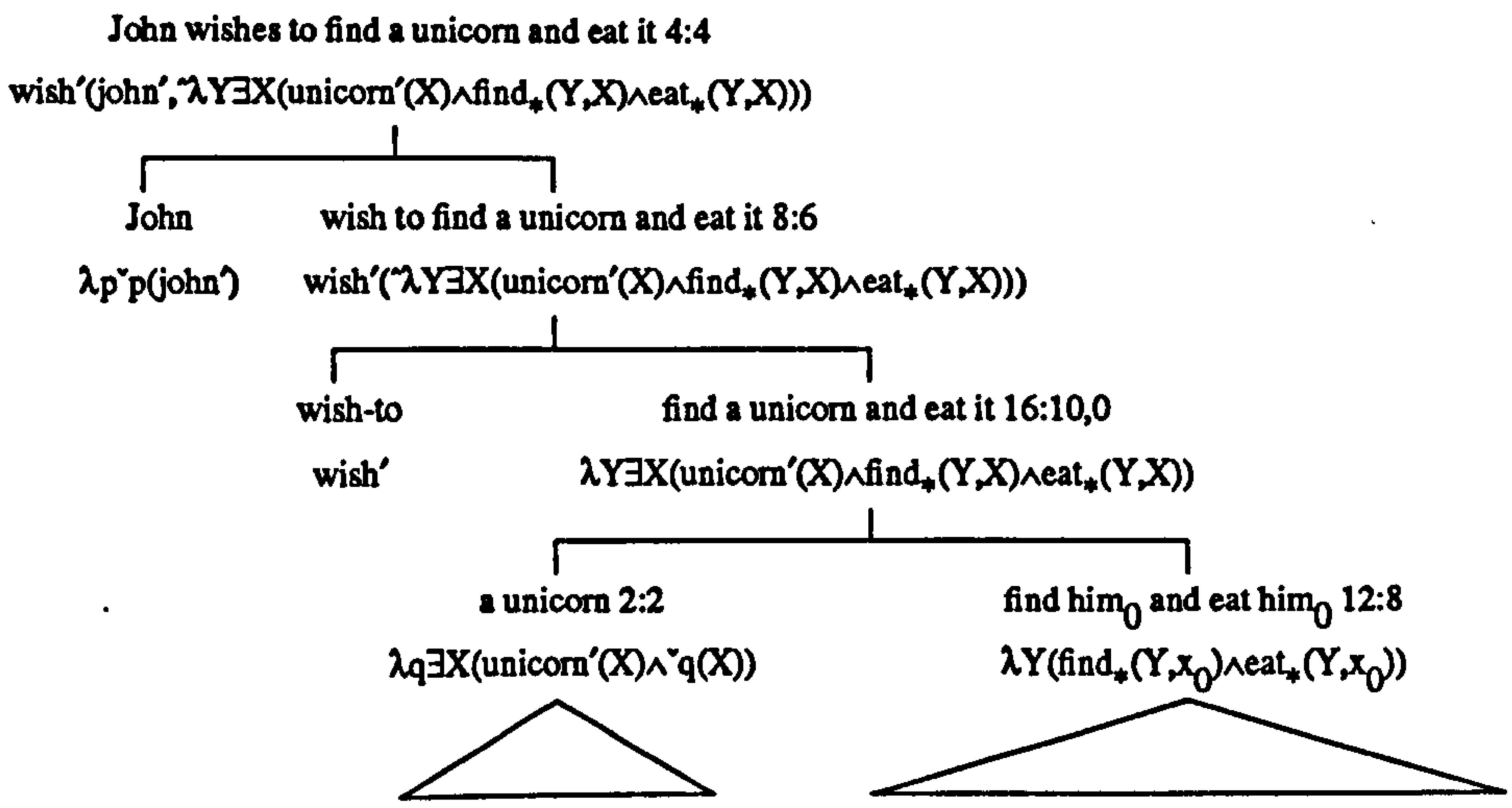


Fig 14

tion), and without which S16 would prove impotent.

Although the utility of S16 is evident only in its application to *compound* verb phrases, the grammar of PTQ provides no mechanism for restricting it to suitable cases. Just as rule S14 affords an alternative to S4 for the introduction of terms into sentences even when, as in fig 4 no semantic distinction is involved, so to S16 provides a sometimes vacuous alternative to S5 for the introduction of objects into verb phrases, leading to massive redundancy of analysis trees.

Montague also includes in PTQ a rule S15 allowing quantification into common noun phrases, but his reasons, at least so far as the linguistic coverage of the fragment is concerned, are obscure. The grammar of PTQ allows the output from one application of the relative clause rule S3 to serve as input to a further application of the same rule; thus it is possible to formulate the sentence:

(51) Every man such that he loses a pen voluntarily such that he finds it walks.

This sentence illustrates the phenomenon of "relative clause stacking" which some linguists apparently find unexceptionable. Friedman and Warren, [F3,erratum], investigating an earlier claim by Partee, [P4], confirm that the only way to give narrow scope to "a pen" in (51) is to quantify by S15 into the common noun phrase:

(52) man such that he loses him_1 voluntarily such that he finds him_1.

as illustrated in fig 15.

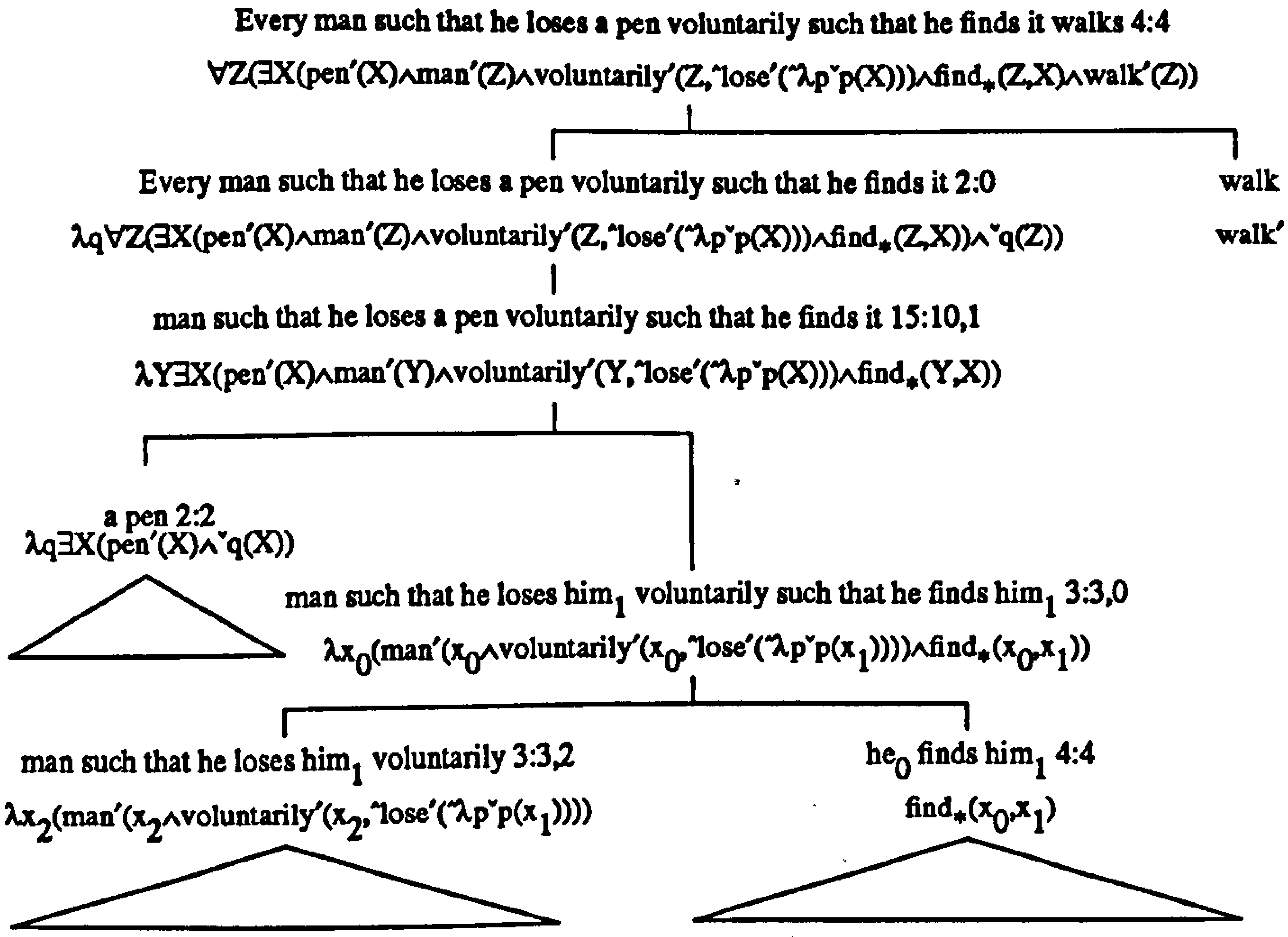


Fig 15

A *sufficient* condition for the retention of S15 is that the grammar generate a sentence S having *both* the following characteristics:

(a) S must *necessarily* involve relative clause stacking.

(b) The relative clauses in S must involve coreferential expressions.

Curiously these conditions tend to exclude each other in a grammar using the stilted relative clause formulation of PTQ; indeed as Friedman and Warren point out, the adverb in (51) is mandatory if the sentence is to conform to condition (a). Without the adverb sentence (51) becomes:

(53) Every man such that he loses a pen such that he finds it walks.

which is structurally ambiguous between *two* less stilted sentences of standard English:

(54) Every man who loses a pen who finds it walks. (?*)

(55) Every man who loses a pen which he finds walks.

If we interpret (53) as (54), then an analysis parallel to that of fig 15, but with "lose*(Z,X)" in place of "voluntarily′(Z,^lose′(^λp˅p(X)))" at the top node and compatible amendments at lower nodes, is one possibility; but there is the alternative analysis of fig 16 which interprets (53) as (55), makes no recourse to

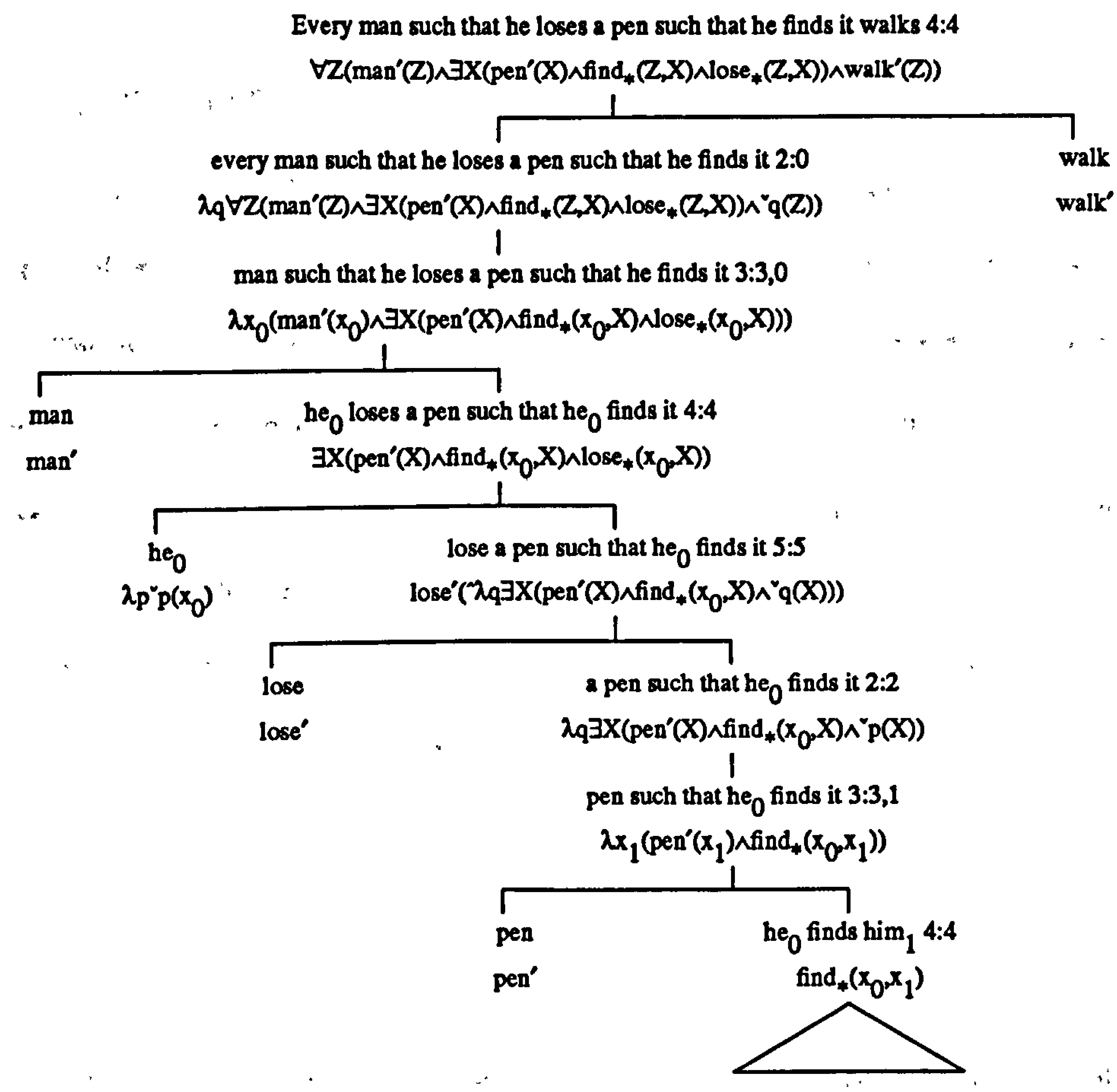


Fig 16

S15, and returns nonetheless an equivalent IL translation at the top node. The availability of alternative paraphrases for (53) is evidence that the sentence has characteristic (b), without which it would be analysable without S15 despite "stacking", but the alternatives vitiate characteristic (a): thus one characteristic is obtained at the expense of the other.

At this point I must confess that I personally find (54), and hence (51), unequivocally deviant, in other words "stacking" is not a phenomenon recognisable in my idiolect. I am for example unable to give to the sentence:

(56) The girl who kissed the man who broke the bank at Monte Carlo has never gambled.

any interpretation in which it is essentially contradictory.[†34]

†34. There may of course be *conjoined* relative clauses, and with three or more conjuncts we might expect all but the last conjunction to be suppressed, but these seem to me to be derived from a common noun and a conjoined sentence.

It would be possible to eliminate "stacking" altogether from PTQ by rewording the input condition of S3:

(S3a) If $\zeta \in P_{CN}$ and ζ is not dominated by 3:3,m, and $\Phi \in P_t$ ....†35

a revision which would make S15 strictly redundant so far as the PTQ fragment were concerned; but whatever the merits of proscribing "stacking", S15 might still be required by grammars with a wider coverage of English. The presence of characteristics (a) and (b) in sentences generated by a grammar, while sufficient to justify S15, does not constitute a necessary condition for its retention.

Janssen suggests, [J4], that examples not involving "stacking", and for which S15 is needed, may arise with grammars which allow common noun phrases of the form "friend of ...".†36 Thus in order to give narrow scope to "a woman" in

(57) Every picture of a woman which is owned by a man who loves her is a valuable object.

it is necessary to quantify into the common noun phrase:

(58) picture of him_1 which is owned by a man who loves him_1.

The real problem with S15 is that, like S16, there is no provision for restricting its use to appropriate cases, ie. cases where its application guarantees a semantic interpretation not otherwise available; thus the question of massive redundancy of analysis trees arises once more. Even within the framework of PTQ the problem of restricting S15 is non trivial. We could specify that the input CN must involve stacking by insisting on a structure having 3:3,m dominated by 3:3,q, but this would not guarantee the presence of coreferential sub components: what is more the restriction would need revision to accommodate extensions that generated (57).

It will transpire that in TMDCG S15 has been implemented, but with a switch set so that it is invariably inhibited in default of user intervention,while the restriction of S16 to compound verb phrases is handled by a subcategorisation facility to be introduced in due course.

---

†35. The legitimacy of rules which refer to the *structure* of their inputs is considered in the discussion of Partee's constraints in the next chapter.

†36. Consonant with our earlier analysis of "author of Waverley" (fn. 20), we might reformulate Janssen's example as:
(57') Every picture which represents a woman which is owned by a man who loves her is a valuable object.
thereby converting it to a "stacking" form.

2.3.5. Prepositions and Adverbs

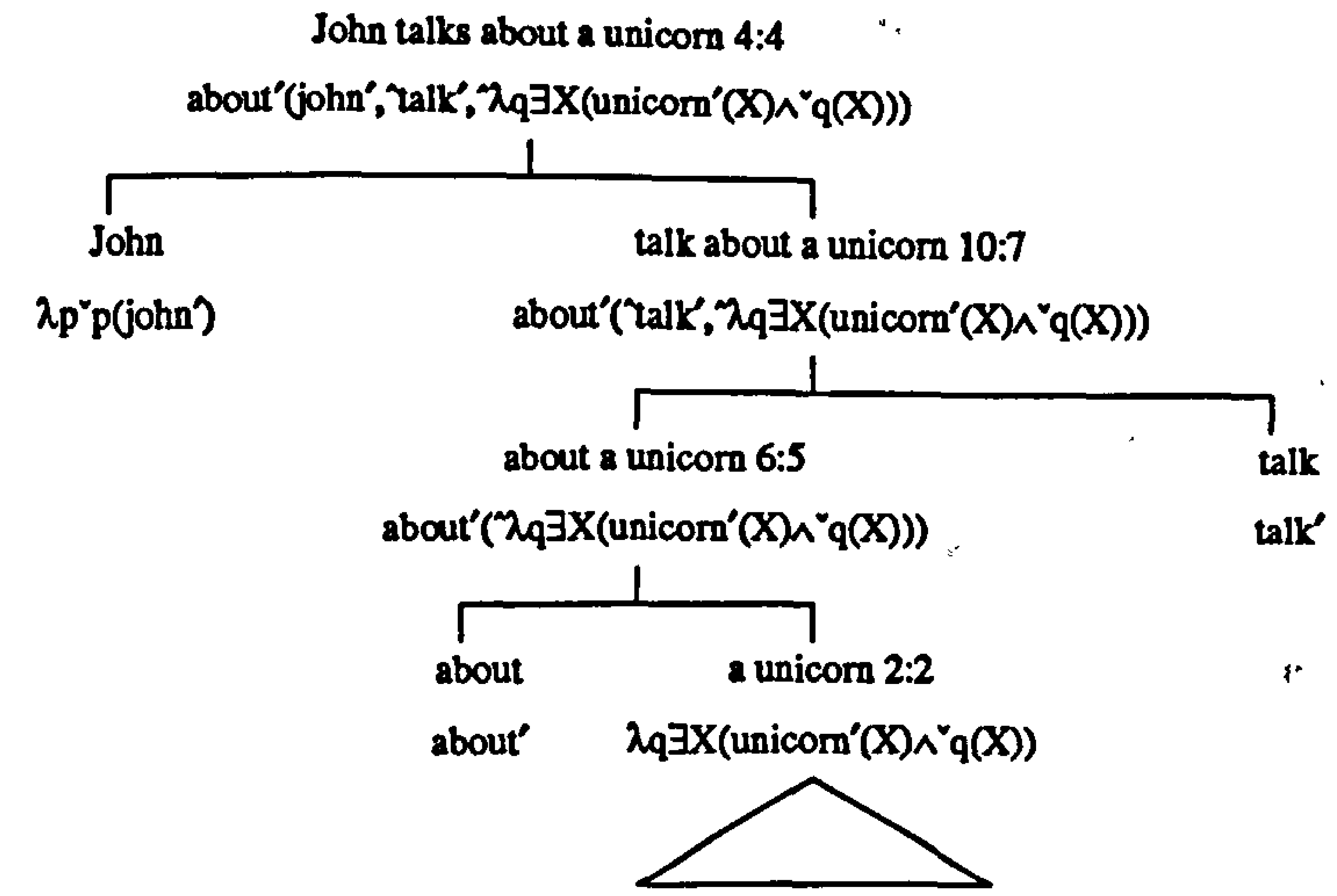


Fig 17

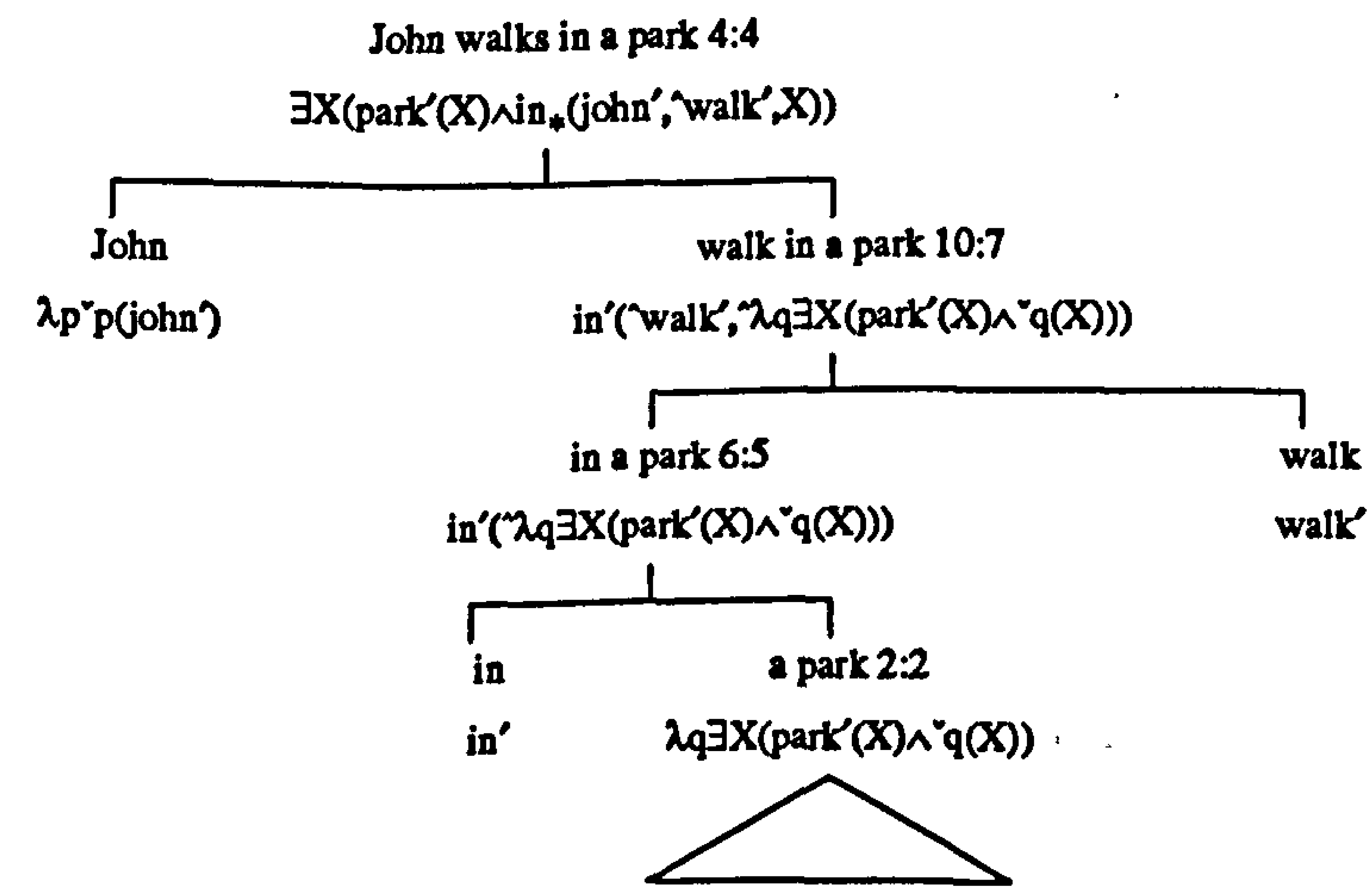


Fig 18

Verb phrase adverbs, verbs taking infinitival complements, and prepositional phrases in the fragment fulfill a common semantic role. All are in a category $IV/_{n}IV$, and map to a semantic type <<s<et>><et>> denoting functions from properties to sets of individuals.

The sub formula "$voluntarily'(x_2, {}^{\wedge}lose'({}^{\wedge}\lambda p\, {}^{\vee}p(x_1)))$" of fig 15 states that x_2 is in the set of individuals that voluntarily have the property of losing x_1, while the final formula of fig 14 indicates that John is in the set of those who wish to have the property of finding a unicorn and eating it.

The function of a preposition in the fragment is in effect to make a verb phrase adverb out of a term, but the appearance of postulate MP8 warns us that a distinction is to be made between *intensional* and *extensional* prepositions, a distinction illustrated by the examples:

(59) John talks about a unicorn.

(60) John walks in a park.

Sample analysis trees for which appear in fig 17 and fig 18. These trees correctly reflect our intuition that whereas the second sentence plainly entails the existence of a park, no ontological commitment to unicorns is involved in the first. The extensionality of the preposition "in" in fig 18 is achieved at the top node by the following reduction steps which take advantage of MP8 as expected.

Construction by T4: $\lambda p\,{}^{\vee}p(john')[{}^{\wedge}in'({}^{\wedge}walk',{}^{\wedge}\lambda q\exists X(park'(X)\wedge{}^{\vee}q(X)))]$

λ conversion: $in'({}^{\wedge}walk',{}^{\wedge}\lambda q\exists X(park'(X)\wedge{}^{\vee}q(X)))(john')$

Relational notation: $in'(john',{}^{\wedge}walk',{}^{\wedge}\lambda q\exists X(park'(X)\wedge{}^{\vee}q(X)))$

MP8 $\lambda q\exists X(park'(X)\wedge{}^{\vee}q(X))[{}^{\wedge}\lambda Y(in_{*}(john',{}^{\wedge}walk',Y))]$

λ conversion: $\exists X(park'(X)\wedge\lambda Y(in_{*}(john',{}^{\wedge}walk',Y))[X])$

λ conversion: $\exists X(park'(X)\wedge in_{*}(john',{}^{\wedge}walk',X))$

As Dowty succinctly expresses it, [D9], "in_{*}" is a function which given a place Y and an activity p returns as value the set of individuals who do p in Y.

****

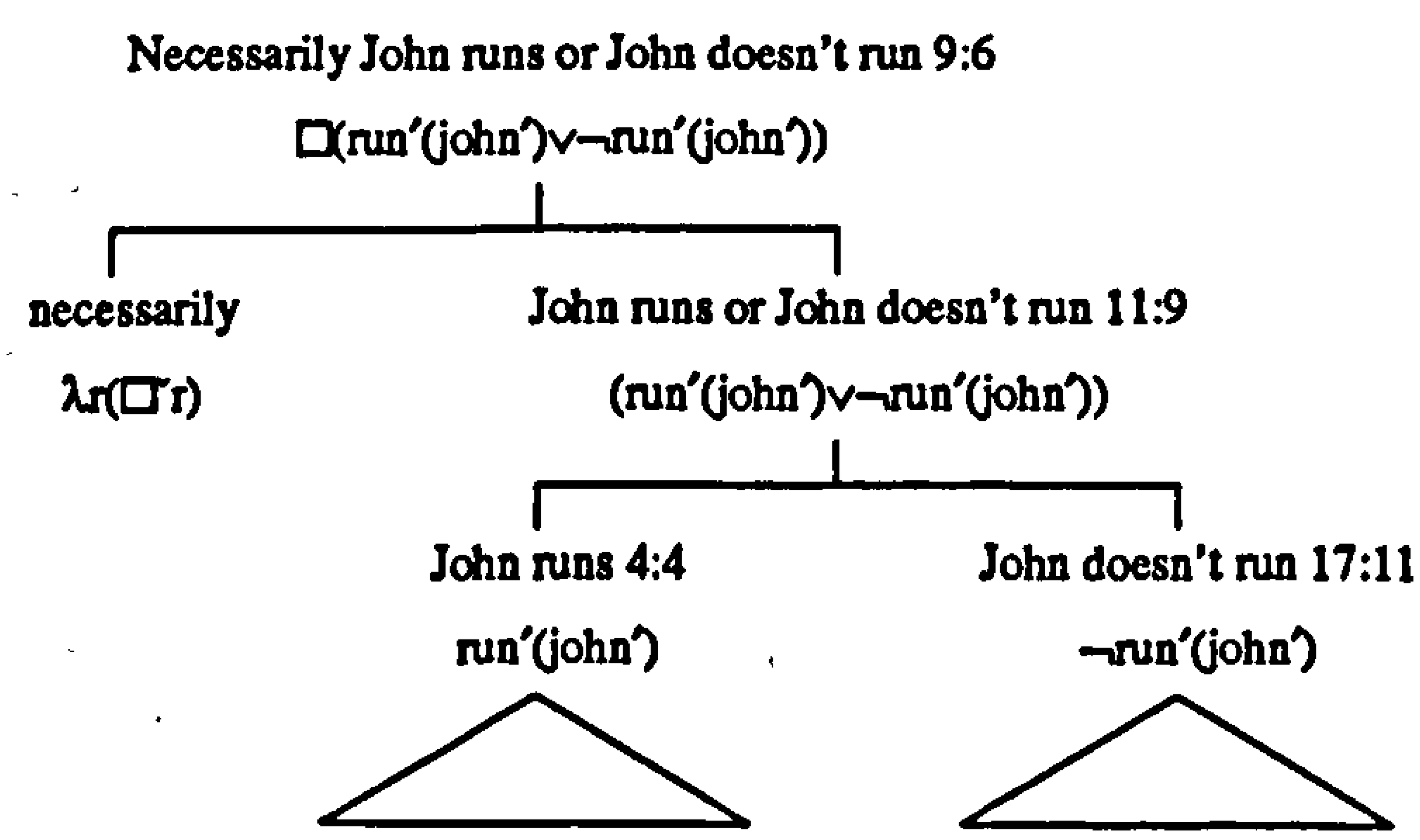


Fig 19

****

The rule for introducing *sentential* adverbs, like that for handling sentential conjunction and disjunction, is predictable and may be illustrated without comment by the tree in fig 19. This same tree may serve also to give to the "tense and sign" rule S17 the perfunctory attention which it deserves.

Montague's use of expressions like "negative third person singular present" indicate total indifference to the grammatical structure of English synthetic verb forms, failing as it does to afford any status to the finite periphrastic auxiliary which must be introduced as a carrier for the negative particle. A *minimum* condition for any adequate account of tense and sign must be the provision of a suitably restricted recursive mechanism for making new members of P_{IV} out of auxiliaries (members of $B_{IV/_n IV}$ for some *n*) and old members of P_{IV}.

As it stands any generalisation of the S17 technique would require the introduction of a separate structural operation for each possible positive and negative chained auxiliary combination, which even if we disallow the recursive loop associated with the semi-auxiliary"go" would necessitate some 296 variations to handle active and passive indicative alone.†37

A compositional analysis of tense and aspect satisfying the above minimum condition is incorporated in TMDCG.

2.3.6. Disjunctive Terms

Only one rule has not yet been illustrated and that is the noun phrase disjunction rule, S13, which is introduced into the grammar in order that a distinction may be made between the interpretations of sentences like:

(61) Every woman loves John or every woman loves Bill.

(62) Every woman loves John or Bill.

A correct interpretation of (62) , with the women distributed over both John and Bill, is illustrated in fig 20.

With regard to S13, Dowty, [D9], makes the puzzling comment that conjoined as opposed to disjoined terms are omitted from the grammar (there is no combination 13:8):

---

†37. A valiant attempt is made by Janssen, [J3], to reduce the population explosion by recourse to hyperrules whilst retaining Montague's own idiom, but only a small sub set of signed tense and aspect combinations is handled, and the recursive nature of the constructions is not considered. As Janssen states, the object of the exercise is to demonstrate the utility of hyperrules rather than to improve upon the fundamental treatment of tense and aspect.

****

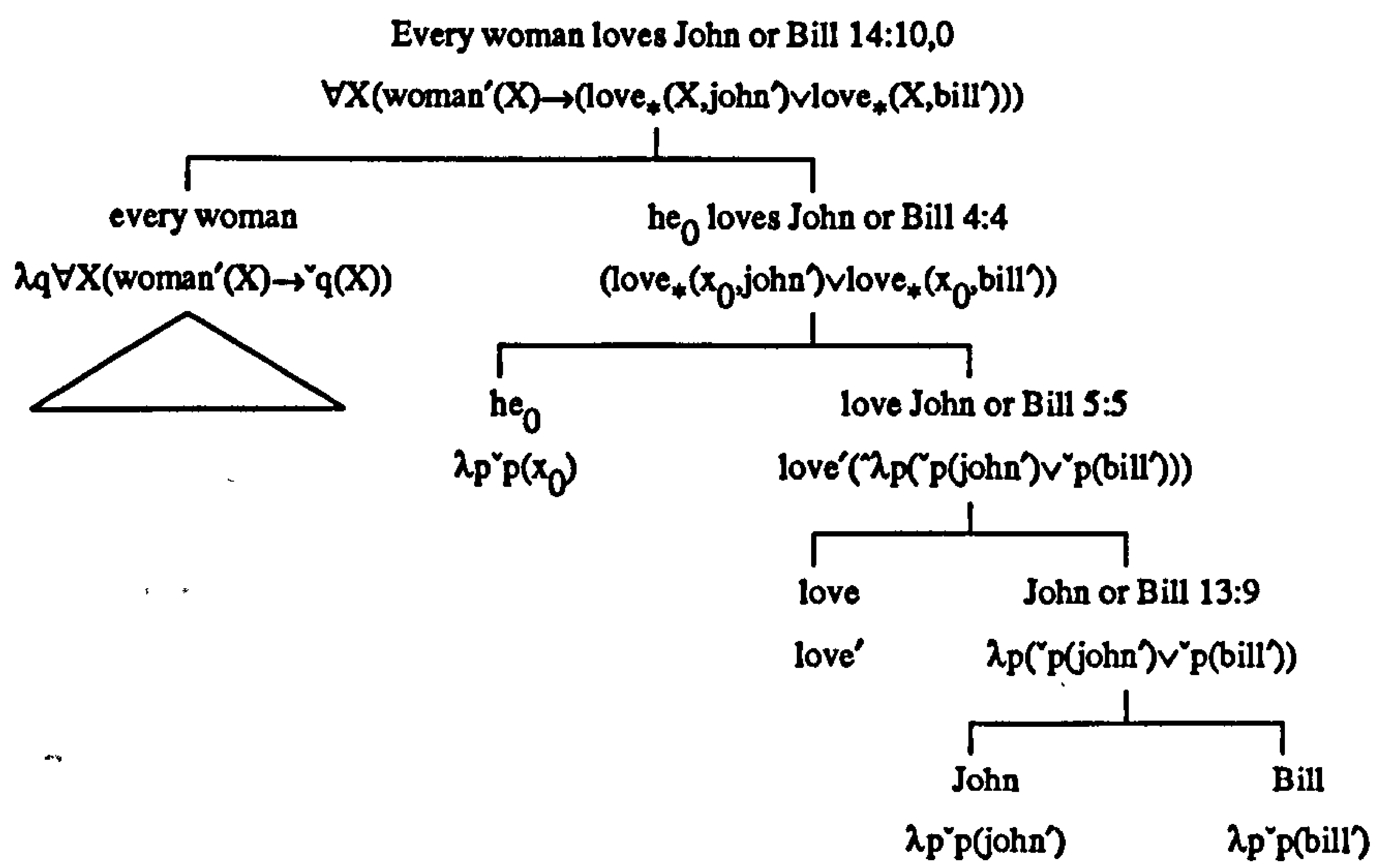


Fig 20

****

"purely for the sake of simplicity, since these would require plural rather than singular verb forms".

Montague's cavalier approach to feature matching is evidenced by his invitation in the definitions of f_3 and f_{10} to identify *informally* the gender of members of B_{CN}, and one suspects that he would have regarded issues of concord as being, from a logicians point of view, equally uninteresting, and to be avoided by equally informal methods. The real reason for the exclusion of plural terms must, it seems to me, have been the absence at the time of any adequate account of the *plural* definite determiner. With the advent of Barwise and Coopers' account of generalised quantifiers, [B6], it becomes possible to remedy this lacuna by agreeing that the referent of "the ζs" is that set of entities characterised by some contextually supplied "witness predicate" W and which also satisfy ζ. Hence:

the (plural) $\rightsquigarrow$ $\lambda p \lambda q \exists W \forall X(\check{W}(X) \rightarrow (\check{p}(X) \wedge \check{q}(X)))$.

where W is of type <s<et>>, but such modification takes us beyond exegesis.

CHAPTER 3: CORRECTIONS & CONSTRAINTS

¶ Known inadequacies in the original grammar of PTQ are well documented. This chapter commences with a discussion of crucial errors involving structural operations f_3, f_4, f_5 and f_{10} identified by a number of authors, and considers the alternative solutions offered. Partee's proposed constraints on the form of acceptable rules in a Montague grammar are next considered and a PROLOG style notation for accommodating them is suggested. Finally Janssen's alternative hyperrule notation is described.

3.1. Catalogued Errors in PTQ

Between them Partee, [P3, P6], Thomason, [T4], Friedman, [F5], and Janssen, [J3], have identified a number of errors in the original grammar of PTQ, and alternative methods of correction have been suggested. These methods share a common characteristic in that all permit structural operations to access some form of structural analysis of their inputs, and to maintain some form of structural analysis of their outputs. Similar devices have also featured in various attempts to extend rather than correct the grammar. Recourse to such tactics inevitably prompts the question raised by Partee, [P6], of whether or not some limitations should be placed upon the possible form of structural operations, a question concerning which Montague himself provides scant advice.

Erroneous sentences generated by PTQ include the underlisted examples adapted from those discussed by Friedman, [F5][†38]

(63) * John walks and talk. (Source of error: f_4.)

(64) * Mary loves Bill and she seeks John or he. (Source of error: f_5.)

(65) * Mary loves Bill and she talks about he or John. (Source of error: f_5.)

(66) * Bill runs and Mary loves John or he. (Source of error: $f_{10,n}$.)

(67) * John or Mary finds a fish and he eats it. (Deviant if "he" is assumed coreferential with the

†38. Some of Friedman's examples involve only *latent* errors because they contain outstanding syntactic variables which *may* be bound in such a fashion as to generate acceptable sentences at the topmost node. Thus she lists "seek John or he_4" as an erroneous application of S5 although the error would be corrected by quantification provided that the occurrence of he_4 remained a *first* (ie. non anaphoric) occurrence. By contrast (64) involves an actual error. A tree with an acceptable top node dominating a deviant subordinate node would however infringe Partee's well formedness constraint, to be discussed below. Example (63) originates in Partee [P3].

disjunction "John or Mary". Source of error: $f_{10,n}$.)

(68) * John seeks Mary or Bill and he finds her. (Deviant if "her" is assumed coreferential with the disjunction "Mary or Bill". Source of error: $f_{10,n}$.)

The first three examples, (63)...(65), illustrate malfunctions in the structural operations triggered by S4, S5, and S6 respectively. Case (63) arises because structural operation f_4, invoked by rule S4, introduces a finite form for only the *first verb* of its verb phrase argument, whereas this marking should be extended to the head verb of each conjunct or disjunct in a compound verb phrase. Sentences (64) and (65) may be derived by consecutively quantifying first "Bill" and then "Mary" into the phrases:

"he_0 loves he_1 and he_0 finds John or he_1"

and

"he_0 loves he_1 and he_0 talks about he_1 or John".

In the first instance "find John or he_1" has been generated by S5, and in the second "about he_1 or John" results from an application of S6. The problems arise because both S5 and S6 invoke f_5 which introduces accusative marking only in case the input term is an elementary syntactic variable. Where the input term is compound, case marking should in fact apply to each syntactic variable which constitutes an atomic disjunct.

Examples (66)...(68) all involve inadequacies in either the substitution mechanism or the cross-referencing provisions of structural operation $f_{10,n}$. In each case the operation has been invoked by the sentence quantification rule S14, but parallel examples using S15 or S16, which trigger the same operation, would be possible. As Friedman observes, [F5], a syntactic variable may occur within a member of P_t in one of four guises:

(i) First and nominative.

(ii) First and accusative.

(iii) Subsequent and nominative.

(iv) Subsequent and accusative.

Occurrences of type (i) and (ii) require substitution by a *term*, while cases (iii) and (iv) demand replacement by *surface anaphora.*

Quantification presents no difficulties in connection with the substitution of terms for *first and nominative* variables, but a problem occurs whenever $f_{10,n}$ is offered a member of P_t containing a *first and accusative* variable him_n as one argument, and for the substituens a disjoined term containing as a disjunct some variable he_k. The source of the difficulty is that $f_{10,n}$ case marks the input substituens only when it is a variable occurring in isolation: the result is a sentence like (66), the derivation for which is illustrated in fig 21, where the error occurs at 14:10,1 level.

****

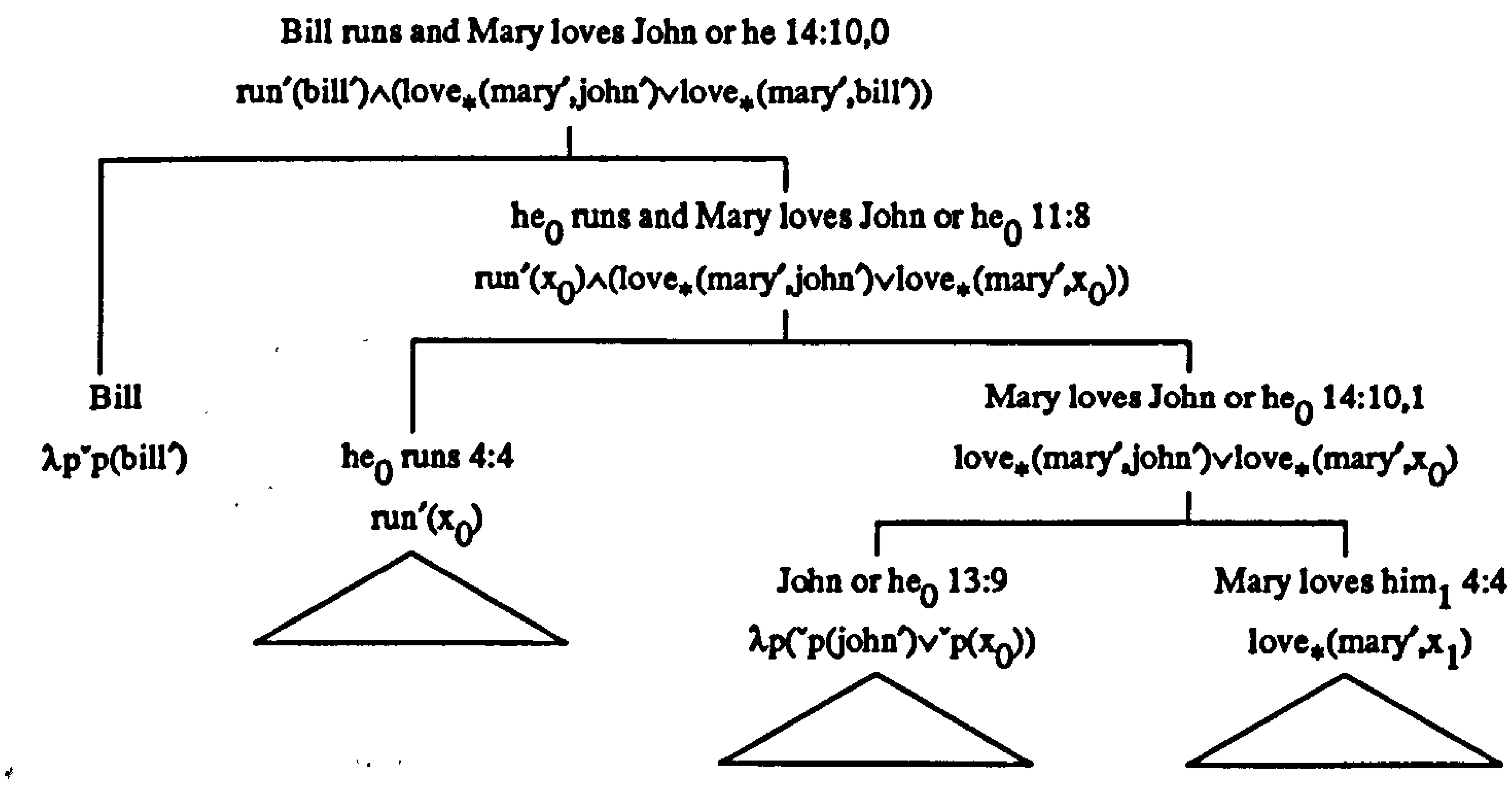


Fig 21

****

Given *any* disjoined term as one argument and a member of P_t containing either a *subsequent and nominative* variable he_n or a *subsequent and accusative* variable him_n, the operation $f_{10,n}$ will generate dubious anaphoric references in which a singular pronoun purports to be coreferential with the whole disjunction. The *subsequent and nominative* case is exemplified by sentence (67), the problematic derivation for which appears in fig 22, while example (68) typifies the *subsequent and accusative* variation assuming a derivation as in fig 23. In both cases the problem arises because $f_{10,n}$ introduces a surface pronoun matching in gender the *first* member of B_{CN} or B_T in the input term.

Bennett, [B8], Partee, [P4], and Thomason, [T4], all comment on the absence of reflexivisation in the grammar of PTQ, although only Partee and Janssen, [J2], explicitly describe this as an error.[†39] Assuming

†39. Montague was apparently aware of this inadequacy, [M4,footnote 12], but regarded it as of no philosophical interest.

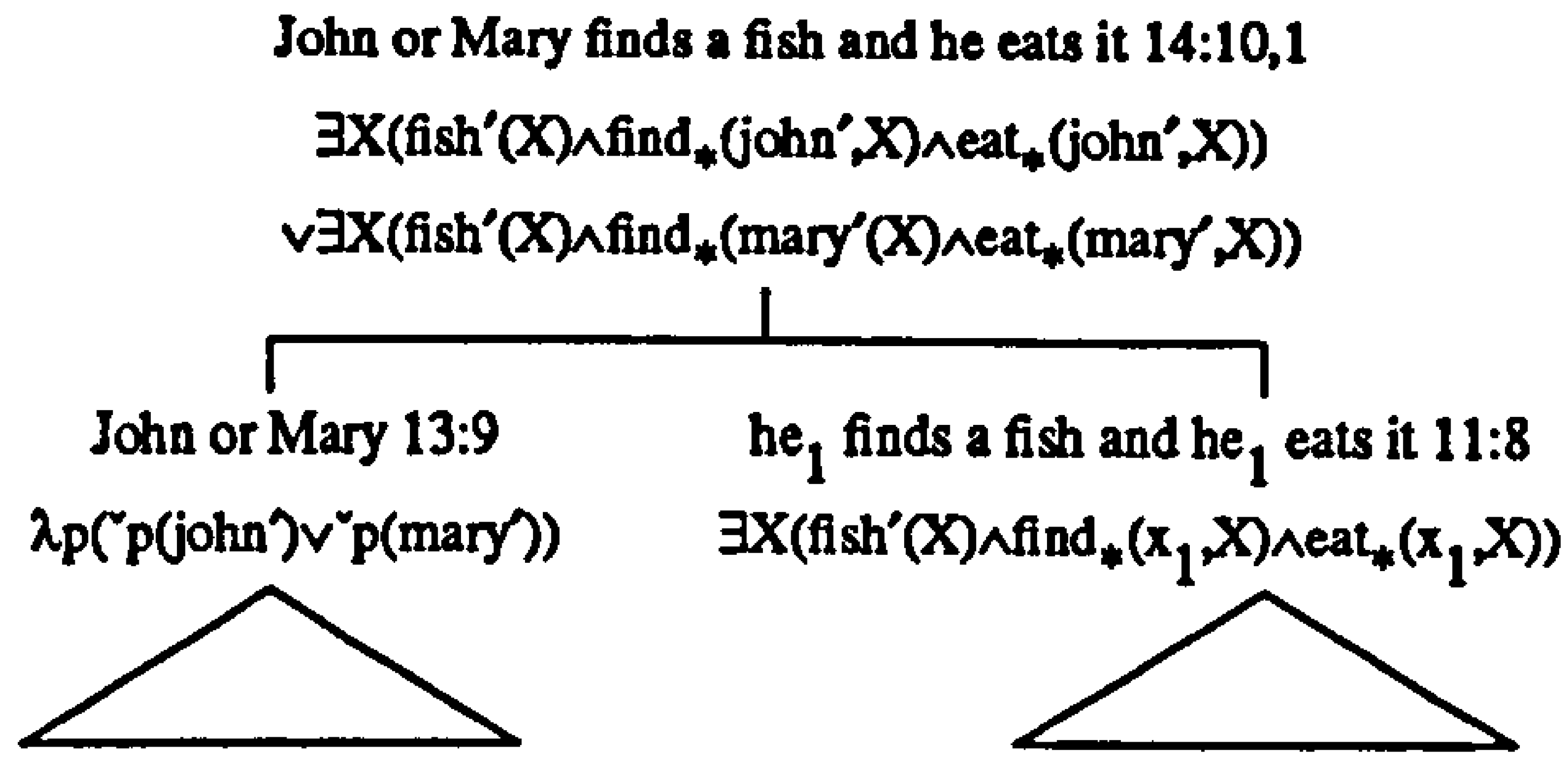


Fig 22

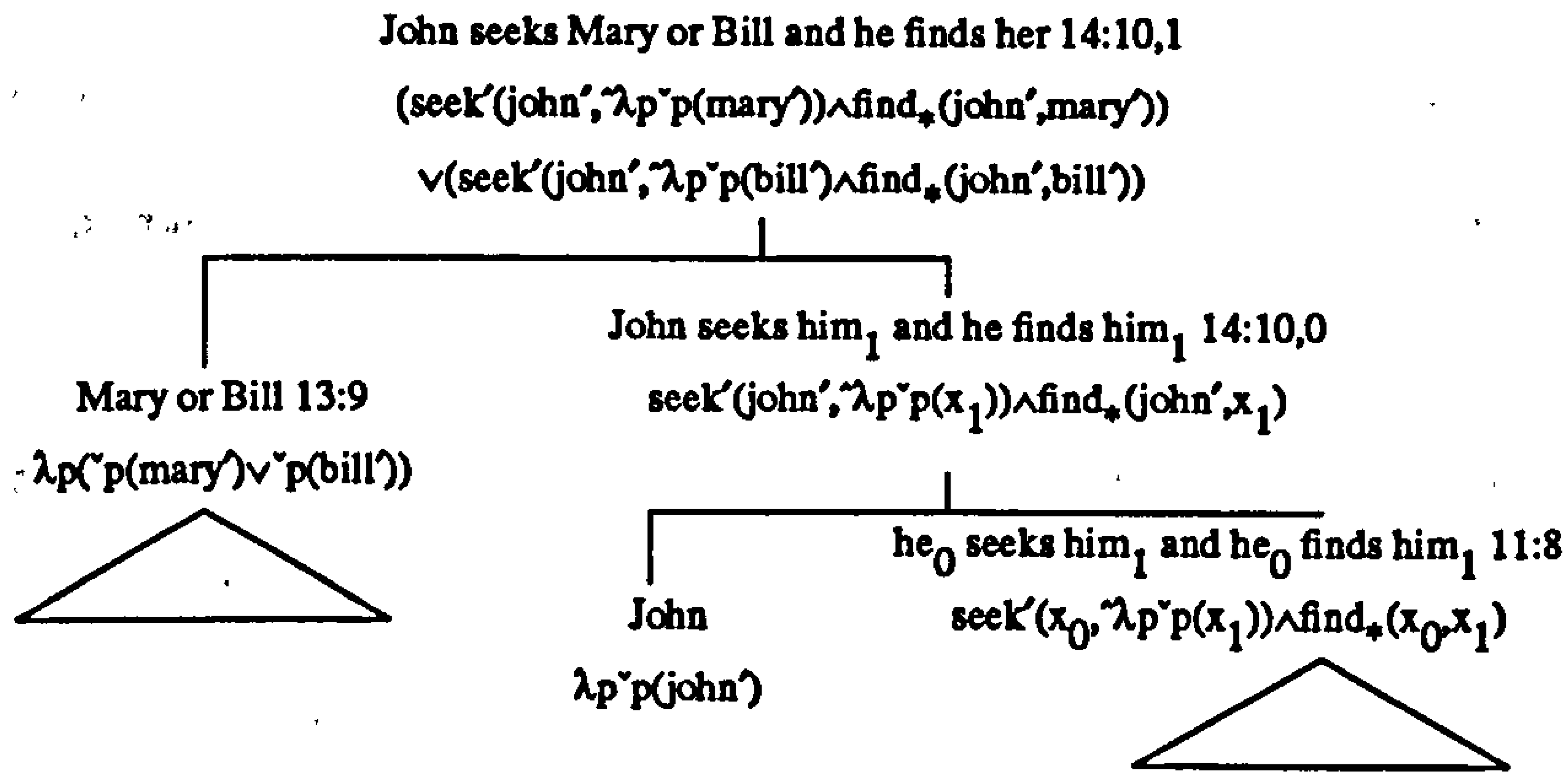


Fig 23

an augmented lexicon, PTQ allows:[†40]

(69) * John shaves him. (Deviant if the pronoun makes anaphoric reference to the subject. Source of error: f_4.)

An accusative pronoun coreferential with the subject in a simplex sentence should always be reflexive. This is a condition which continues to hold when the subject is itself a pronoun making anaphoric reference to a head term occurring outside the simplex as in:

(70) John hopes that Mary believes that he shaves himself.

Furthermore it holds when the head term is not itself nominative:

(71) Mary loves John and he loves himself.

†40. It will prove tedious to mention each extension to the lexicon needed for the generation of particular examples, thus when the nature of the extension is obvious it will be introduced without comment.

Bennett and Thomason both locate the source of the inadequacy in operation f_4 which allows expressions of the form:

"he_1 shaves him_1"

and suggest amendments which would insist upon:

"he_1 shaves $himself_1$".

Curiously, the accommodation of the reflexive turns out to be one of the few manoeuvres which are easier to accomplish in the original generative style of syntax than in its definite clause grammatical inverse.

The condition specified by S3 and S14 under which $f_{3,n}$ and $f_{10,n}$ may respectively be applied does not stipulate that a syntactic variable containing index n must necessarily appear in the sentential input $\Phi \in P_t$.†41 This lack of constraint gives rise to instances of vacuous application which Janssen, [J4], has dubbed "not there cases".

A vacuous application of S14 is syntactically innocuous since the redundant term argument is simply eliminated, as demonstrated in fig 24: but this combined analysis and translation tree serves also to illustrate that the application has been semantically pernicious. Allegedly there is a reading of:

"John loves Mary".

which entails the existence of unicorns!

****

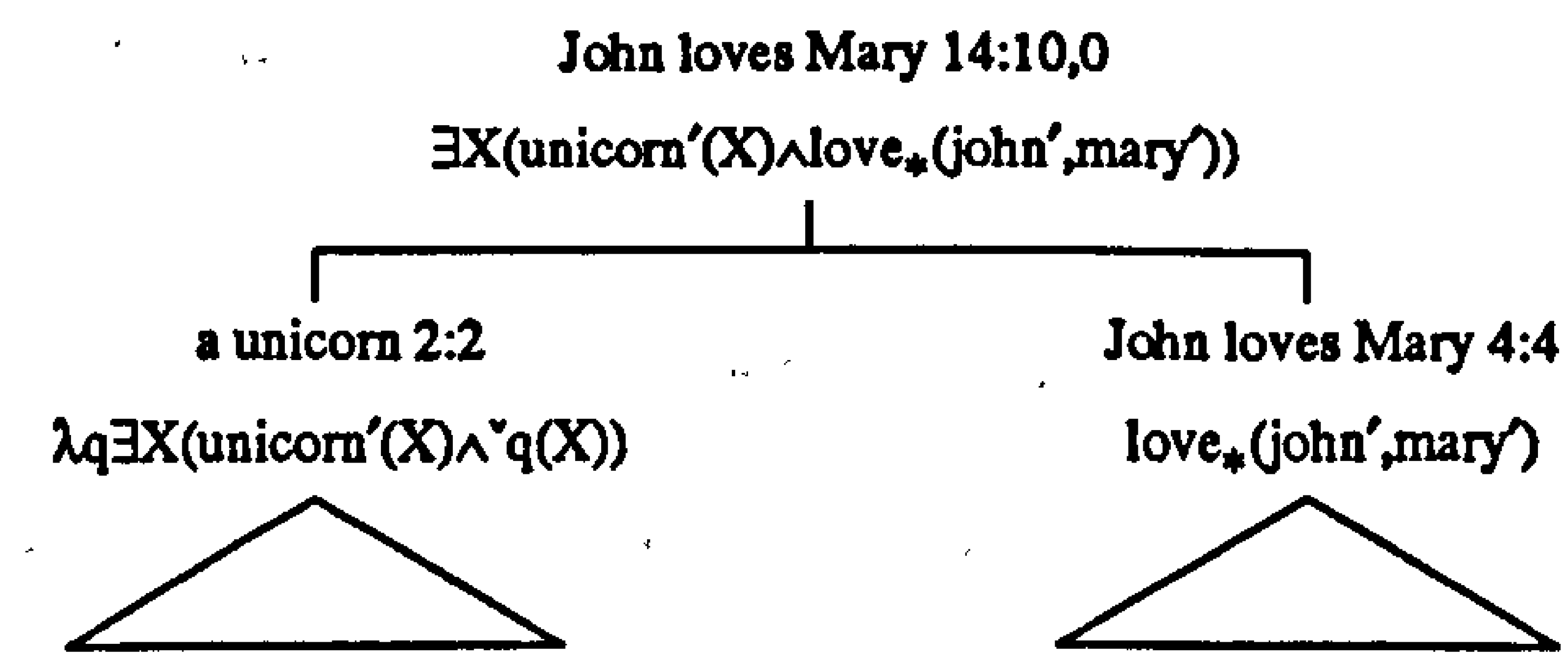


Fig 24

****

Sometimes a vacuous application of S3 may give rise to a *syntactically* deviant sentence such as:

(72) John loves the man such that Mary talks.

---

†41. Similar licence is tolerated by S15 and S16.

but a more subtle error discussed by Janssen, and derived originally from Groenendijk and Stokhof, arises when an apparently acceptable sentence is assigned an improper translation. The sentence:

(73) John loves the man such that he walks.

may be derived in PTQ by the process illustrated in fig 25 which includes a vacuous application of 3:3,2.

****

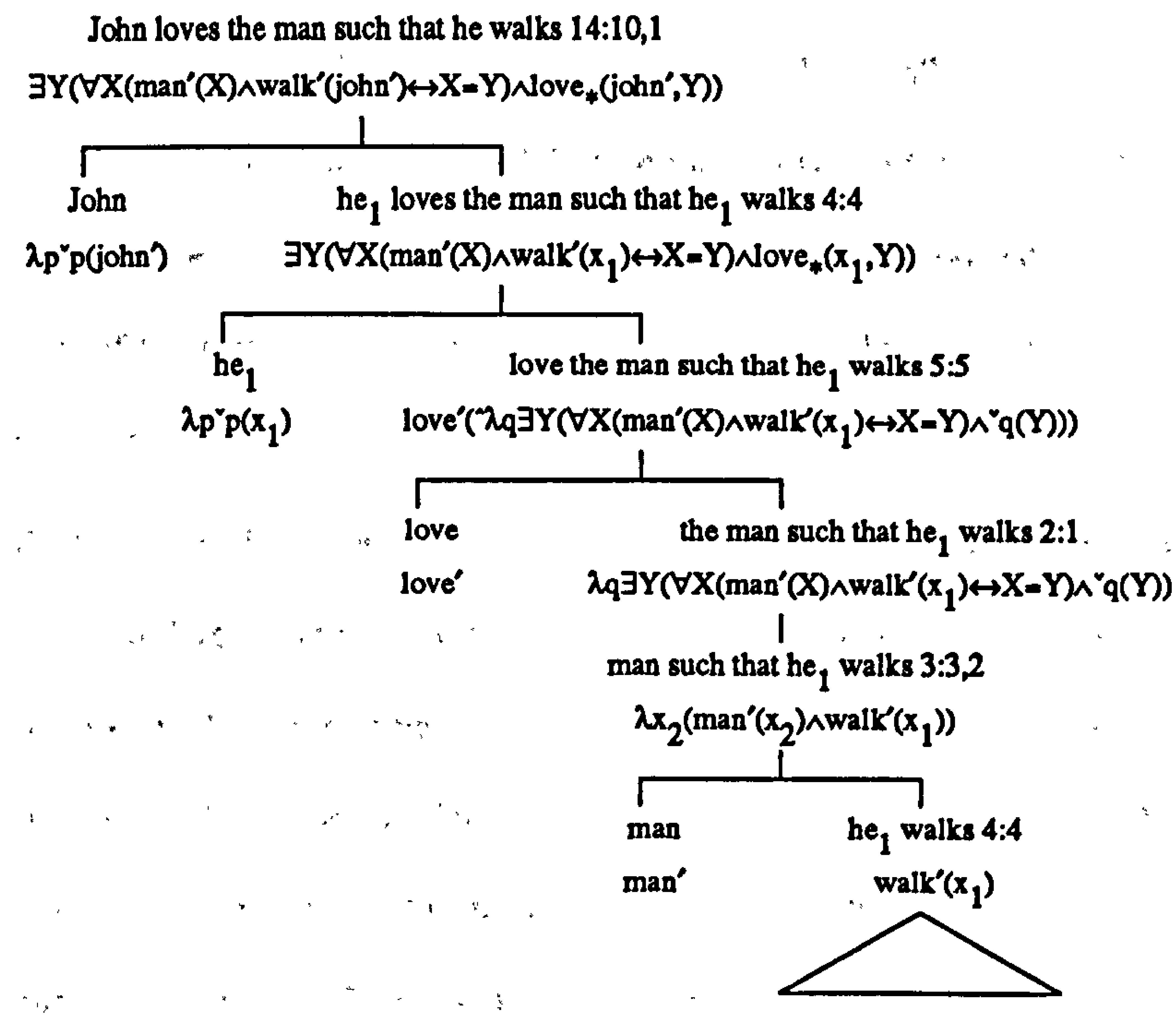


Fig 25

****

Inspection of the topmost node reveals a translation according to which there is only one individual answering the description "man", John loves that person, and coincidentally John walks; whereas the only acceptable interpretation of (73) requires John to love a man who is identifiable as a walker.

To summarise therefore, the following forms of amendment to the structural operations of PTQ appear to be required:

$f_{3,n}$ should be applicable only in case a syntactic variable with index *n* appears in its sentential argument.

f_4 should introduce finite forms for *all* head verbs in its verb phrase argument, and should reflexivise appropriate variables therein in case the term argument is a syntactic variable.

f_5 should case mark every syntactic variable which constitutes a head term within its term argument.

$f_{10,n}$ should be applicable only in case a syntactic variable with index n appears in its second argument. Furthermore special provision should be made for cases where a first and accusative variable is to be replaced by a compound term, and where subsequent variables make anaphoric reference to compound terms.

3.2. Friedman's Unlabeled Bracketing Solution

That f_4 should introduce finite forms not for all verbs but only for all *head* verbs in its verb phrase argument is obvious from the fact that "try to walk" is an intransitive verb phrase which should combine with "John" to give "John tries to walk". Indeed, given that PTQ allows no preceding adverbial modifiers, Montague's statement of f_4 correctly identifies the head verb in verb phrases of arbitrary complexity, but fails for verb phrase compounds.

In order to identify *all* the head verbs in an intransitive verb phrase it is necessary to be able to distinguish between a *compound* verb phrase having conjunction or disjunction as the most recently applied operation, and a *complex* verb phrase including a conjoined or disjoined verb phrase complement in an embedded position. Partee, [P3], was the first to observe that this distinction cannot be made without recourse to *constituent structure* because "try to walk and talk" is ambiguous. Her solution was to adopt the conventional labeled bracketing of Transformational Grammar and to distinguish:

(74) $_{iv}[_{iv}[$try to $_{iv}[$walk$]]$ and $_{iv}[$talk$]]$.

(75) $_{iv}[$try to $_{iv}[_{iv}[$walk$]$ and $_{iv}[$talk$]]]$.

Assuming the availability of some method for *inspecting* the constituent structure, the first, which has "and" as primary operator, may be identified as a compound with two head verbs, and the second as a complex with only one.

An *unlabeled* bracketing alternative solution is adopted by Friedman, [F5],whose preference, although not explicitly justified, appears to be motivated by a desire to make the constituent analysis look as much like a conventional data structure (viz. list) in a computer programming language as possible. In Friedman's reformulated PTQ rules, all basic expressions are bracketed†42 and all structural operations

†42. The inclusion of brackets round basic expressions is described by Friedman as a "preferred option" which she proceeds *not* to exercise on the grounds of readability. I find this mystifying since the fully bracketed notation is almost exactly comparable with Partee's minus the labels.

except f_{10} introduce outer brackets, so that for all A, each member of P_A is a balanced bracketed expression. The alternatives to (74) and (75) then become:

(76) [[[try to] [walk]] and [talk]].

(77) [[try to] [[walk] and [talk]]].

Friedman next introduces a number of functions defined on bracketed constituents which may in effect be specified as follows:

(a) First term

If $\gamma \in P_T$ and $\gamma = [\alpha \text{ or } \beta]$ then firstterm(γ) = firstterm(α) $\cup$ firstterm(β).

If $\gamma \in P_T$ and $\gamma \neq [\alpha \text{ or } \beta]$ then firstterm(γ) = $\{\gamma\}$.

(b) First variable

If $\gamma \in P_T$ then firstvar(γ) = $\{h: h \in$ firstterm(γ) $\wedge$ h is a syntactic variable$\}$.

(c) First intransitive verb phrase

If $\gamma \in P_{IV}$ and $\gamma = [\alpha \text{ and } \beta]$ or $\gamma = [\alpha \text{ or } \beta]$ then firstivp(γ) = firstivp(α) $\cup$ firstivp(β).

If $\gamma \in P_{IV}$ and $\gamma \neq [\alpha \text{ and } \beta]$ and $\gamma \neq [\alpha \text{ or } \beta]$ then firstivp(γ) = $\{\gamma\}$.

(d) First verb

If $\gamma \in P_{IV}$ then firstverb(γ) = $\{\upsilon: \exists\delta(\delta \in$ firstivp(γ) $\wedge$ υ is leftmost member of B_{Verb} in $\delta)\}$.

(where $B_{Verb} = B_{IV} \cup B_{TV} \cup B_{IV/t} \cup B_{IV//IV}$)

The informality introduced by the phrase " leftmost member" could be eliminated by adopting forthwith the PROLOG list notation according to which [H|T] is a list having H as first element and having a tail T comprising a list of the remaining elements. The empty list is represented as []. We might then define:

firstbasicv([]) = {}.

firstbasicv([H|T]) = firstbasicv(H) iff firstbasicv(H) ≠ {} else firstbasicv([H|T]) = firstbasicv(T).

firstbasicv(V) = {V} iff V $\in B_{Verb}$ else firstbasicv(V) = {}.

and hence:

(d') First verb

If $\gamma \in P_{IV}$ then firstverb(γ) = $\{\upsilon: \exists\delta(\delta \in$ firstivp(γ) $\wedge$ υ = firstbasicv(δ))$\}$.

Armed with these functions Friedman is able to state her corrections of f_4, f_5 and f_{10} as follows:

(e) $f_4(\alpha,\delta) = [\alpha\ \gamma]$ where γ comes from δ by replacing each member of firstverb(δ) by its 3rd. person singular present.

(f) $f_5(\delta,\beta) = [\delta\ \beta']$ where β' comes from β by replacing each member of firstvar(β) of the form he_n by him_n.

(g) $f_{10,n}(\alpha,\phi)$ comes from ϕ by replacing:

(i) Any first and nominative occurrence of he_n by α.

(ii) Any first and accusative occurrence of him_n by α' where α' comes from α by replacing each member of firstvar(α) having form he_k by him_k.

(iii) Any subsequent and nominative occurrence of he_n by α'' where α'' comes from α by replacing each member of firstterm(α) *not* of form he_k by a nominative surface pronoun matching in gender the first member of B_{CN} or B_T in α.

(iv) Any subsequent and nominative occurrence of he_n by α''' where α''' comes from α by replacing each member of firstvar(α) of form he_k by him_k and all other members of firstterm(α) by an accusative surface pronoun matching in gender the first member of B_{CN} or B_T in α.

Since Friedman does not identify reflexivity as a problem, it is unsurprising that the correction to f_4 makes no allowance for this phenomenon: within her terms of reference it succeeds in that it successfully introduces finite verb forms at all appropriate places. Likewise the amendment to f_5 achieves the intended correction.

With regard to the changes to $f_{10,n}$ the situation is more problematic. Sub clauses (i) and (ii) are unexceptionable, for (i) merely retains the status quo from PTQ, while (ii) guarantees that the node at 14:10,1 level in fig 21 will acquire the label "Mary loves John or him_0", thus correcting (66): the other two sub clauses however seem unacceptable. Given the changes in sub clauses (iii) and (iv), the topmost nodes of the analysis trees in figs 22 and 23, which correspond to (67) and (68), become:

(78) John or Mary finds a fish and he or she eats it.

(79) John seeks Mary or Bill and he finds her or him.

That Friedman accepts (iii) and (iv) at all is puzzling since she has already observed that the replacement of "Mary" by "Bill" in (78) would demand the whimsical anaphoric reference "he or he". The introduction of a more neutral phrase such as "one of them" might alleviate this difficulty, but there is a more serious problem. Sentence (78) could be true in cases where John does the finding and Mary the eating, however the translation tree in fig 22 indicates an explicit interpretation in which whoever finds also eats. The anaphoric disjunction introduced by (iii) suggests a further choice which the semantic representation rules out, thus the syntax and semantics are no longer in step. Indeed the nuance captured in PTQ when an indexed variable becomes coreferential with a disjoined term is difficult to reproduce in English without a custom built circumlocution such as "whichever candidate fulfills the condition".

Similar considerations apply in connection with sentence (79) and the root translation of fig 23. The semantic representation implies that, whatever the utterer's uncertainty concerning the object of John's quest, John is single minded in his seeking and achieves his objective; but such an interpretation is hard to reconcile with the surface sentence resulting from the adoption of (iv). Faced with such difficulties it might even be better to adopt the solution, with which Friedman toys, of proscribing anaphoric reference to disjunctions altogether.

3.3. Handling Reflexivity

In her earlier papers, [P2, P3, P4], Partee suggests a solution to the problem of reflexivising appropriate pronouns which adapts from Transformational Grammar the convention that a transformation may be defined in terms of the partition of the input structure into a fixed finite number of factors. This tentative solution results in the introduction of a new reflexive rule which cites an analogous form of structural analysis as one condition for its application:

(RR) If $\phi \in P_t$ and ϕ is a simplex sentence of the form α he$_k$ β him$_k$ γ then $f_R(\phi) \in P_t$ where $f_R(\phi) = \alpha$ he$_k$ β him$_k$self γ.

Two problems, both acknowledged by Partee herself, are evident in this rule, which it may be noted makes f_R a partial function. Firstly the condition that the input be a *simplex* sentence is imprecise, and must presumably make additional and covert reference to the tree structure built to date; secondly, the reflexive rule RR so formulated must be *obligatory*. Prior to the application of RR a sentence level node must have been constructed, the label of which is an ill formed expression: the function of RR is thus corrective. Partee's later constraints, to be discussed in the next section, proscribe solutions having this characteristic.

Thomason, [T4], adopts a general strategy of making the inputs to structural operations not expressions alone but *analysed* expressions, where an analysed expression is a pair $<\alpha,T>$ such that α is an expression and T an analysis tree for α. Rather than generate an incorrect sentence and then correct it, Thomason constrains his equivalent to f_4 to generate correct reflexive anaphora in the first place. When the input *term* is of form he_k, appropriate occurrences of him_k within the input *verb phrase* must be replaced by $him_k self$, where appropriate occurrences are simply those not already dominated by a sentence level node embedded within the analysis tree for that verb phrase. Expressed in terminology matching as closely as possible that of PTQ†43 and assuming that all head verbs are to be made finite by a technique such as Friedman's, Thomason's suggestion may be formulated as follows:

(S4) If $\alpha \in P_t$ and has analysis T and $\delta \in P_{IV}$ and has analysis T′ then

$f_4(<\alpha,T>,<\delta,T'>) \in P_t$.

$f_4(<\alpha,T>,<\delta,T'>) = [\alpha\eta]$ where γ comes from δ by replacing each member of firstverb(δ) by its third person singular present, and in case $\alpha = he_k$ replacing each occurrence of him_k in δ that is not dominated by a sentence level node in T′ by an occurrence of $him_k self$.

Whether or not this solution is considered acceptable depends upon one's attitude towards the heavy reliance placed by Thomason upon the appeal to previous derivational history.

Perhaps the most elegant solution is that offered by Bennett, [B8]. Whenever an accusative syntactic variable is introduced by f_5, Bennett decrees that it be flagged with an asterisk.†44 A subsequent application of S4 where the input term is of form he_k reflexivises all variables of form $*him_k$ and deletes all *other* asterisks, while an application of S4 having any other form of input term deletes *all* asterisks. Bennett's rule formulations are obfuscated by the desire to accommodate phenomena not germane to the present discussion, moreover he adopts an alternative technique for identifying head verbs. If however we extract the

†43. Thomason's terminology seems to me to deviate unnecessarily: moreover his own formulation makes no allowance for compound verb phrases.

†44. Bennett acknowledges a problem with prepositional phrases qualifying transitive verb phrases for which no solution is offered. The analysis discussed would generate "John takes a pen with himself", however the restriction required to prevent this remains obscure.

above suggestion for handling reflexivity and amalgamate it with previous results we derive the following:

$f_4(\alpha,\delta) = [\alpha\gamma]$ where γ comes from δ by

(i) replacing each member of firstverb(δ) by its 3rd. person singular present and

(ii) in case α is of form he_k replacing all occurrences of $*him_k$ by $him_k self$ and

(iii) deleting all remaining asterisks.

$f_5(\delta,\beta) = [\delta\beta']$ where β' comes from β by replacing each member of firstvar(β) of form he_k by $*him_k$.

In both Thomason's and Bennett's solutions, reflexivisation is triggered by the occurrence of a syntactic variable in subject position when the subject predicate rule is applied. It is immaterial at what point thereafter the variable becomes eventually bound by a quantification rule, hence the accommodation of sentences like (70) and (71) presents no special difficulties.

The price paid for Bennett's solution is the introduction of asterisked indexed variables. Should these be included in the lexicon or are they, as Bennett implies, ordinary variables marked with a feature? Are we entitled to call a string containing asterisked variables a well formed expression, and if not should such strings be admitted as node labels? Partee's insistence that innovations in Montague grammar be subject to constraints raises just such questions as these.

3.4. Partee's Constraints & Innovations

Montague's general theory, as expounded in UG, was designed to characterise "language" as an abstraction of which natural language is but a concrete realisation. It is Partee's contention, [P5], that if the main interest in the theory lies in its applicability to *natural* language, and given the linguist's preoccupation with the maximally restrictive classification of possible natural languages, restraints on the form of admissible rules must be introduced.

3.4.1. Constraints Ascribed to Montague

Only three constraints are attributed by Partee to Montague himself. These may be described as follows (where the numbering departs from that of Partee's original exposition):

(C0):The Compositionality Constraint.

To every syntactic rule S*n* there must correspond a semantic rule T*n* which constructs a semantic representation for the output of S*n* from the semantic representations of its inputs.

The apparent strength of C0 is vitiated according to Partee by the absence of restrictions both on the form which structural operation f_k might take and on the form of correspondence between f_k of S*n* and g_j of T*n* other than that imposed by the requirement for category - type compatibility in their respective results.

C1:The Free Order Constraint.

The only precondition for the application of a rule should be the availability of members of the input categories required. For no *n*, *m* is it stipulated that S*n* must be applied before S*m*, ie. there is no *extrinsic* rule ordering.

C2:The (Weak) Well Formedness Constraint.

The output from each rule S*n* must be a well formed expression.

According to the general theory of UG, of which PTQ is an instance, a language L is to be characterised as a pair:

<DL,R>

such that DL is a *disambiguated* language and R is an ambiguating relation. In PTQ expressions of the disambiguated language are represented by complete analysis trees and R is simply the relation:

$\{\langle\alpha,l\rangle: \alpha$ is an analysis tree $\wedge \exists m(\langle l,m\rangle$ labels the root of $\alpha) \wedge l \in$ *Esen*$\}$

That is to say, the ambiguating relation involves deletion of all the tree except for the expression field of the root label, thus leaving behind a surface sentence of English.

Partee's dissatisfaction with the weak well formedness constraint stems from the fact that only at the root of an analysis tree corresponding to an English sentence do we have empirical evidence for determining well formedness. Ostensibly *anything* could count as a well formed expression at other nodes provided that it were licenced by rule modifications.

3.4.2. Proposed Additional Constraints

In order to invest C2 with some restrictive powers Partee suggests that it be replaced by a stronger formulation which is tantamount to:

C3:The (Strong) Well Formedness Constraint.

The output from each rule S*n* must be a well formed expression of the disambiguated language DL; and such expressions may differ from those of L in at most that they may include:

(i) labeled bracketing to be deleted by the ambiguating relation

(ii) indexed syntactic variables

(iii) morphological representations *having phonological realisation.*

Given this constraint, deviant English expressions cannot become embedded in the output from any S*n*. An immediate consequence is that no rule need be obligatory,†45 for an obligatory rule is corrective in nature. Accordingly Partee's own earlier formulation of the reflexive rule RR becomes untenable.

Morphological representations take the form of feature specifications in Partee's revision, but the ambiguating relation does not process such specifications. There is thus an implication that the form of expression returned at the root node will vary from that generated by PTQ. The syntax component of the grammar will produce strings containing root forms of lexical items together with feature markings to be converted into surface forms by a morpho - phonological component. A simple sentence like "John runs" will be generated by the grammar in the form:

(80) $\text{John}_{[\text{Masc, 3rd., Sg, Nom, MT+}]}\ \text{run}_{[\text{Pres, 3rd., Sg, MV+}]}$.†46

By adopting C3 as formulated one is committed to considerably more than the elimination of ill formedness at subordinate nodes: the introduction of morphological representations is innovative.

What now happens to Bennett's analysis of reflexives? Asterisks must presumably be classified as morphemes, but do they have a phonological realisation? The answer appears to be "sometimes", ie. when asterisked variables are reflexivised but not when the asterisks are merely deleted. In the latter case they are purely abstract morphemes which C3 proscribes. Partee's claim is that only in cases of phonological realisation is there empirical evidence to justify the retention in the analysis of a morphological representation,

†45. For some reason Partee regards the restrictions "no obligatory rules" and "no purely abstract morphemes" as separate constraints, not implications of C3.

†46. Partee represents feature bundles in conventional matrix form, each row being a feature:value pair. For simplicity of exposition I show such bundles as lists of values. Provided that the value sets for features are pairwise disjoint there need be no ambiguity.

and that without such evidence there adoption would be gratuitous: but what sort of evidence would justify their *elimination*? Given two sets of rules achieving a common end one of which employs abstract morphemes while the other does not there might indeed be good grounds for preferring the latter, but can we be sure that such an alternative always exists? I confess to a vested interest in defending Bennett's usage since in TMG I propose a number of markers p of which ("?yn", "?" and "agent") have no phonological realisation[†47] I must therefore express reservations regarding the restriction encapsulated in C3 (iii).

Just as Friedman's unexplained preference for unlabeled brackets appears to derive from their resemblance to computational data structures, so Partee's alternative preference seems to stem from a conservative predilection for the conventions of Transformational Grammar. The question of how best to reconcile the conflicting design requirements of a notation both familiar to the practising linguist and computationally convenient may at this juncture fruitfully be addressed. Partee would accept as a top level structural analysis of a simplex sentence:

(a) $_s[_{np}[\Phi]_{vp}[\Psi]]$.

Since this is unrecognisable as a data structure for computational purposes Friedman would prefer:

(b) $[[\phi][\Psi]]$

ie. a list in which the ultimate elements after "flattening" would be terminal symbols, the onus of recalling to what category each sublist belongs devolving on the user.

A possible compromise would be to introduce the category as first member in each list and to adopt:

(c) [s,[np,Φ],[vp,Ψ]].

Provided that we remember that the first element is always an alien to be eliminated when restoring the surface string there need be no confusion: however the intrusion of control information into the data could be considered offensive. By abandoning list notation and opting for PROLOG style structures as our mode of representation we might arrive at:

(d) s(np(Φ),vp(Ψ)).

†47. The first is used in the construction of Karttunen style "proto questions" and the second to introduce a logical subject for agentless passives.

Whereas lists may be of variable length, structures, to be consistent, need a constant number of argument positions: thus the notation would be satisfactory were all nodal brackets in the original, labeled or otherwise, to enclose a *pair* of elements. Lists could then be replaced by binary predicates except at leaf positions where the predicates would be unary. Such uniformity is unfortunately undermined by cases such as:

$_{cn}[_{cn}$[man] such that $_{s}$[he walks]].

I have never subscribed to the myth that bracketing impairs readability, and accordingly I suggest that the virtues of (b) and (d) be amalgamated in the PROLOG style structure:

(e) s([np([Φ]),vp([Ψ])])

in which the category symbol becomes a unary predicate the sole argument of which is a list of *n* elements, those of comparable form being substructures. Elimination of the parentheses virtually restores Partee's notation, thus despite the "noise" of which some complain, this notation preserves both the linguist's conventions and computational feasibility. A further advantage is that in due course further argument places could be added to accommodate phenomena such as features without intrusion on the basic information held in the constituent list. For ease of reference I shall refer to such an alternation of structures and lists as a *"structured list"*.

Labeled brackets are, as one might expect, introduced in Partee's version of PTQ by the structural operations. In order to prevent any operation from being used retrospectively to change the structural analysis of its inputs, Partee introduces:

C4:The Structure Building Constraint.

No rule S*n* may add more than an outer pair of brackets labeled with the output category. There can be no internal structure building.

The avowed motivation for Partee's introduction of labeled bracketing is no longer merely that the convention is essential if (74) and (75) are to be distinguished. She writes:

"my motivation ... is to make it possible to restrict the class of rules by making labeled bracketing one of a limited set of properties of expressions to which rules may refer, and in particular to disallow reference to previous stages of the derivational history."

Hence we derive the further constraint:

C5:The Historical Appeal Constraint.

No rule *Sn* may refer to the derivational history of its inputs, though it may refer to their derived constituent structure as represented by labeled bracketing.

At first sight C5 appears to involve a distinction without a difference. Labeled bracketing is essentially an alternative notation for representing tree structure. The only feature of an analysis tree *not* represented in the labeled bracketing notation is the citation of the operation number with which the expression at each node is paired, but the introduction of a category index as a label along with the brackets largely compensates for this loss. Observe for example that Thomason's reformulated S4 rule, a paradigmatic example of recourse to derivational history, requires only the identification of a node having category S (ie. t): it is unnecessary to determine which rule for making members of this category has been applied. The constraint would be vacuous were it not for the fact that Partee imposes restrictions on the permissible forms of reference to labeled bracketing formulated in terms of her innovations.†48

3.4.3. Partee's Innovations

One innovation, viz. the replacement of surface sentences at the root of analysis trees by strings composed of lexical roots plus feature bundles, has already been noted. This may be formulated as:

(In1) Syntax should be separated from morphology. The S rules should produce strings suitable for input to a conventional morphological component to be defined elsewhere.

The features specified in morphological representations are to be viewed as the values of recursively defined properties of expressions. Just as the S rules provide a recursive definition for each category of expression, associated PS (property specification) rules may furnish a corresponding definition of the features of members. Such a definition supposes that features may be identified functionally hence we arrive at:

(In2) A property is a function which applied to an expression yields a feature as value.

†48. Although Partee numbers her constraints (somewhat differently from the manner here adopted) her introduction of innovations is rather less formal. The exposition in the following section accordingly imposes a formal structure not present in the original paper.

and

(In3) Each rule S*n* may have an associated property specification rule PS*n* which determines features of the output in terms of features of the input.

Seven basic property functions are identified of which the first four are classified as morphological, the fifth as categorial, and the sixth and seventh as special. The seven, together with their value sets, are:

gender(α) $\in$ {Masc, Fem, Neut, Com} iff $\alpha \in P_T$.

person(α) $\in$ {1st., 2nd., 3rd.} iff $\alpha \in P_T$.

number(α) $\in$ {Sg, Pl} iff $\alpha \in P_T$.

case(α) $\in$ {Nom, Acc} iff $\alpha \in P_T$.

verb(α) $\in$ {V+, V-} = V+ iff $\alpha \in B_{Verb}$.

pro(α) $\in$ {H+, H-} = H+ iff α is a pronoun or syntactic variable.

index(α) $\in$ {I-, I1, I2, ...} = I*n* if α has subscript *n*, I- otherwise.

Partee also introduces as tactical alternatives to Friedman's functions "firstterm" and "firstverb" two *relational properties* which could in effect be specified as fulfilling the equivalence conditions:

mainterm(α,β) $\in$ {MT+, MT-} = MT+ iff $\alpha \in$ firstterm(β).

mainverb(α,β) $\in$ {MV+, MV-} = MV+ iff $\alpha \in$ firstverb(β).

There is however a subtle distinction between Partee's relational properties and Friedman's functions. The latter inspect freely available bracketed structures and identify members of the appropriate sets by discovering their structural relationships. The former on the other hand are used polymorphically both to inspect preset flags and, in the case of PS rules, to set other flags by stipulation. Direct recourse to constituent structure becomes unnecessary because items that would be detectable were such recourse permitted now bear distinctive markers: the flag, as Partee admits, is an overt encoding of derivational history. Although the relational properties are proposed as a means of *avoiding* direct reference to constituent structure, any stipulative use in PS rules which failed to preserve the above equivalences would be improper. Indeed were the equivalences to be treated as *definitions* the PS rules would survive as descriptive summaries of the effects of percolation. Such definitions would be legitimate for, as Friedman observes, the behaviour of her functions would not be prejudiced by the introduction of bracket labels.

It will prove convenient to introduce a final property function which Partee adopts only informally in her own exposition, but which is needed to extract the *category* of an expression:

cat(α) = the category of α as indicated by labeled bracketing.

Of the above properties, only "cat" makes specific reference to the labeled bracketing allowed by C3.

Absence of any restriction upon the form which structural operations may take has already been identified by Partee as a deficiency. The availability of recursively defined properties, which structural operations are allowed to access and modify, contributes to her solution as summarised in the following innovative constraint:

(In4) Each structural operation f_k must be specifiable as a composition of subfunctions themselves defined in terms of primitive operations which may access or modify properties.

A total of five primitive operations are provided as a basis for subsequent definitions. Subject to some minor cosmetic variations in style of presentation, these may be specified as follows:†49

concat(n,ξ,<α_1,...,α_n>) = $_\xi[\alpha_1 ... \alpha_n]$.

sub(α,β,δ) = the result of substituting α for β in δ.

esub(α,β,δ) = the result of substituting α for all occurrences of β in δ provided that index(β) ≠ I-.

specify(β,ξ) = β marked with feature(s) ξ.

copy(ξ(γ)) = the value returned by ξ(γ) where ξ is a property.

Of these primitive operations, only "concat" directly refers to labeled bracketing, and then only to introduce an outer pair as licenced by C4.

The subfunctions defined in terms of these primitives are:

nom(α) = sub(specify(β,[Nom]),β,α for all β such that mainterm(β,α) = MT+.

acc(α) = sub(specify(β,[Acc]),β,α) for all β such that mainterm(β,α) = MT+.

---

†49. I have adopted a more strictly linear formulation of both the primitive operations and the subfunctions than that given by Partee herself. She prefers to present features in matrix form, and does not introduce a special predicate for "specify" and "copy". Her version of "concat" is of the form: $concat_{n,\xi}(\alpha,\delta)$, where the first subscript indicates the number of input arguments and the second the category label. In essence my formulations are equivalent.

$agr(\alpha,Tense) = [Tense,copy(number(\alpha)),copy(person(\alpha)]$.

$attach(\xi,\delta) = sub(concat(2,cat(\gamma),<\gamma,\beta>),\gamma,\delta)$ for all γ such that $mainverb(\gamma,\delta) = MV+$.

$proform(\alpha) = concat(1,T,<specify(he,[copy(gender(\alpha)),copy(number(\alpha)),H+,I-])>)$.

$prosub(\alpha,n,\delta) = esub(proform(\alpha),he_n,\delta)$.

One final innovation must be recorded before we turn our attention to the solutions proposed for the problems catalogued above. Partee decrees:

(In5) The structural operation invoked by rule S*n* may be a partial function to be applied only upon fulfillment of a structural analysis condition SA*n*.

The purpose of this concession, which was latent in her early treatment of reflexivity, is now to eliminate vacuous quantification and vacuous relativisation, thus contributing a solution to the problems identified by Janssen. What is curious is that in commenting on this concession Partee continues to insist that within a structural analysis condition *direct* reference to constituent structure may only be formulated in terms of a partition into a fixed finite number of factors, an insistence which treats the convention from Transformational Grammar adopted in her earlier papers as sacrosanct. In this idiom, to the infelicity of which we shall in due course revert, the condition that a syntactic variable with index *n* must occur in the input sentence ϕ before quantification is permitted is included in the following, which also isolates the first such occurrence:

$\phi = {}_t[\gamma\ \delta\ \xi]$ where $\delta = he_n$ and γ does not contain he_n.

3.4.4. Partee's Reformulations

Extremal clauses for Partee's relational properties are contained in a revised rule for lexical introduction which reads:

(S1) For all $A \in Cat$, if $\alpha \in B_A$ then $concat(1,A,<\alpha>) \in P_A$.

PS1 (i): If $\alpha \in B_T$ then $mainterm(\alpha,concat(1,T,<\alpha>)) = MT+$.

PS1 (ii): If $\alpha \in B_{Verb}$ then $mainverb(\alpha,concat(1,Verb,<\alpha>)) = MV+$.

Given Partee's constraints and innovations, the problematic rules and structural operations have the following reformulations:

(S3) If $\alpha \in P_{CN}$ and $\phi \in P_t$ and ϕ conforms to SA3 then $f_{3,n}(\alpha,\phi) \in P_{CN}$.

SA3: ϕ must have form $_t[\beta\ he_n\ \gamma]$.

$f_{3,n}(\alpha,\phi)$ = concat(4,CN,<α,such,that,prosub(α,n,ϕ)>).

PS3: gender($f_{3,n}(\alpha,\phi)$)=gender(α) and number($f_{3,n}(\alpha,\phi)$)=number(α).

(S4) If $\alpha \in P_T$ and $\delta \in P_{IV}$ then $f_4(\alpha,\delta) \in P_t$.

$f_4(\alpha,\delta)$ = concat(2,t,<nom(α),attach(agr(α,Pres),δ)>).

PS4: mainverb(γ,attach(agr(α,Pres),δ) = MV+ for all γ such that mainverb(γ,δ) = MV+.

(S5) If $\delta \in P_{TV}$ and $\beta \in P_T$ then $f_5(\delta,\beta) \in P_{IV}$.

$f_5(\delta,\beta)$ = concat(2,IV,<δ,acc(β)>).

PS5: mainverb(γ,$f_5(\delta,\beta)$) = MV+ for all γ such that mainverb(γ,δ) = MV+.

(S14) If $\alpha \in P_T$ and $\phi \in P_t$ and ϕ conforms to SA14 then $f_{10,n}(\alpha,\phi) \in P_t$.

SA14 (i): pro(β) = H- for all β such that mainterm(β,α) = MT+.

SA14 (ii): $\phi = {}_t[\gamma\ \delta\ \xi]$ where $\delta = he_n$ and γ does not contain he_n.

$f_{10,n}(\alpha,\phi)$ = concat(3,t,<γ,α,prosub(α,n,ξ)>).

Apart from the fact that, as anticipated, it generates feature marked strings as required by C3 instead of surface sentences, Partee's reformulation of S4 is equivalent in power to Friedman's: neither offers a satisfactory account of reflexivisation. A corrected version of (63) is returned as:

(81) $_t[_T[\text{John}]_{[\text{Masc,3rd.,Sg,Nom,MT+}]}\ _{IV}[_{IV}[\text{walk}]_{[\text{Pres,3rd.,Sg,MV+}]}$ and $_{IV}[\text{talk}]_{[\text{Pres,3rd.,Sg,MV+}]}]]$.

while "John tries to walk" is generated in the form:

(82) $_t[_T[\text{John}]_{[\text{Masc,3rd.,Sg,Nom,MT+}]}\ _{IV}[_{IV//IV}[\text{try}]_{[\text{Pres,3rd.,Sg,MV+}]}$ to $_{IV}[\text{walk}]]]$.

This reformulated subject predicate rule succeeds provided that the main verbs are already correctly flagged in the *input* verb phrase, for the sub function "attach" forces agreement only on marked head verbs. When verb phrases are conjoined or disjoined by S12 the main verbs of *both* inputs retain their markings in the result since associated with S12 we have:

PS12: mainverb(β,$f_n(\gamma,\delta)$) = MV+ for all β such that mainverb(β,γ) = MV+ *or* mainverb(β,δ) = MV+, where n = 8 or 9.

However the property specification rule associated with S8 which introduces verb phrase complements removes the "mainverb" flag from the incoming complement:

PS8: mainverb(β,$f_6(\delta,\gamma)$) = MV+ for all β such that mainverb(β,δ) = MV+.

Partee's and Friedman's reformulations of f_5 are also equipotent and equally satisfactory: either will correct (64) and (65) as required. It is the amended version of S14 which is once again open to question.

The problem that arises when substituting for *first and accusative* variables is overcome in Partee's rule by the expedient of forbidding the quantifying in of *all* term phrases containing subscripted variables as main terms. This is achieved by SA14 (i). To prevent the quantifying in of variables *in isolation* is defensible on economic grounds since the manoeuvre so forbidden does no more than generate trivial "alphabetical variants", but why the derivation in fig 21 should be proscribed rather than corrected is far from clear.

With regard to the handling of *subsequent and nominative* and *subsequent and accusative* variables, Partee's rule can be made to work with some minor modification. As it stands results superior to Friedman's are achievable by Partee's rule whenever disjoined candidates for quantification contain disjuncts of differing genders; for in such cases Partee's property specification rule PS13 affords the whole the features "[Com(*mon*),Sg]". The effect of "prosub" in the definition of $f_{10,n}$ is then to replace occurrences of "he_n" with a pronominal form likewise marked "[Com,Sg]", and this could easily be realised by the morpho-phonemic part of the grammar as for example, "the one in question", thus avoiding the mismatch between syntax and semantics identified earlier.

A disjunction of form "$_T$[α or β]" having elements of uniform gender is however given the features "[copy(gender(α)),Sg]" by PS13, the property specification rule associated with S13, consequently "prosub($_T$[α or β],*n*,ξ)" can do no more than introduce a pronominal form "$he_{[Masc,Sg]}$", "$he_{[Fem,Sg]}$" or "$he_{[Neut,Sg]}$" as the case may be: anaphora are handled exactly as in PTQ.

The correction of sentences (67) and (68) presents no problem since the disjoined noun phrases have the desired characteristic of multiple gender, but were "Mary" to be replaced by "Henry", Partee's rule would do no more than preserve the status quo. A pragmatic, albeit ad hoc, solution would be to ascribe "common" gender to *all* disjoined term phrases irrespective of the genders of the disjuncts; for in this way anaphoric references of the form "the one in question" would be invoked in every case.

There can be no doubt that SA3 and SA14 (ii) do indeed eliminate Janssen's "not there" examples of vacuous relativisation and vacuous quantification: nevertheless Partee herself expresses dissatisfaction with her own formulations. In neither case does the factorisation demanded by "$\phi = {}_t[\beta\ he_n\ \gamma]$" or "$\phi = {}_t[\gamma\ he_n\ \xi]$" guarantee that β and γ or γ and ξ are well formed constituents, but whereas this consideration is no more than an irritation in the case of S3, it vitiates the definition of $f_{10,n}$ as employed by S14.

The structural operation $f_{10,n}$ involves the primitive operations "concat" and "esub", the latter called by the sub function "prosub"; but the primitive operations are well defined only for constituents.[†50] Thus:

$$concat(3,t,\langle\gamma,\alpha,esub(proform(\alpha),he_n,\xi\rangle)$$

is strictly speaking undefined. One tentative reformulation of $f_{10,n}$ in terms of well formed constituents, viz:

$$f_{10,n}(\alpha,\phi) = esub(proform(\alpha),he_n,sub(\alpha,he_n,\phi))$$

in which the third argument to "esub" is the entire structure after the replacement of the first indexed variable by α, is rejected as it would fail should α itself contain occurrences of he_n.

The root causes of the problem are Partee's preoccupation with the convention that a structure to be transformed must be specified, as in SA3 and SA14 (ii), as a partition into fixed finite factors together with her suspicion of *explicit* references to derivational history.

That the factoring convention is singularly inappropriate is apparent in her discussion of "everywhere substitution"; for her decision to introduce "esub" as an informally defined primitive stems from her inability to express in terms of finite factors the condition that an input structure may contain an arbitrary number of variables. No representation of the form:

$${}_t[\alpha\ he_n\ \beta\ he_n \ldots \chi\ he_n\ \psi\ he_n]$$

could in principle suffice to this end. Moreover the comparison of her relational properties with Friedman's functions confirms that her avoidance of explicit references to derivational history has been

†50. Partee's definitions do not in fact make the stipulation that the arguments must be constituents rather than strings, however her intention is deducible from her dissatisfaction.

****

```
/* Substitute A for 1st. occurrence of B in X leaving Y else fail */
/* if no such occurrence.                                         */

sub(A,B,X,Y) :- X=..[P,X1], sub(A,B,X1,Y1), Y=..[P,Y1].
sub(A,B,[B|T],[A|T]) :- !.
sub(A,B,[H|T],[H1|T]) :- sub(A,B,H,H1).
sub(A,B,[H|T],[H|T1]) :- sub(A,B,T,T1).

/* Substitute A for every occurrence of B in X leaving Y. */

esub(A,B,X,Y) :- X=..[P,X1], esub(A,B,X1,Y1), Y=..[P,Y1].
esub(A,B,[B|T],[A|T1]) :- esub(A,B,T,T1).
esub(A,B,[H|T],[H1|T1]) :- esub(A,B,H,H1), esub(A,B,T,T1).
esub(A,B,X,X).

/* Substitute A for 1st. occurrence of B in X and C for subsequent */
/* occurrences of B in X leaving Y else fail if no occurrences. */

qsub(A,C,B,X,Y) :-
        sub(np,B,X,X1),
        esub(C,B,X1,X2),
        sub(A,np,X2,Y).
```

Fig 26

****

achieved only at the expense of equivalent *implicit* references. If we can always simulate explicit references by encoding relevant information in features, why should we baulk at making the references explicit? Partee does indeed wish to constrain the use of features so as to make the implicit references to derivational history less arbitrary, but could not such constraints equally well be formulated for *explicit* references?

Just as the form of structural operations is effectively constrained by In4, which is unashamedly designed to make such operations simulate computer programs, so to references to derivational history would be effectively constrained were they required to be computable. If, as suggested above, labeled bracketing is represented in the form of a data structure recognizable to an actual programming language, then legitimate references could be limited to implementable procedures in that language.

With this in mind I suggest the PROLOG redefinitions of "sub" and "esub" appearing in fig 26, which presuppose data structures of the form described in §3.4.2 (e).

In terms of "sub" and "esub" we may define a further procedure "qsub" as illustrated, and then, given that we accept SA14 (i) as a valid prohibition, we may adopt a hybrid pseudo code incorporating both PROLOG style predicates and conventional functions to specify:

(S14) If $\alpha \in P_T$ and $\phi \in P_t$ and SA14 is fulfilled then $f_{10,n}(\alpha,\phi) \in P_t$.

SA14 (i): pro(β) = H- for all β such that mainterm(β,α) = MT+.

SA14 (ii): $f_{10,n}(\alpha,\phi)$ is defined for ϕ.

$f_{10,n}(\alpha,\phi) = \psi$ such that qsub(α,proform(α),he_n,ϕ,ψ).

Since "sub", and consequently "qsub", both fail in the absence of at least one occurrence of B in X, $f_{10,n}$ continues to be a partial function as allowed by In5, and vacuous applications of S14 continue to be eliminated: we merely require SA14 (ii) to stipulate that the structural operation must succeed.

The relative clause rule may be revised in like manner to read:

(S3) If $\alpha \in P_{CN}$ and $\phi \in P_t$ and SA3 is fulfilled then $f_{3,n}(\alpha,\phi) \in P_{CN}$.

SA3: $f_{3,n}(\alpha,\phi)$ is defined for ϕ.

$f_{3,n}(\alpha,\phi) = \psi$ such that concat(4,CN,<α,such,that,qsub(proform(α),proform(α),he_n,ϕ,ψ).

PS3: As before.

Note that "qsub" only works as defined on the assumption that terms are to be maintained as root forms plus feature bundles in accordance with Partee's concession C3 (iii). All variables to be replaced are identifiable as root forms he_n irrespective of their case, and a convention that non conflicting features are retained guarantees that, although proform(α) yields a pronominal form marked only for gender and number, the case feature is inherited upon replacement of a variable.

If we prefer to abandon the tactic of holding expressions as root forms plus feature bundles, and to reintroduce inflected forms to analysis trees, it becomes necessary to define an alternative to "sub" which replaces *any* first occurrence of a variable with index *n* whatever its case, and to redefine "esub" more specifically so that it constructs appropriate pronominal replacements in situ from the information available at the time. Such loss of generality is hardly critical since in Partee's fragment "esub" has no use other than to accomplish pronominal replacement of variables.

A prerequisite is that we be able to identify syntactic variables of a given index irrespective of their inflection, and to construct pronominal forms with amalgamated features. In §3.4.2 it was suggested that an advantage of the structured list convention in the representation of expressions was that additional

****

```
/* Arg1 is a syntactic variable with index N */

synvar(term([V],[Case]),N) :-
                        name(N,Suffix),
                        varstem(Case,Stem),
                        append(Stem,Suffix,Ascii),
                        name(V,Ascii).

/* Arg1 is a syntactic variable with index N, Arg3 is a term, and Arg4 is a       */
/* pronoun marked with the case of Arg1 and the number and gender of Arg3 */

pform(term([V],[Case]),N,term([A],[Gen,Num,_]),term([P],[Gen,Num,Case])) :-
                        synvar(term([V],[Case]),N),
                        pronoun(P,[Gen,Num,Case]).
```

Fig 27

****

information could be incorporated without corrupting the basic constituents. Suppose therefore that all expressions be held in the form:

category([Constituents],[Features]).

It then becomes trivially easy to accommodate the aforementioned prerequisites by defining the PROLOG style procedures "synvar" and "pform" illustrated in fig 27. The only assumption made is that legitimate variable stems and feature marked pronouns be somewhere recorded, a prologue to the lexicon being the obvious place.

With the prerequisites complete, a new procedure "psub" and revised versions of "esub" and "qsub"†51 may be defined as in fig 28, whereafter $f_{10,n}$ and $f_{3,n}$ may be respecified as follows:

$f_{10,n}(\alpha,\phi) = \psi$ such that qsub(α,n,ϕ,ψ).

$f_{3,n}(\alpha,\phi) = $ cn([α such that ψ],[G,N,_]) given
esub(α,n,ϕ,ψ) and $\alpha = $ cn([β],[G,N,_]).

The hybrid nature of the pseudo code has coincidentally been alleviated so that the formulations now approximate to PROLOG, but the new definition of $f_{3,n}$ does have the disadvantage that it necessitates resurrection of SA3 to eliminate vacuous applications.

---

†51. The procedure "lform" is merely a local version of "unif" which does *not* accept a first argument already in list form. Whereas in CPROLOG [A,B]=..X will return X as [.,A,B], lform([A,B],X) will fail.

****

```
/* Substitute A for 1st occurrence of B in Arg3 leaving Arg4 else fail */
/* if no such occurrence */

sub(A,B,[B|T],[A|T]) :- !.
sub(A,B,[H|T],[H1|T]) :- sub(A,B,H,H1).
sub(A,B,[H|T],[H|T1]) :- sub(A,B,T,T1).
sub(A,B,X,Y) :- lform(X,[P,X1,F]), sub(A,B,X1,Y1), lform(Y,[P,Y1,F]).

/* Substitute A for first occurrence of a variable with index N in Arg3 */
/* leaving Arg4 else fail if no such occurrence                         */

psub(A,N,[V|T],[A|T]) :- synvar(V,N),!.
psub(A,N,[H|T],[H1|T]) :- psub(A,N,H,H1).
psub(A,N,[H|T],[H|T1]) :- psub(A,N,T,T1).
psub(A,N,X,Y) :- lform(X,[P,X1,F]),psub(A,N,X1,Y1),lform(Y,[P,Y1,F]).

/* Substitute a pronoun with number and gender of A and case of variable for */
/* each variable in Arg3 with index N leaving Arg4 */

esub(A,N,[V|T],[P|T1]) :- pform(V,N,A,P),esub(A,N,T,T1).
esub(A,N,[H|T],[H1|T1]) :- esub(A,N,H,H1),esub(A,N,T,T1).
esub(A,N,X,Y) :- lform(X,[P,X1,F]),esub(A,N,X1,Y1),lform(Y,[P,Y1,F]).
esub(A,N,X,X).

/* Substitute A for first occurrence of variable with index N in X and */
/* a suitable pronominal form for subsequent occurrences leaving Y     */
/* else fail if no occurrences.                                         */

qsub(A,N,X,Y) :-
                    psub(np,N,X,X1),
                    esub(A,N,X1,X2),
                    sub(A,np,X2,Y).
```

Fig 28

****

I do not propose that this somewhat arcane structured list notation be always adopted in the specification of rules for a Montague grammar. It should be sufficient that the rigorous option be available should defence of a formulation be demanded. If $f_k(\zeta,\xi) \in P_{cat}$ then the following may be seen as alternatives:

(a) $f_k(\zeta,\xi) = \theta$ where θ has features F and θ *could* be expressed as a composition of sub functions.

(b) $f_k(\zeta,\xi) = {}_{cat}[\theta]$ where θ has features F and θ *is* a composition of subfunctions.

(c) $f_k(\zeta,\xi) =$ cat([θ],[F]) where θ is defined in terms of PROLOG predicates.

The original PTQ version (a) may indeed be conceived as a shorthand for either Partee's preference (b) or the structured list notation (c); moreover if we adopt the convention that only the field θ be printed on analysis trees then the visible output from all three will be identical. Accordingly I shall from time to time employ whichever alternative seems most appropriate for the occasion. In deference to its ontogenesis I shall refer to a syntax rule S*n* having PROLOG style definitions of all strings referenced either in structural analysis condition SA*n* or in the specification of structural operation f_k as being in PNF[†52]

3.5. Janssen's Hyperrules

Vacuous relativisation and vacuous quantification are regarded as offensive by Janssen, [J3, J4], not only on empirical grounds, but also because they constitute a breach of what he terms the variable principle. This is in reality a set of related principles which may be abbreviated as follows:[†53]

(VP-1) S*n* may introduce a syntactic variable iff T*n* introduces a related logical variable.

(VP-2) S*n* may remove all occurrences of a syntactic variable iff T*n* binds all occurrences of the corresponding logical variable.

(VP-3) If S*n* purports to remove occurrences of a syntactic variable it must actually do so.

(VP-4) If a sequence of rule applications generates a string with extant syntactic variables the sequence is incomplete.

Of these principles, VP-3 alone proscribes vacuous rule applications, while VP-4 rejects sentences containing "left overs", ie. members of P_t appearing in top nodal positions with surviving syntactic variables. The effect of the first two principles is to inhibit attempts to accommodate the other two by dubious methods such as erasing the subscripts from left over variables and pretending that they are indexical.

Partial functions are needed to eliminate vacuous rule applications so long as the input categories for which a rule is defined may contain inappropriate instances. A system of categories which permitted finer discriminations might serve to isolate only the appropriate instances thus allowing total functions to be

---

†52. On the intended reading of the mnemonic, PNF = Partee normal form, but should she prefer to disassociate herself then PNF = PROLOG normal form.

My contention then is that all rules should be *expressible* in PNF even if, for reasons of readability and where no controversy arises, they are sometimes less formally expressed.

†53. VP-1 corresponds to Janssen's 1(a) and 1(b), VP-2 to 2(a) and 2(b), VP-3 to 3(b) and VP-4 to 3(a). Although Janssen's principles come in couples, the (a) and (b) clauses are converses only in the case of the first two pairs.

defined thereon. Accordingly an ingenious method for introducing such categories, and for modifying the grammar rules so as to accommodate them, is devised by Janssen, [J3], who wishes to eliminate "not there" cases whilst not appealing to partial functions which he distrusts.

Janssen observes that the grammar of PTQ already includes some rule schemata containing *meta variables*. The relative clause and quantification rules are such schemata which resolve into actual rules only when the subscript *n* in references to the structural operation and to "he_n" or "him_n" is replaced by an integer. Such rules are strictly speaking hyperrules which should be augmented by meta rules determining the valid substitution instances for the meta variables. For example, since the subscript A in the definition of S1 is a meta variable, S1 is a hyperrule which requires augmentation by the meta rule:

A → IV | CN | T | IAV | TV | SCVERB | ICVERB | PREP | SADV

In order to eliminate "not there" cases, Janssen introduces compound category symbols of the form"(CAT,BAG)", where "CAT" is the name of a syntactic category and "BAG" identifies a bag or multiset of unremoved indices. If $\phi \in P_{(t,\{1,2,2,3\})}$ then $\phi \in P_t$ and ϕ contains the syntactic variables with indices 1, 2 (twice), and 3. Plainly we may now redefine *Esen* as follows:

$Esen = P_{(t,\{\})}$.

The meta rules for defining legitimate substitution instances for bag variables are:

BAG → {SEQ}

SEQ → NUM | NUM,SEQ | λ

NUM → 0 | NZ

NZ → 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | NZ0 | NZNZ

Additionally, bags are subject to the certain operations which may be defined as follows:†54

Bag Union: {SEQ} ∪ {SEQ′} = {SEQ, SEQ′}.

Bag Difference: {SEQ} - NUM = {SEQ} iff {SEQ} ≠ {SEQ′} ∪ {NUM} where {SEQ′} ⊂ {SEQ} otherwise {SEQ} - NUM = {SEQ′} - NUM.

†54. The definitions given here are my own formulations not those of Janssen.

Bag Specification: {SEQ} *with* NUM = {SEQ} iff ({SEQ} - NUM) $\subset$ {SEQ} otherwise {SEQ} *with* NUM = {SEQ} $\cup$ {NUM}.

The relative clause rule and sentence quantification rule can now be reformulated as hyperrules ie. rule schemata to be instantiated in accordance with the above meta rules.[†55] The conditions of application may be completely stated in categorial terms thus:

(S3) If $\alpha \in P_{(CN,BAG)}$ and $\phi \in P_{(t,BAG'\ with\ n)}$ then $f_{3,n}(\alpha,\phi) \in P_{(CN,BAG \cup (BAG' - n))}$.

(S14) If $\alpha \in P_{(T,BAG)}$ and $\phi \in P_{(t,BAG'\ with\ n)}$ then $f_{10,n}(\alpha,\phi) \in P_{(t,BAG \cup (BAG' - n))}$.

The structural operations are not affected by these modifications, but, as required by Janssen, they now become total functions.

3.6. Formulaic and Processing Parsimony

To every syntactic rule S*n* invoking a structural operation f_k there corresponds a translation rule T*n* licensing a semantic operation g_j. In PTQ the sets of structural operations and semantic operations are functionally dependent upon the sets of syntax rules and translation rules respectively.

The only advantage gained by the many:1 relationship between syntax rules and structural operations is parsimony in *formulation*: the effect of a structural operation need be defined only on the first occurrence. Were we to sacrifice such parsimony we could adopt the alternative convention of letting each syntactic rule S*n* invoke a structural operation f_n with individual, albeit non unique, PNF formulations of the effects provided on a rule by rule basis. Indeed we could then abandon altogether the divers two placed structural operations by introducing instead a single three placed operation such that:

$$f(n,\alpha,\beta) = f_n(\alpha,\beta).$$

I have already expressed dissatisfaction with the practice of labelling analysis tree nodes with operation indices alone since, given the many:1 convention, these indices do not serve to determine the syntactic rule applied without covert reference to the daughter categories. My provisional solution hitherto has been to include in each nodal label a *pair* of indices *<n:m>* where *n* is the rule number and *m* the operation

†55. Janssen also introduces hyperrules in revised versions of S4, S8, S10 and S17, but the issues raised have no bearing on the present chapter.

****

Semantic Op.	Definition	Invoking Rule	PTQ Structural Op.
$g_{every}(\zeta')$	$\lambda p\forall X(\zeta'(X)\rightarrow{}^{\vee}p(X))$	S2	f_0
$g_{the}(\zeta')$	$\lambda p\exists Y(\forall X(\zeta'(X)\leftrightarrow X=Y)\wedge{}^{\vee}(Y))$	S2	f_1
$g_a(\zeta')$	$\lambda p\exists X(\zeta'(X)\wedge{}^{\vee}p(X))$	S2	f_2
$g_0(\theta',\eta')$	$\theta'({}^{\wedge}\eta')$	S4,S5,S6,S7,S8,S9,S10	f_4,f_5,f_6,f_7
$g_{2,n}(\alpha',\phi')$	$\alpha'({}^{\wedge}\lambda x_n\phi')$	S14	f_{10}
$g_{3,n}(\zeta',\phi')$	$\lambda x_n(\zeta'(x_n)\wedge\phi')$	S3	f_3
$g_{4,n}(\alpha',\theta')$	$\lambda Y\alpha'({}^{\wedge}\lambda x_n[\theta'(Y)])$	S15,S16	f_{10}
$g_5(\gamma',\delta')$	$\lambda X(\gamma'(X)\wedge\delta'(X))$	S12	f_8
$g_6(\gamma',\delta')$	$\lambda X(\gamma'(X)\vee\delta'(X))$	S12	f_9
$g_{11}(\phi',\psi')$	$(\phi'\wedge\psi')$	S11	f_8
$g_{12}(\phi',\psi')$	$(\phi'\vee\psi')$	S11	f_9
$g_{14}(\alpha',\beta')$	$\lambda q(\alpha'(q)\vee\beta'(q))$	S13	f_9
$g_{not}(\alpha',\delta')$	$\neg\alpha'({}^{\wedge}\delta')$	S17	f_{11}
$g_{will}(\alpha',\delta')$	$fut\alpha'({}^{\wedge}\delta')$	S17	f_{12}
$g_{wont}(\alpha',\delta')$	$\neg fut\alpha'({}^{\wedge}\delta')$	S17	f_{13}
$g_{has}(\alpha',\delta')$	$past\alpha'({}^{\wedge}\delta')$	S17	f_{14}
$g_{hasnt}(\alpha',\delta')$	$\neg past\alpha'({}^{\wedge}\delta')$	S17	f_{15}

Fig 29

****

number (augmented in need by a variable index). Adoption of the convention whereby syntax rule and structural operation bore the same index would introduce overt redundancy into this provisional solution, but even without such a change the value of retaining the operation number in the node label may be questioned. Like the syntactic rule number, the operation number supplies *historical* information: it tells us *after the event* how a nodal phrase was derived, information which we could in need recover at our leisure by inspecting the rules.

What the present node labelling system does *not* provide is any direct *prognostic* guidance: we are offered no prescription regarding the *semantic* operation to be applied when post ordering the tree. We may of course determine this information *indirectly* by inspecting the appropriate translation rule, but in processing terms such an indirect reference proves unnecessarily inefficient.How much better to record the prescription on the analysis tree in the first place.

Parsimony in processing as opposed to mere formulation would be achievable were the semantic operations to be indexed, and were such indices to be paired with the syntax rule numbers in node labels. Ironically, in PTQ it is structural operations which are specifically indexed while semantic operations are not, although it now transpires that the latter indices would have the greater utility. Accordingly I suggest that the most auspicious node label must have the form:

<l,<n:j>>

where ***l*** is the nodal phrase, ***n*** the index of the syntactic rule S*n* and ***j*** the semantic operation invoked by the translation rule T*n*. As previously ***j*** may be augmented by a syntactic variable number if S*n* has the nature of a hyperrule.

A total of seventeen semantic operations feature in PTQ, and these may for convenience be indexed as in fig 29. Of these operations g_{every}, g_{the} and g_a will become obsolete once a binary version of S2 has been devised, while g_{not}, g_{will}, g_{wont}, g_{has} and g_{hasnt} will not survive in a more adequate treatment of tense and aspect. The significance of the present discontinuous numbering will become more apparent once the complete rule set for TMG has been developed; meanwhile I shall from now on adopt both the new labelling system and the convention that S*n* invokes f_n. By replacing the structural operation numbers as listed in fig 29 with the corresponding semantic operation indices previous examples may with ease be converted to the new format.

¶ Certain of the many suggested extensions to Montague grammar seem more fundamental than others and to merit inclusion in the basic grammar which TMDCG is to simulate. In this chapter we consider Rodman's reformulated relative clause rules, and the treatment of indirect interrogatives proposed by Karttunen and Peters. All rules discussed are redrafted in PNF preparatory to inclusion in TMG.

4.1. Extending the Coverage of Montague's Fragment.

Attempts to extend the range of linguistic phenomena covered by Montague grammar have been legion. The phenomena to be accommodated have included dative movement, [C8, D7], subject raising, [P4], object raising, [C8, P4], double object verbs, [B8, C8], tough movement, [P4], tense and aspect [D6, D8, T4], passivisation, [B1, B2, P4], non-stilted relativisation, [R3], and the introduction of interrogatives, [B9, B11, G8, H2, K4, K5]. Although none of these endeavours is without interest, two seem to me successfully to remedy more fundamental deficiencies than the remainder. These two, namely Rodman's revised treatment of relative clauses and the introduction of indirect interrogatives by Karttunen and Peters, will accordingly be considered in detail. During the course of discussion the rules provided will be massaged into the form (PNF) in which they will eventually appear in TMG, the target grammar implemented in TMDCG. The accommodation of tense, aspect and passivisation is of course equally fundamental, but the suggestions to date less satisfactory, thus these issues will be addressed in a separate chapter.

4.2. Rodman's Relative Clause Rules

Failure to eliminate "not there" cases is not, according to Rodman, [R4], the only unsatisfactory feature of the relative clause syntax in PTQ.†56 That it generates a stilted and archaic form of expression is itself grounds for criticism; but more significantly S3 permits constructions which infringe some important linguistic constraints. These constraints are of two kinds which Karttunen, [K4], identifies as "constraints on replacement" and "constraints on extraction". The former involve the scope of quantification in relative

†56. The semantics is unexceptionable and remains constant throughout Rodman's reformulations, ie. $g_{3,n}$ is the appropriate semantic operation.

clauses whilst the latter are equivalent to the "island constraints" of Ross, [R8]. Rodman suggests a uniform mechanism for accommodating constraints of both kinds, and incorporates this in rules to replace the original S3.

4.2.1. Rodman's Constraint on Quantifier Scope

There does not appear to be any plausible interpretation for the sentence:

(83) John dates a woman who loves every man.

in which the phrase "every man" has wider scope than "a woman". This consideration prompts Rodman to postulate the principle:

> "In a relative clause the element that is relativised always has wider scope than any other element in that relative clause."

a principle which he calls the ***constraint upon quantifier scope***", and which, he suggests, must govern any adequate relative clause rule.

A revised rule S3,*n* would, in an ideal Montague grammar, generate a common noun phrase "woman who loves him_m" from the common noun "woman" and the disambiguated language sentence "he_n loves him_m by prefixing the common noun together with a pronoun marked with the case of the first variable having index *n* (which in the example happens to be *first and nominative*), and thereafter deleting that variable.

Given no additional restrictions, the sentence "John dates a woman who loves every man" with "every man" illicitly in wide scope could then be generated by substituting the term "every man" for the *first and accusative* variable "him_m" in the relative clause within the disambiguated sentence "John dates a woman who loves him_m" as typified in fig 30. To generate an error rule S14 must be applied after S3 has operated, and the variable to be substituted, ie. the *first* occurrence of the variable in question, must fall within the expression formed by S3.

Rodman refers to the substitution of a term for a *first and nominative* or *first and accusative* variable as *binding by quantification*: the replacement of *subsequent* variables by surface anaphora he calls *binding by pronominalisation*. Thus the implication for Montague grammar is that once S3 has been applied none

****

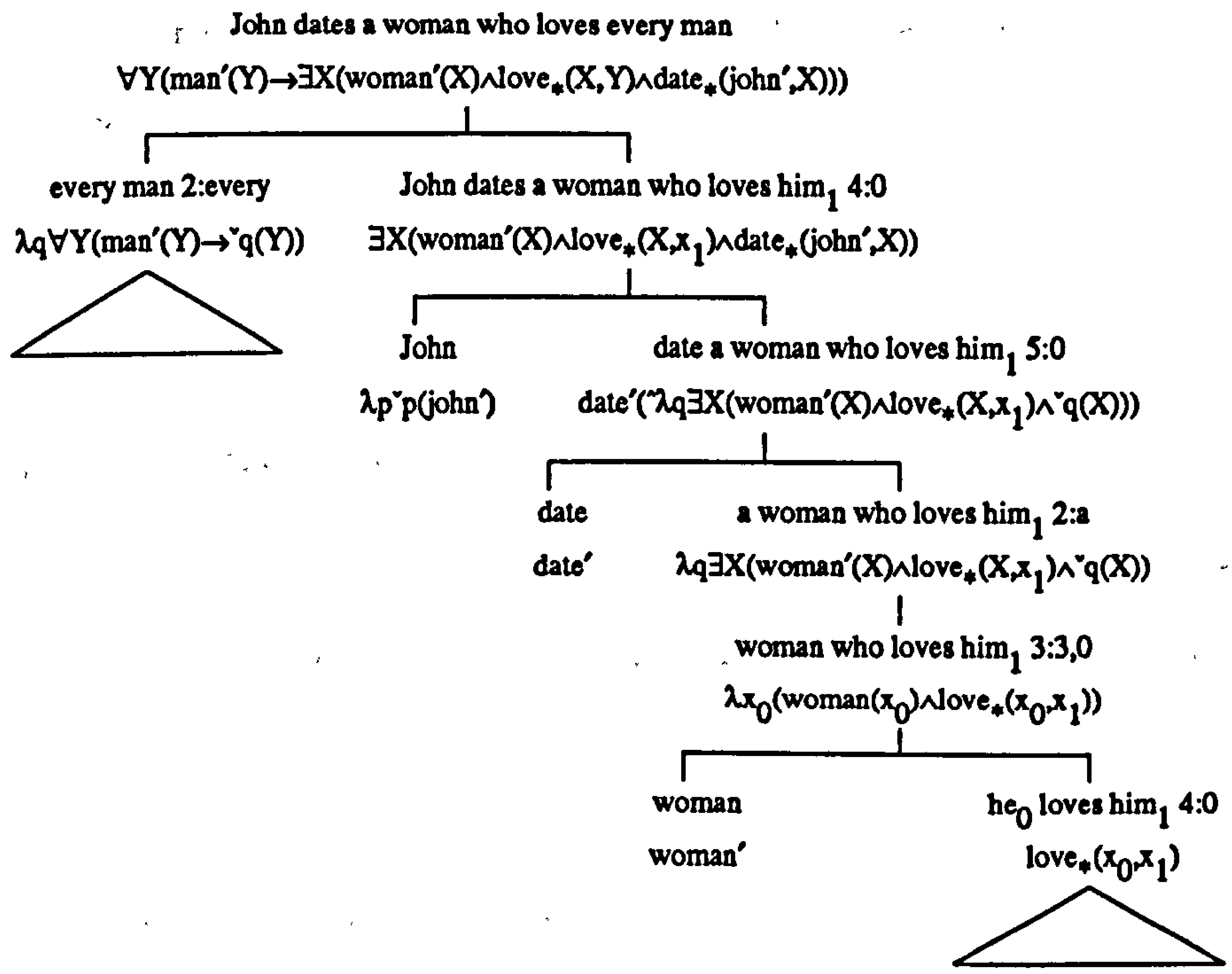


Fig 30

****

of the quantification rules (S14, S15, S16) should be permitted subsequently to *bind by quantification* a variable within the relative clause so formed. There is no need, according to Rodman, similarly to restrict *binding by pronominalisation*; for there can be no objection to quantifying the noun phrase "a cow" into the expression:

(84a) Bill owns him_1 and John has dated a woman who has milked him_1.

in order to generate:

(84b) Bill owns a cow and John has dated a woman who has milked it.

where the variable bound by quantification is outside the relative clause, while that within it is merely pronominalised.

Unfortunately Rodman's contention is far too strong;[†57] for as we have seen, (§2.3.2 and footnote

---

†57. Karttunen, [K4] has also argued that the quantifier scope constraint should be treated as weaker than the island constraints since there appears to be a reading of

"John wants to date every girl who goes out with a professor who failed him in Linguistics"

wherein the existential quantifier has wider scope than "every".

32), any construction having a pronoun in the main clause coreferential with a noun phrase within a relative clause requires that noun phrase to have wide scope. Not only do problematic "donkey sentences" have this characteristic, so too do innocuous examples like:

(85) The man who loved Mary kissed her.

which must plainly be encompassed by the grammar.

4.2.2. The Ross "Island" Constraints

Superficially, the effect of relativisation is to remove a noun phrase from within a sentence and to reposition it together with a relative pronoun marked with the original case to the extreme left thus creating a new noun phrase, ie. the original noun phrase is subjected to left extraposition. From " a woman loves a man" we may expect to derive "a man whom a woman loves". Formulated in terms of surface appearances there is a ***constraint on relativisation*** which states that a noun phrase already embedded within a relative clause cannot itself be extraposed and relativised. In transformational terms, this restriction is tantamount to the first of the Ross "island" constraints, ie. the ***complex noun phrase constraint***:

> "No element contained in a sentence dominated by a noun phrase with a lexical head noun may be moved out of that noun phrase by a transformation."

Infringement of the constraint would allow us to generate not only:

(86) A man whom a woman loves walks.

and:

(87) John has dated a woman who loves a man.

but also:

(88) * A man whom John has dated a woman who loves walks.

The last sentence is clearly unacceptable although curiously the stilted formulations of PTQ disguise the problem; for we may in fact accept:

(89) A man such that John has dated a woman such that she loves him walks.

In the non-stilted form, relative clauses form "islands" from which constituents may not be extracted.

A second island constraint proposed by Ross, is the ***coordinate structure constraint***:

"In a coordinate structure, no conjunct may be moved, nor may any element contained in a conjunct be moved out of that conjunct."

This constraint inhibits the generation of sentences like:

(90) * John loves the woman who and the dog walk in the park.

for without it a noun phrase within the conjoined subject of the sentence "the woman and the dog walk in the park." could be extraposed and relativised.

The analysis in terms of surface structure is potentially misleading in that it suggests the derivation of a relative clause from a *noun phrase* and a sentence from which that phrase has been extracted. As Partee has pointed out, [P3, P4], the semantic consequences of such a derivation would be that, where the noun phrase involved the definite article, unique individuation (albeit contextually assisted) of the referent must be possible *before* attachment of the relative clause which would then amount to an afterthought.[†58] For this reason Montague's rule S3 combines a sentence not with a term but with a common noun, thus deferring any claim to unique individuation until the *result* is combined with a determiner. Closer investigation thus reveals that at a more fundamental level the expression removed from a sentence by relativisation must in fact be a syntactic variable, and that the preposed expression must be not a noun phrase but a common noun. Evidently the *binding by relativisation* of a variable involves its deletion, therefore the Ross constraints entail that no subsequent application of a revised rule S3 should be allowed to *bind by relativisation* a variable contained in an expression created by a previous application of S3, or by a previous application of S11.

4.2.3. Rodman Variables

Rodman's technique for accommodating all the above constraints is to flag those variables surviving after an application of S3 or S11 with a superscript *R*, ie. the variables are "rodmanised". The only variables that may be *bound* either by relativisation using S3 or by quantification using S14, S15, or S16 are those which have *not* been rodmanised. The technique is plainly similar to that employed by Bennett to

†58. Such an analysis, as Partee observes, would be acceptable for *unrestrictive* relative clauses.

handle the reflexive phenomena, but whereas Bennett's variables commence with flags which they *lose* when no longer subject to reflexivisation, Rodman's variables *gain* flags when not available for binding. Thus even if we reject Rodman's ***constraint upon quantifier scope*** his reformulations remain of interest not only in virtue of their eliminating Montague's stilted syntax, but also because rodmanised variables provide a means for accommodating reflexivisation, albeit remaining bindable.

Rodman's revised relative clause rules are designed to allow either the neutral relative pronoun "that" or the case and gender sensitive "wh-" types "who", "whom", and "which". Reduced relative clauses with no relative pronoun are also permitted when the relativised term is not nominative. The rules are four in number:

(S3) If $\alpha \in P_{CN}$ and $\phi \in P_t$ then $f_{3,n}(\alpha,\phi) \in P_{CN}$.

$f_{3,n}(\alpha,\phi) = \alpha$ that ϕ' where ϕ' comes from ϕ by replacing the first occurrence of he_n or him_n with $wh\text{-}he_n$ or $wh\text{-}him_n$ and (ii) replacing all further occurrences of he_n, he_n^R, him_n or him_n^R by a surface pronoun of like case and matching the gender of α, and (iii) flagging all occurrences of he_m and him_m, $m \neq n$, with superscript R.

(S3D)(Relative Pronoun Deletion)

If $\alpha \in P_{CN}$ then $f_{3D}(\alpha) \in P_{CN}$.

$f_{3D}(\alpha) = \alpha'$ where α' comes from α by deleting $wh\text{-}he_n$ or $wh\text{-}him_n$ *provided that it is not preceded by a member of* P_{CN}.

(S3P) (Wh- Preposing)

If $\alpha \in P_{CN}$ then $f_{3P} \in P_{CN}$.

$f_{3P}(\alpha) = \alpha'$ where α' comes from α by replacing the first "that" with the first $wh\text{-}he_n$ or $wh\text{-}him_n$ and then changing that occurrence to either the appropriate case of "who" or "whom" or to the neuter "which" depending on the gender of the head nominal.

(S3R) ("that" Reduction)

If $\alpha \in P_{CN}$ then $f_{3R}(\alpha) \in P_{CN}$.

$f_{3R}(\alpha) = \alpha'$ where α' comes from α by deleting the first occurrence of "that" *provided that it is followed immediately by a member of* P_T

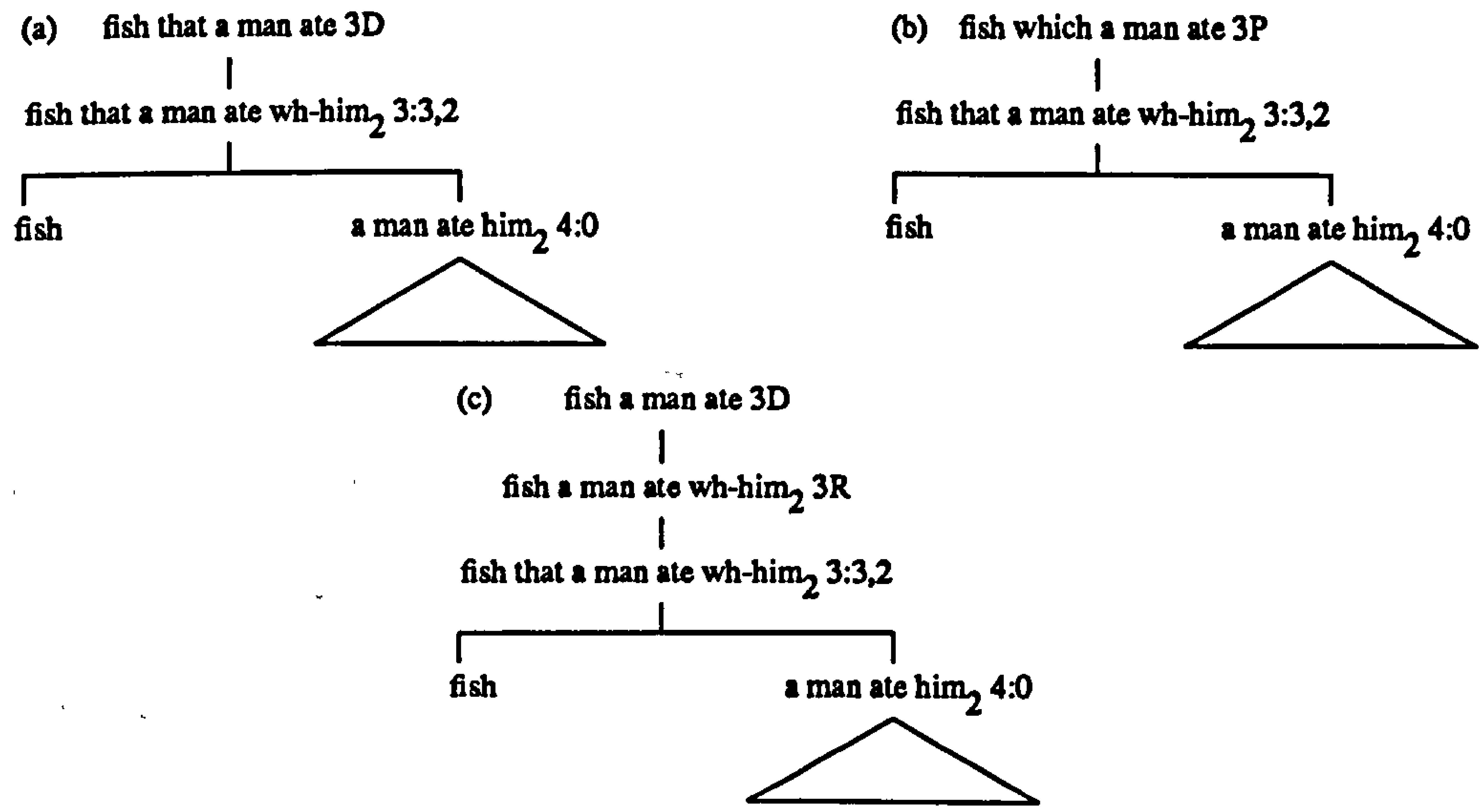


Fig 31

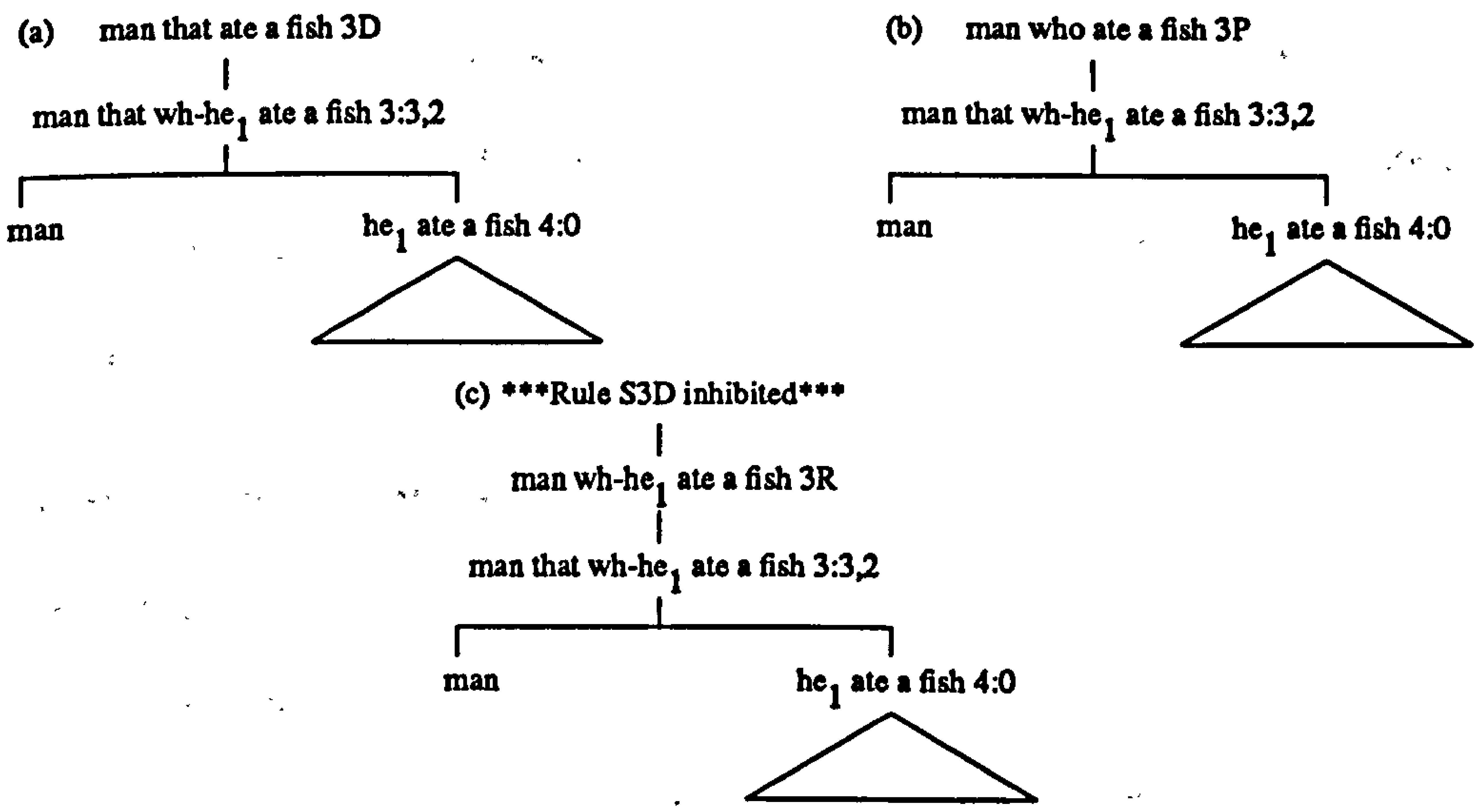


Fig 32

If we commence with the inputs “fish” and “a man ate him_2”, then by applying S3 we may generate:

(91) fish that a man ate wh-him_2.

whereafter we may apply either S3D to achieve:

(92) fish that a man ate.

or S3P to derive:

(93) fish which a man ate.

or finally S3R followed by S3D (or vice versa) to obtain:

(94) fish a man ate.

These possibilities are illustrated in fig 31.

If however we start with "man" and "he_1 ate a fish" we first generate by S3:

(95) man that wh-he_1 ate a fish.

from which, as illustrated in fig 32, we may either extract by S3D:

(96) man that ate a fish.

or alternatively derive by S3P:

(97) man who ate a fish.

The combination S3R followed by S3D will however now fail because after deleting "that" the variable wh-he_1 will be preceded by a member of P_{CN}. Nor can we apply S3D followed by S3R, for on deleting wh-he_1 in (96) the surviving "that" is not followed immediately by a term; hence we cannot generate as a putative member of P_{CN}:

(98) * man ate a fish.

4.2.4. Conservative Reformulations of Rodman's Rules

As formulated, Rodman's suggestions are vulnerable to the accusation that they infringe Partee's constraints on valid rules. Although S3R is optional and need not be applied even when its condition is fulfilled, an application of S3 *must* be followed by an application of either S3D or S3P. The latter are jointly obligatory because Rodman's version of S3 is in breach of the well formedness constraint C3. Moreover both S3D and S3R make explicit appeal to derivational history in defiance of C5 since their applicability depends upon the relative locations of the elements to be deleted.

The fundamentals of Rodman's solution may however be preserved in rules reformulated in such a way that Partee's constraints are not so flagrantly violated. Obligatory rules may be avoided by amalgamating S3 with S3D to obtain a new rule S30 which deals with relative clauses employing "that", and similarly amalgamating S3 with S3P to produce a new rule S31 to handle relative clauses involving "wh" relative pronouns. Although recourse to derivational history is not completely eliminated in the formulations which follow, references to constituent structure are limited to those formulable in terms of computable procedures, a limitation which I have suggested provides a workable compromise.

****

```
/* Arg1 is a rodmanised variable with index N */

rsynvar(term([R],[C]),N) :-
                    name(N,Suffix),
                    rstem(C,Stem),
                    append(Stem,Suffix,Ascii),
                    name(R,Ascii).

/* V is the set of rodmanised variables with index N in Arg2 */

rvariables(N,[term([R],[C])|B],V) :-
                    rsynvar(term([R],[C]),N),!,
                    rvariables(N,B,V1),
                    union([R],V1,V).
rvariables(N,[H|T],V) :-
                    rvariables(N,H,V1),
                    rvariables(N,T,V2),
                    union(V1,V2,V).
rvariables(N,B,V) :- lform(B,[P,B1,F]),rvariables(N,B1,V).
rvariables(N,B,[]).

/* Arg3 results from rodmanising variable Arg1 with index N */

rform(term([V],[C]),N,term([R],[C])) :-
                    name(V,W),
                    freestem(C,Stem),
                    append(Stem,Suffix,W),
                    name(N,Suffix),integer(N),
                    append(Stem,[82|Suffix],W1),
                    name(R,W1).

/* Arg2 is the result of rodmanising all variables in Arg1 */

rsub([V|T],[R|T1]) :- rform(V,_,R),!,rsub(T,T1).
rsub([H|T],[H1|T1]) :- rsub(H,H1),rsub(T,T1).
rsub(X,Y) :- lform(X,[P,X1,F]),rsub(X1,Y1),lform(Y,[P,Y1,F]).
rsub(X,X).
```

Fig 33

****

The procedures required to handle rodmanised variables are "rsynvar" which identifies a single such variable, "rvariables" which identifies the set of Rodman variables in an expression, "rform" which pro-

```
*****************************************************************************************

        /* D1 and D2 are the binary daughters of M */

        daughters(M,D1,D2) :- lform(M,[P,[D1,D2],F]).

        /* W is the first word in Arg1 */

        firstword([H|T],W) :- firstword(H,W).
        firstword(B,W) :- lform(B,[P,B1,F]),firstword(B1,W).
        firstword(W,W).

        /* V is leading variable with index N in Arg2 and V has case C */
        /* else fail if none found                                     */

        leadvar(N,[term([V],[C])|B],V,C) :- synvar(term([V],[C]),N),!.
        leadvar(N,[H|T],V,C) :- leadvar(N,H,V,C),!.
        leadvar(N,[H|T],V,C) :- leadvar(N,T,V,C).
        leadvar(N,B,V,C) :- lform(B,[P,B1,F]),leadvar(N,B1,V,C).

        /* Y is the result of deleting the leading variable with index N in X */
        /* else fail if none found                                           */

        delete(N,X,Y) :- leadvar(N,X,V,C),erase(term([V],[C]),X,Y).
        erase(V,[V|T],T) :- !.
        erase(V,[H|T],[H1|T]) :- erase(V,H,H1),!.
        erase(V,[H|T],[H|T1]) :- erase(V,T,T1).
        erase(V,X,Y) :- lform(X,[P,X1,F]),erase(V,X1,Y1),lform(Y,[P,Y1,F]).

        /* Given input X, Y is the result of deleting the leading variable with */
        /* index N, replacing all other N-indexed variables by surface pronouns */
        /* of like case and matching A in number and gender, and rodmanising    */
        /* all other variables                                                  */

        dsub(A,N,X,Y) :-
                          delete(N,X,X1),
                          esub(A,N,X1,X2),
                          rsub(X2,Y).

                                  Fig 34
*****************************************************************************************
```

duces a rodmanised version of a hitherto free variable, and "rsub" which rodmanises all outstanding free variables in its input argument. These four procedures are specified in fig 33.

Were the object of the exercise to produce maximally efficient code rather than to demonstrate that *in principle* Rodman's rules can be formulated in terms of computable procedures without obfuscation modified tactics might be required. For example "rform" ascertains that its first argument is a free variable by removing the suffix of any term which has a variable like stem and considering whether or not it is an integer. Although psychologically plausible, this technique is hardly efficient: plainly it would be preferable to record that the item were a variable as one of its features, but then parallelism with the original rules would be jeopardised. In passing we may observe that "varstem" in the original definition of "synvar" must subsume both "freestem" and "rstem".

Four further procedures, which are defined in fig 34, are needed in the reformulation of Rodman's rules. Of these the import of"daughters" and "firstword" is obvious, while "leadvar" both identifies the leading variable in an expression and remembers its case so that a suitable relative pronoun may in need be retrieved from the lexicon.[†59] "delete" removes the leading variable with index N, and lastly "dsub", after calling "delete", replaces all other variables of index N with surface pronouns properly marked for gender,number and case by recourse to the revised version of "esub", and then rodmanises all outstanding free variables. With these procedures to hand Rodman's rules may be paraphrased thus:

(S30) If $\alpha \in P_{CN}$ and $\phi \in P_t$ and SA30 is fulfilled then $f_{30,n}(\alpha,\phi) \in P_{CN}$.

SA30 leadvar(n,ϕ,V,C), rvariables(n,ϕ,R), and V $\notin$ R.

$f_{30,n}(\alpha,\phi)$ = cn([α that ψ],[G,N,_]) given dsub(α,n,ϕ,ψ) and α = cn([β],[G,N,_]).

(S31) If $\alpha \in P_{CN}$ and $\phi \in P_t$ and SA31 is fulfilled then $f_{31,n}(\alpha,\phi) \in P_{CN}$.

SA31 leadvar(n,ϕ,V,C), rvariables(n,ϕ,R), and V $\notin$ R.

$f_{31,n}(\alpha,\phi)$ = cn([α ω ψ],[G,N,_])

given dsub(α,n,ϕ,ψ) and α = cn([β],[G,N,_]) and relpron(ω,[G,N,C]).

(S3R) If $\alpha \in P_{CN}$ and $\phi \in P_t$ and SA3R is fulfilled then $f_{3R}(\alpha,\phi) \in P_{CN}$.

SA3R leadvar(n,ϕ,V,C), rvariables(n,ϕ,R), firstword(ϕ,W), V $\notin$ R and V $\neq$ W.

$f_{3R}(\alpha,\phi)$ = cn([α ψ],[G,N,_]) given dsub(α,n,ϕ,ψ) and α = cn([β],[G,N,_]).

4.3. The Interrogative Theory of Karttunen & Peters

Divers attempts to incorporate interrogatives within the framework of a Montague grammar have been published, ([B9, B11, [G8], [K4], [K5]), of which that by Karttunen and Peters typifies but one approach. My reason for affording prominence to the latter stems from its amenability to inversion and consequently its suitability for conversion to DCG format. Although in some respects more powerful, the

†59. Although the surface relative pronoun is introduced syncategorematically, a lexical entry having no semantic representation provides a mechanism for achieving the correct feature marking.

rules proposed by Groenendijk and Stokhof defy inversion since on occasion they make reference to an accompanying *wh-reconstruction tree* which can in principle only be constructed bottom up.

The Karttunen & Peters account is embodied in two papers, [K4] and [K5] of which the first is attributable to Karttunen alone. In his initial attempt Karttunen commits himself to the view that, in an indirect question, interrogative noun phrases must always have wider scope than ordinary noun phrases. This limitation is removed in the second paper, however whereas the first attempt introduces rules formulated in orthodox terms, the rules of the second are expressed in "Cooper syntax". My exposition will accordingly be eclectic: I shall endeavour to present the corrected thesis formulated in terms of a conventional Montague grammar.

The priority of indirect questions is vouchsafed by the fact that a direct question may always be formulated as the subsumption of an indirect question I.Q. under a performative ie:

(99) I ask you to tell me <I.Q.>.

thus Karttunen commences with a classification of indirect questions. In so doing he introduces an eminently sensible terminology which he proceeds to abandon in favour of the conservative albeit cacaphonic convention of linguistics. I find this regrettable, and so shall employ his novel terminology indicating the conventional correlates parenthetically: the four fold classification is depicted in fig 35.

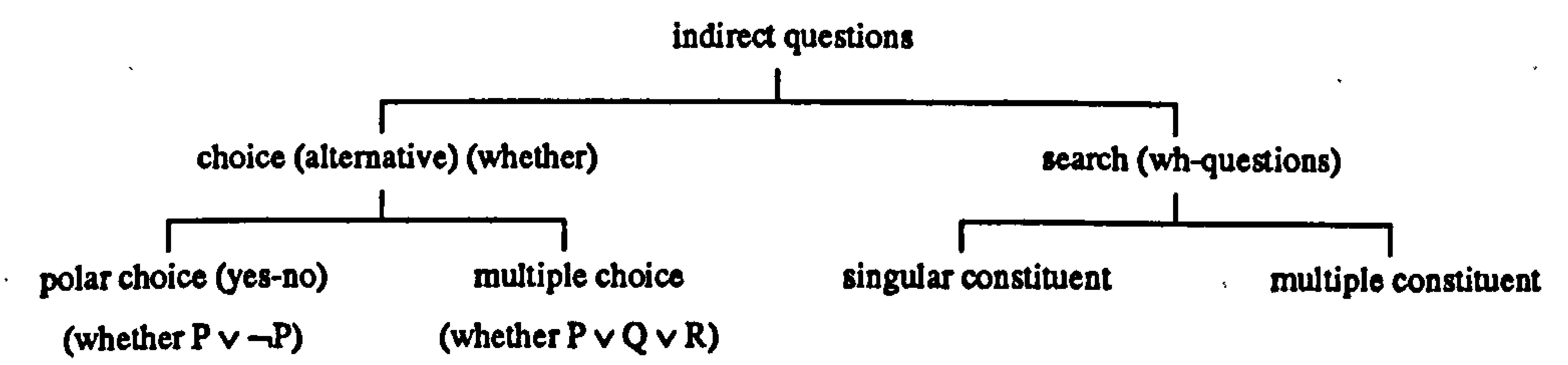


Fig 35

Polar choice questions are exemplified by any of the forms:

(100) Bill knows whether John walks.

(101) Bill knows whether or not John walks.

and

(102) Bill knows whether John walks or not.

while multiple choice questions may be exemplified by:

(103) John knows whether Mary smokes or Bill drinks.

on the preferred interpretation wherein the corresponding direct question does *not* invite the bald response "yes" (or "no").

A singular constituent search question contains a single interrogative noun phrase and no universally quantified noun phrases as for example:

(104) John knows who dates Mary.

while the characteristic of a multiple constituent question is the occurrence of *either* multiple interrogative noun phrases or some combination of interrogative and universally quantified noun phrases as in

(105) John knows which farmer milks which cow.

(106) Bill knows which student each professor recommends.

4.3.1. Basic Building Blocks

The suggestion that the denotation of a direct question should be the set of propositions constituting *possible* answers is due to Hamblin, [H2]. Karttunen adapts this suggestion and proposes that the denotation of an indirect question be the set of propositions representing *true and complete answers*. This form of denotation is to apply uniformly to all types of indirect question, thus all are to belong to the syntactic category Q = t//t corresponding to semantic type <<st>t>.

In defiance of the strong well formedness constraint, Karttunen introduces as a basic building block the proto-question to provide an intermediary stage in the formulation of questions proper. Proto questions are introduced by the rule:

SPQ If $\phi \in P_t$ then $f_{PQ}(\phi) \in P_Q$.

$f_{PQ}(\phi) = ?\phi$.

TPQ If $\phi \rightsquigarrow \phi'$ then $f_{PQ}(\phi) \rightsquigarrow \lambda r[{}^{\vee}r \wedge r = {}^{\wedge}\phi']$.

Since "$^{\wedge}\phi$" represents a proposition, ie. a function from indices to truth values, the entire translation of

$f_{PQ}(\phi)$ represents the set of all r such that r is a function from indices to truth values and r evaluates to true for the given point of reference. Like Montague's S2 this rule is unary, but a binary alternative will be proposed in due course.

All *real* indirect questions may be subsumed under question embedding verbs which Karttunen assigns to the category IV/Q. His rule QE for introducing such verbs becomes S63 in TMG and is specified as follows:

(S63) If $\delta \in P_{IV/Q}$ and $\phi \in P_Q$ and SA63 is fulfilled then $f_{63}(\delta,\phi) \in P_{IV}$.

SA63: ϕ does not begin with "?".

$f_{63}(\delta,\phi) = \delta\phi$.

(T63) If $\delta \Rightarrow \delta'$ and $\phi \Rightarrow \phi'$ then $f_{63}(\delta,\phi) \Rightarrow g_0(\delta',\phi') = \delta'(\hat{}\phi')$.

The structural analysis statement SA63 is unexceptionable since it amounts to ¬firstword(ϕ,?), and is accordingly formulable in procedural terms.

4.3.2. Choice Questions

****

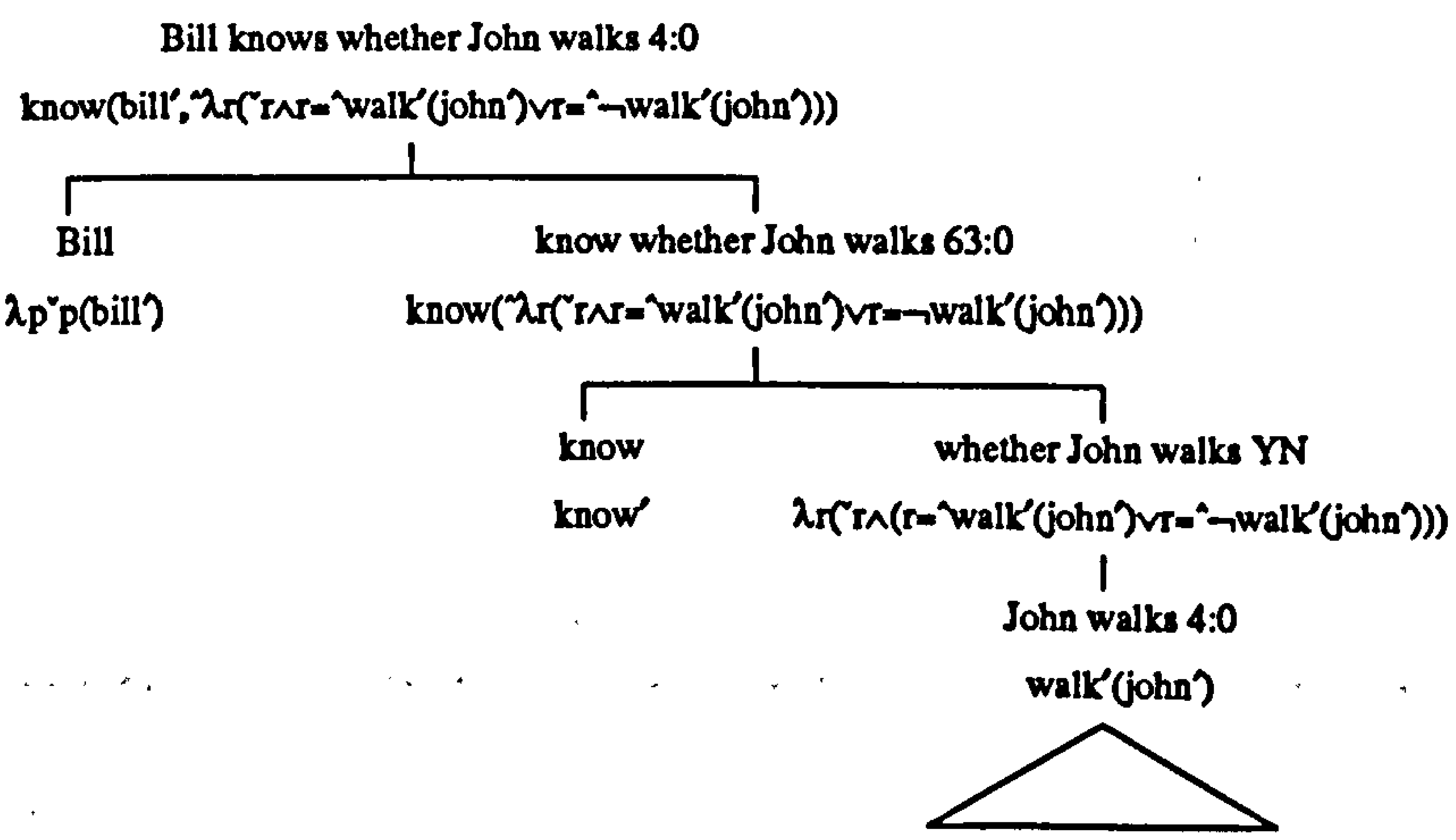


Fig 36

****

Polar choice (yes/no) questions are introduced in Karttunen's system by the rule:

(SYNQ) If $\psi \in P_Q$ and ψ has form ?ϕ then $f_{yn1}(\psi)$, $f_{yn2}(\psi)$ and $f_{yn3}(\psi) \in P_Q$.

$f_{yn1}(\psi)$ = whether ϕ, $f_{yn2}(\psi)$ = whether or not ϕ, $f_{yn3}(\psi)$ = whether ϕ or not.

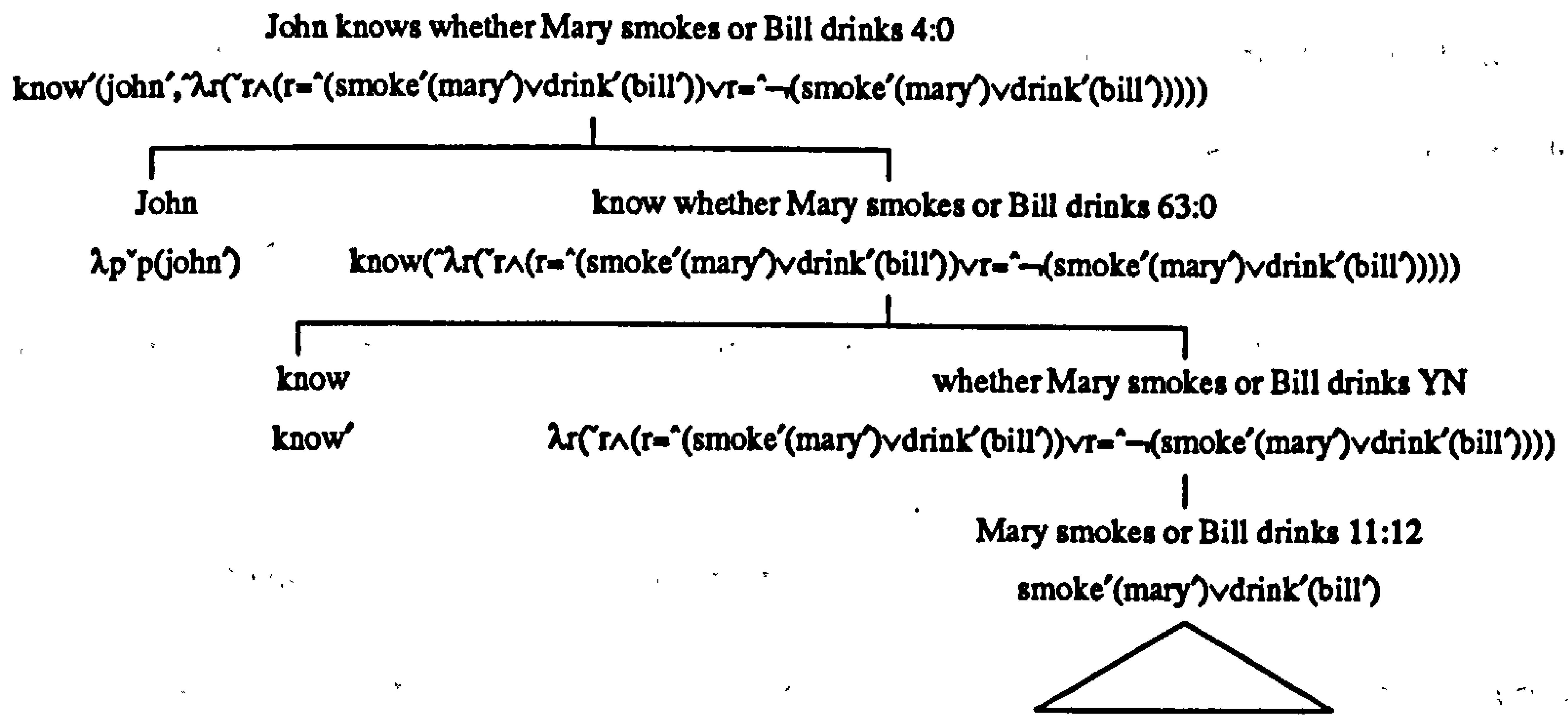


Fig 37

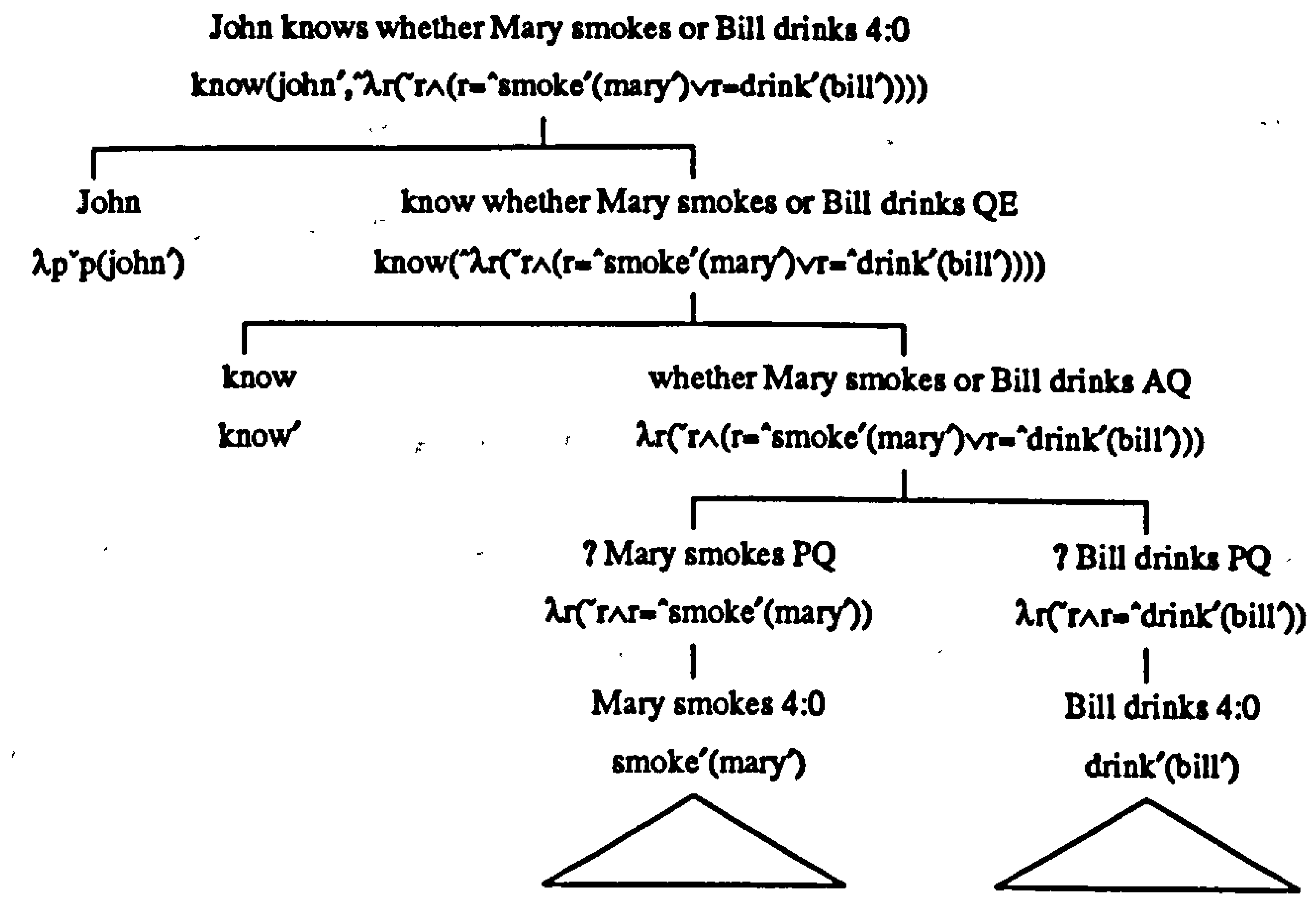


Fig 38

(TYNQ) If $\psi \Rightarrow \psi'$ then $f_{yn_n}(\psi)$ $\lambda r(\psi'(r) \vee (\neg\exists s\psi'(s) \wedge r = {}^\wedge\neg\exists s\psi'(s)))$.

where assuming that, as specified in TPQ, $\psi \Rightarrow \lambda r({}^\vee r \wedge r = {}^\wedge\phi')$, the final translation amounts to:

(a) $\lambda r(\lambda r({}^\vee r \wedge r = {}^\wedge\phi')(r) \vee (\neg\exists s\lambda r({}^\vee r \wedge r = {}^\wedge\phi')(s) \wedge r = {}^\wedge\neg\exists s\lambda r({}^\vee r \wedge r = {}^\wedge\phi')(s)))$

concerning which Karttunen writes that although the reduction is not obvious, the formula is equivalent to:

(b) $\lambda r({}^\vee r \wedge (r = {}^\wedge\phi' \vee r = {}^\wedge\neg\phi))$

which is "precisely what we were aiming for".

The tactic of employing a basic building block for *all* interrogatives seems to me at this point extremely curious, for in the context of polar choice questions the intervening proto question introduces nothing but obscurantism. If (b) is the translation ultimately required, then it would be simpler and more lucid to generate polar choice questions from declarative sentences directly by the rule:

(SYN) If $\phi \in P_t$ then $f_{yn1}(\phi)$, $f_{yn2}(\phi)$ and $f_{yn3}(\phi) \in P_Q$.

(TYN) If $\phi \Rightarrow \phi'$ then $f_{ynn}(\phi) \Rightarrow \lambda r(\check{}r \wedge (r = \hat{}\phi' \vee r = \hat{}\neg\phi'))$.

which is the version I shall adopt. The translation represents the singleton set containing either the proposition "$\hat{}\phi'$" or the proposition "$\hat{}\neg\phi'$" depending on which evaluates to "true" at the current point of reference.

Multiple choice (alternative) questions are admitted by the rule:

(SAQ) If $\psi_1, \psi_2, ..., \psi_n \in P_Q$ and ψ_k has the form $?\phi_k$ then $f_{AQ}(\psi_1, \psi_2, ..., \psi_n) \in P_Q$.
$f_{AQ}(\psi_1, \psi_2, ..., \psi_n) =$ whether ϕ_1 or ϕ_2 or ... or ϕ_n.

(TAQ) If $\psi_k \Rightarrow \psi'_k$ then $f_{AQ}(\psi_1, \psi_2, ..., \psi_n) \Rightarrow \lambda r(\psi'_1(r) \vee \psi'_2(r) \vee ... \vee \psi'_n(r))$.

where the translation represents the set of all the $\psi'_k(r)$ which are true.

Operation of the choice question rules is illustrated in figs 36...38. The tree in fig 37 represents the eccentric reading of (103) where the embedded question is conceived as offering a polar choice between a disjunction and its negative: the more normal interpretation in which the question offers a multiple choice is depicted in fig 38.

4.3.3. Binary Versions of the Basic and Choice Question Rules

Both SPQ and SYN are unary rules, while the multiple choice question rule SAQ is *n*-ary for arbitrary *n*. Computational implementation of the translation mechanism is simplified if all rules (including Montague's rule S2) are reduced to binary form, thus the three above mentioned rules require some modification to this end.

Conversion of the two unary rules requires the introduction of two pseudo lexical entries in the syntactic category Qmark such that:

Qmark = (t//t)/t

f(Qmark) = <<st><<st>t>>.

B_{Qmark} = {?yn, ?}.

?yn ≈» $\lambda s \lambda r({}^{\vee}r \wedge (r = {}^{\wedge\vee}s \vee r = {}^{\wedge}\neg{}^{\vee}s))$.

? ≈» $\lambda s \lambda r({}^{\vee}r \wedge r = s)$.

As already observed, these are markers having no phonological representation, thus I cannot remain committed to C3 (iii).

In place of SYN and SPQ we may now define the rules S20 and S21 as follows:

(S20) If α = ?yn and $\phi \in P_t$ then $f_{20.1}(\alpha,\phi)$, $f_{20.2}(\alpha,\phi)$, and $f_{20.3}(\alpha,\phi) \in P_Q$.

$f_{20.1}(\alpha,\phi) = {}_q$[whether ϕ], $f_{20.2}(\alpha,\phi) = {}_q$[whether or not ϕ], $f_{20.3}(\alpha,\phi) = {}_q$[whether ϕ or not].

(T20) If α ≈» α' and ϕ ≈» ϕ' then $f_{20.n}(\alpha,\phi)$ ≈» $g_0(\alpha',\phi') = \alpha'({}^{\wedge}\phi')$.

(S21) If α = ? and $\phi \in P_t$ then $f_{21}(\alpha,\phi) \in P_Q$.

$f_{21}(\alpha,\phi) = {}_q[\alpha\ \phi]$.

(T21) If α ≈» α' and ϕ ≈» ϕ' then $f_{21}(\alpha,\phi)$ ≈» $g_0(\alpha',\phi') = \alpha'({}^{\wedge}\phi')$.

In both cases the translation rule imposes straight forward functional application.

****

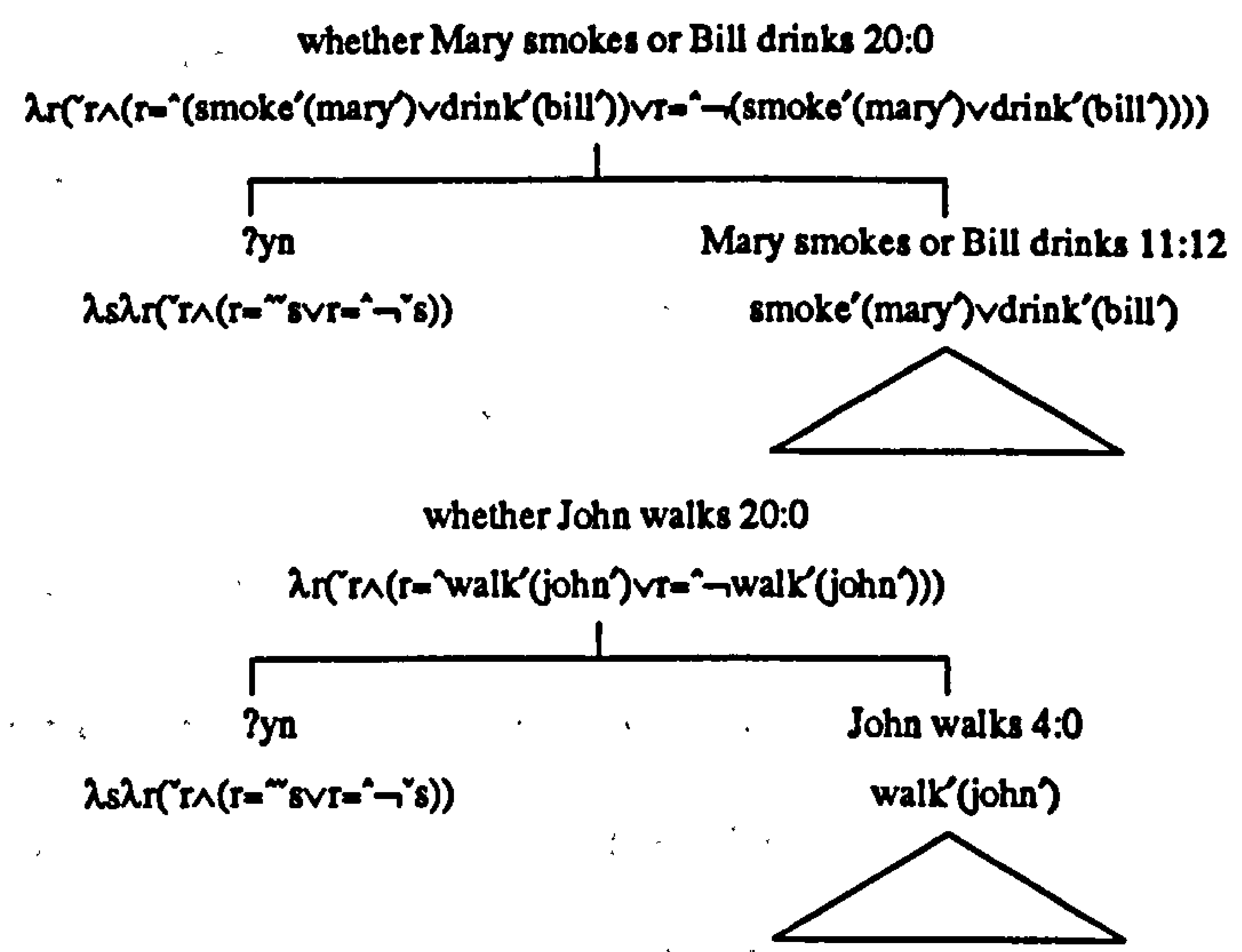


Fig 39

****

****

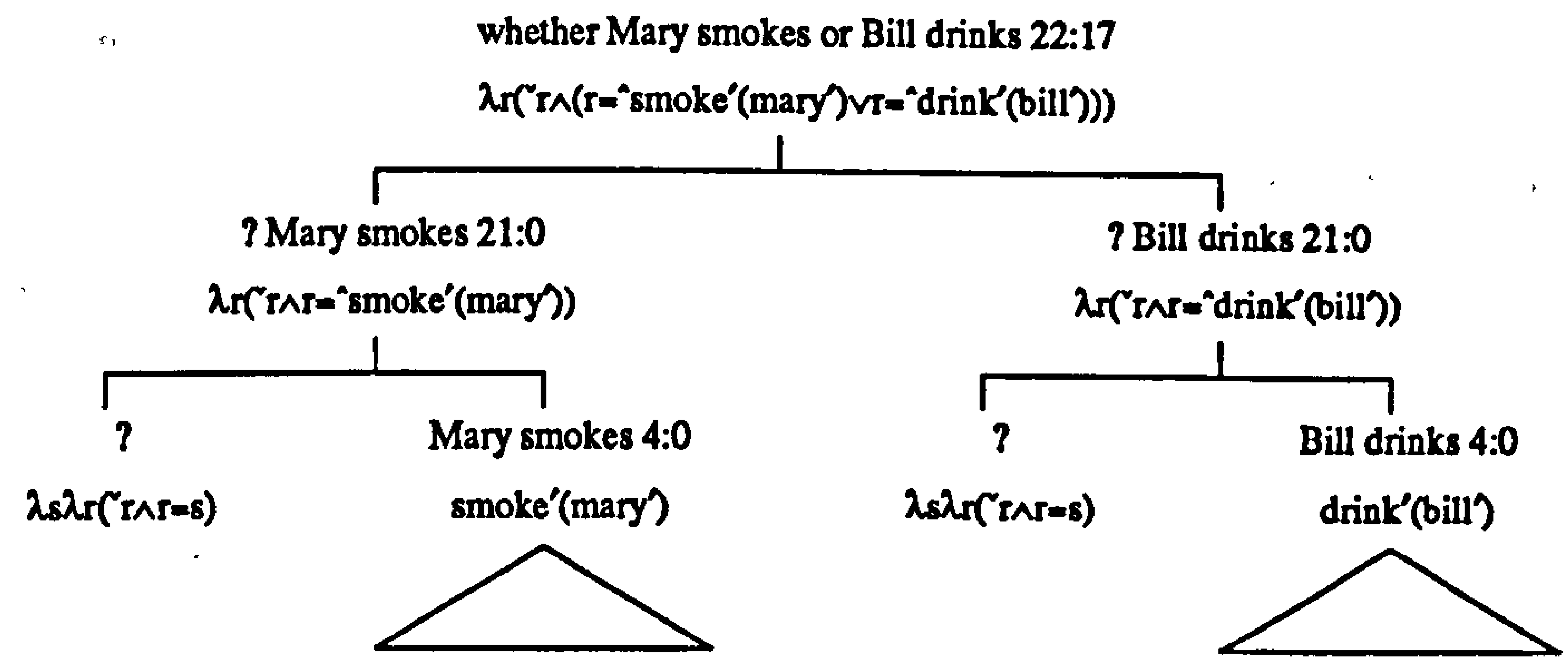


Fig 40

****

The *n*-ary rule SAQ must be replaced by a pair of rules, one to handle the initial choice between two proto questions, and the second to govern further choices between previous results and another proto question. This pair of rules is as follows:

(S22) If ψ_1 and $\psi_2 \in P_Q$ and SA22 is fulfilled then $f_{22}(\psi_1,\psi_2)$ P_Q.

SA22: daughters(ψ_1,?,ϕ_1), daughters(ψ_2,?,ϕ_2).

$f_{22}(\psi_1,\psi_2) = {}_q$[whether ϕ_1 or ϕ_2].

(T22) If $\psi_n \leadsto \psi'_n$ then $f_{22}(\psi_1,\psi_2) \leadsto g_{17}(\psi'_1\psi'_2) = \lambda r(\psi'_1(r) \vee \psi'_2(r))$.

(S23) If ψ_1 and $\psi_2 \in P_Q$ and SA23 is fulfilled then $f_{23}(\psi_1,\psi_2) \in P_Q$.

SA23: daughters(ψ_1,whether,ϕ_1), daughters(ψ_2,?,ϕ_2).

$f_{23}(\psi_1,\psi_2) = {}_q[\psi_1$ or $\phi_2]$.

(T23) If $\psi_n \leadsto \psi'_n$ then $f_{23}(\psi_1,\psi_2) \leadsto g_{17}(\psi'_1\psi'_2) = \lambda r(\psi'_1(r) \vee \psi'_2(r))$.

With the introduction of binary rules the YN subtrees in figs 36 and 37 must be replaced by those of fig 39, and the AQ subtree of fig 38 by that of fig 40.

4.3.4. Interrogative and Ordinary Noun Phrases

In compositional semantics the legitimacy of a syntactic structure is governed by its suitability for semantic interpretation: the semantic considerations have logical priority. This principle is admirably demonstrated in the development of Karttunen's account of interrogative noun phrases, the introduction of which is a prerequisite for any analysis of search questions.

Since the denotation of an indirect question is to be the set of true and complete answers, Karttunen suggests that:†60

> "the translation of *which girl sleeps* denotes a set which contains, for each sleeping girl, the proposition that she sleeps".

ie.

(107) which girl sleeps $\lambda r \exists X(\text{girl}'(X) \wedge {}^{\vee}r \wedge r = {}^{\wedge}\text{sleep}'(X))$.

but how is this formula to be derived?

One facility already available is the proto question, for it has been determined above that:

(108) ? he_n sleeps $\Rightarrow$ $\lambda r({}^{\vee}r \wedge r = {}^{\wedge}\text{sleep}'(x_n))$.

and Karttunen proposes that (107) should be derived from (108) as a result of quantifying in the interrogative noun phrase "which girl". It now transpires that if we admit:

(109) which girl $\Rightarrow$ $\lambda q \exists X(\text{girl}'(X) \wedge {}^{\vee}q(X))$.

then the translation in (107) comes from those of (108) and (109) by an operation analogous to T15, ie:

$$\lambda r(\lambda q \exists X(\text{girl}'(X) \wedge {}^{\vee}q(X))({}^{\wedge}\lambda x_n(\lambda r({}^{\vee}r \wedge r = {}^{\wedge}\text{sleep}'(x_n))(r)))) =$$

$$\lambda r(\lambda q \exists X(\text{girl}'(X) \wedge {}^{\vee}q(X))({}^{\wedge}\lambda x_n({}^{\vee}r \wedge r = {}^{\wedge}\text{sleep}'(x_n)))) =$$

$$\lambda r(\exists X(\text{girl}'(X) \wedge \lambda x_n({}^{\vee}r \wedge r = {}^{\wedge}\text{sleep}'(x_n))(X))) =$$

$$\lambda r \exists X(\text{girl}'(X) \wedge {}^{\vee}r \wedge r = {}^{\wedge}\text{sleep}'(X)).$$

Although syntactically distinct, "which girl" turns out to be semantically equivalent to "some girl" or "a girl", and by parity of reasoning the atomic interrogative noun phrases "who" and "what" should be semantically equivalent to "someone" and "something". Thus interrogative and ordinary noun phrases must map to the same semantic type, viz. <<s<et>>t>, and must be syntactically distinguishable in the Montague idiom as "slash" categories. Accordingly Karttunen introduces:

WH = t//IV.

†60. The condition embodied, which they regard as inadequate, is termed by Groenendijk and Stokhof, [G8], *weak exhaustiveness*. To be strongly exhaustive the set would also have to contain, for each non sleeping girl, the proposition that she does *not* sleep. Karttunen specifically excludes strong exhaustiveness (by forbidding interrogative quantification into choice questions, which would permit it), since he denies that "which girl sleeps" and "which girl *does not* sleep" are synonyms.

$f(\mathrm{WH}) = <<s<et>>t>$.

$B_{WH} = \{$who, whom, what$\}$.[†61]

who, whom, what $\rightsquigarrow$ $\lambda p \exists X\, {}^{\vee}p(X)$.

and formulates a unary rule for introducing synthetic interrogative noun phrases as follows:

(SWHP) if $\zeta \in P_{CN}$ then $\ulcorner$ which $\zeta \urcorner$ and $\ulcorner$ what $\zeta \urcorner \in P_{WH}$.

(TWHP) If $\zeta \rightsquigarrow \zeta'$ then $\ulcorner$ which $\zeta \urcorner$ and $\ulcorner$ what $\zeta \urcorner \rightsquigarrow \lambda p \exists X(\zeta'(X) \wedge {}^{\vee}p(X))$.

As with previous innovations, we shall adopt a binary alternative S29, which will involve the introduction of a category of basic interrogative determiners. In view of the affinities shared by interrogative and ordinary noun phrases, it will prove convenient at this juncture also to replace Montague's original S2 with a binary variation. Let:

DET = T/CN.

WDET = WH/CN.

$f(\mathrm{DET}) = <<s<et>><<s<et>>t>>$.

$f(\mathrm{WDET}) = <<s<et>><<s<et>>t>>$.

$B_{DET} = \{$a, the, every$\}$.

$B_{WDET} = \{$which, what$\}$.

then:

a $\rightsquigarrow$ $\lambda p \lambda q \exists X({}^{\vee}p(X) \wedge {}^{\vee}q(X))$.

the $\rightsquigarrow$ $\lambda p \lambda q \exists Y(\forall X({}^{\vee}p(X) \leftrightarrow X = Y) \wedge {}^{\vee}q(Y))$.

every $\rightsquigarrow$ $\lambda p \lambda q \forall X({}^{\vee}p(X) \rightarrow {}^{\vee}q(X))$.

which, what $\rightsquigarrow$ $\lambda p \lambda q \exists X({}^{\vee}p(X) \wedge {}^{\vee}q(X))$.

given which facilities we may formulate:

(S2) If $\alpha \in P_{DET}$ and ζ P_{CN} then $f_2(\alpha,\zeta) \in P_T$.

---

†61. Since in English "who" and "whom" (and compounds thereof) are the only case marked interrogative noun phrases, it is economical to treat *both* as primitive. I have therefore modified Karttunen's account in this respect.

$f_2(\alpha,\zeta) = \alpha\,\zeta.$

(T2) If $\alpha \Rightarrow \alpha'$ and $\zeta \Rightarrow \zeta'$ then $f_2(\alpha,\zeta) \Rightarrow g_0(\alpha',\zeta') = \alpha'(\hat{}\zeta')$.

(S29) If $\alpha \in P_{WDET}$ and ζ P_{CN} then $f_{29}(\alpha,\zeta) \in P_{WH}$.

$f_{29}(\alpha,\zeta) = \alpha\,\zeta.$

(T29) If $\alpha \Rightarrow \alpha'$ and $\zeta \Rightarrow \zeta'$ then $f_{29}(\alpha,\zeta) \Rightarrow g_0(\alpha',\zeta') = \alpha'(\hat{}\zeta')$.

4.3.5. Search Questions

In his initial exposition, [K4], Karttunen handles the formulation of search questions by a single rule WHQ (wh- quantification) having two alternative subdivisions (A) and (B). Alternative (A) applies only when an interrogative noun phrase is quantified into a proto question, as adumbrated above, in order to form a search question. If the proto question contains only syntactic variables of a single index, then the result will be a single constituent search question: if however variables of divers indices are present then the possibility of further quantification arises. The (B) alternative accordingly accommodates the quantifying into an existing search question of additional interrogative noun phrases.

Case (A) simulates the effect of the wh- movement transformation, ie. the interrogative noun phrase is *preposed*, and the leading syntactic variable of appropriate index deleted: thus from "who" and " ?he_n dates Mary" we derive "who dates Mary", while from "what" and "? John reads him_n" we obtain "what John reads". In the alternative case (B) there is no movement and the interrogative noun phrase simply replaces the leading syntactic variable of required index so as to allow the derivation of "which farmer sells which horse" from "which horse" and "which farmer sells him_n".

One consequence of this ploy is to ensure that in a search question involving multiple interrogative noun phrases, the preposed phrase, ie the one initially quantified into a proto question, must have minimal scope.†62

†62. Bennett, [B9], protests the opposite, but neither of his arguments appears convincing to me. His first objection is based on the assumption that the *direct* questions:

(i) which woman in the room loves which man ?

and

(ii) which man in the room does which woman love?

make the respective presuppositions:

(iii) exactly one woman is such that she loves exactly one man.

and

Karttunen's own statement of the rule is as follows:

(SWHQ) If $\alpha \in P_{WH}$ and $\phi \in P_Q$, ϕ contains a variable with index n, and ϕ does not begin with "whether" then $f_{WHQ,n}(\alpha,\phi) \in P_Q$.

(A) If $\phi = ?\,\psi$ then $f_{WHQ,n}(\alpha,\phi) = \phi'$ where ϕ' is derived as follows:

(i) substitute ? by α adjusted if necessary to match the case of the first variable in ψ with index n.

(ii) delete the first variable with index n.†63

(iii) replace all other variables with index n by surface pronouns of corresponding case and matching α in gender.

(B) If $\phi \neq ?\,\psi$ then $f_{WHQ,n}(\alpha,\phi) = \phi'$ where ϕ' is derived as follows:

(i) substitute first variable with index n by α suitably adjusted to preserve the case of the substituens.

(ii) replace all other variables with index n by surface pronouns of corresponding case and matching α in gender.

(TWHQ) If $\alpha \Rightarrow \alpha'$ and $\phi \Rightarrow \phi'$ then $f_{WHQ,n}(\alpha,\phi) \Rightarrow \lambda r(\alpha'(\hat{}\lambda x_n(\phi'(r))))$.

This rule may conveniently be replaced by two rules, S24 and S25, each devoted exclusively to one case:

(S24) If $\alpha \in P_{WH}$ and $\phi \in P_Q$ and SA24 is fulfilled then $f_{24,n}(\alpha,\phi) \in P_Q$.

SA24: firstword(ϕ,?), leadvar(n,ϕ,V,C), α = wh([W],[G,N,C]).†64

(iv) exactly one man is such that exactly one woman loves him.

Given a room containing only John, Bill, Mary and Lucy such that Mary loves only John but Lucy loves both men, Bennett claims that the only correct answers to (i) and (ii) are:

(v) Mary loves John.

and

(vi) Bill is loved by Lucy.

In my idiolect neither question solicits a unique pairing. Question (i) asks for a list of *all* pairings, 1:1 or otherwise, and accordingly I agree with Karttunen and Peters that (i) is equivalent to "which woman in the room loves each man in the room" on the interpretation where "each man" has wide scope. With regard to (ii) I am less happy. There may indeed be an expectation that all pairings are 1:1, but I find it difficult to dismiss a list of *all* pairings as erroneous.

Bennett's second objection is that Karttunen cannot handle:

(vii) which man loves which woman whom he knows.

but the problem here is analagous to that of the "donkey sentence", a solution to which is hardly to be expected in a theory of interrogatives.

†63. Karttunen also suggests rodmanising all variables to the left of the deletion so as to accommodate the *crossing constraint* of Kuno and Robinson and thus block the derivation of:

"tell me whom who killed".

I do not find this sentence deviant, moreover Karttunen's use of rodmanised variables would be incompatible with Rodman's own which I wish to preserve.

****

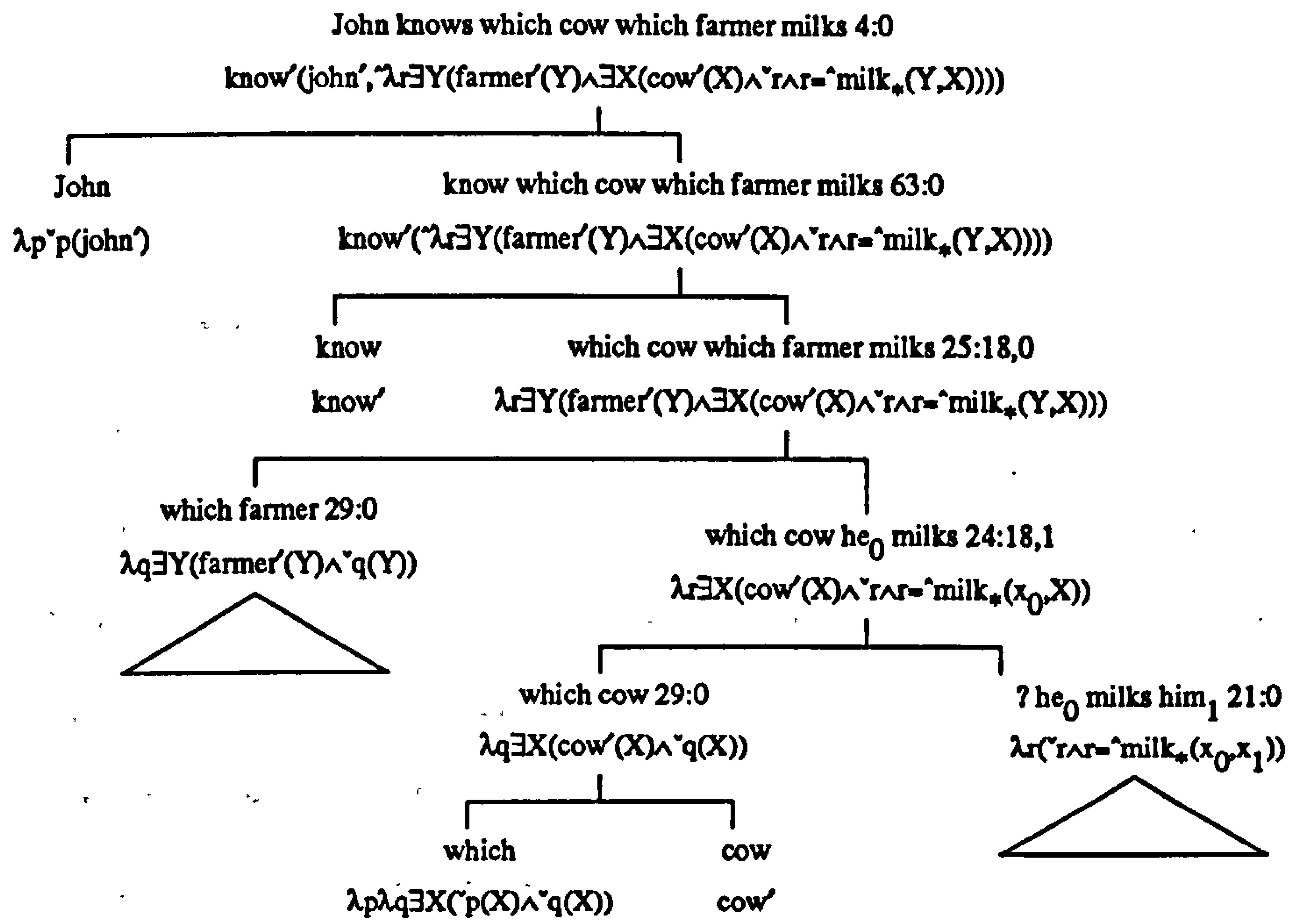


Fig 41

****

****

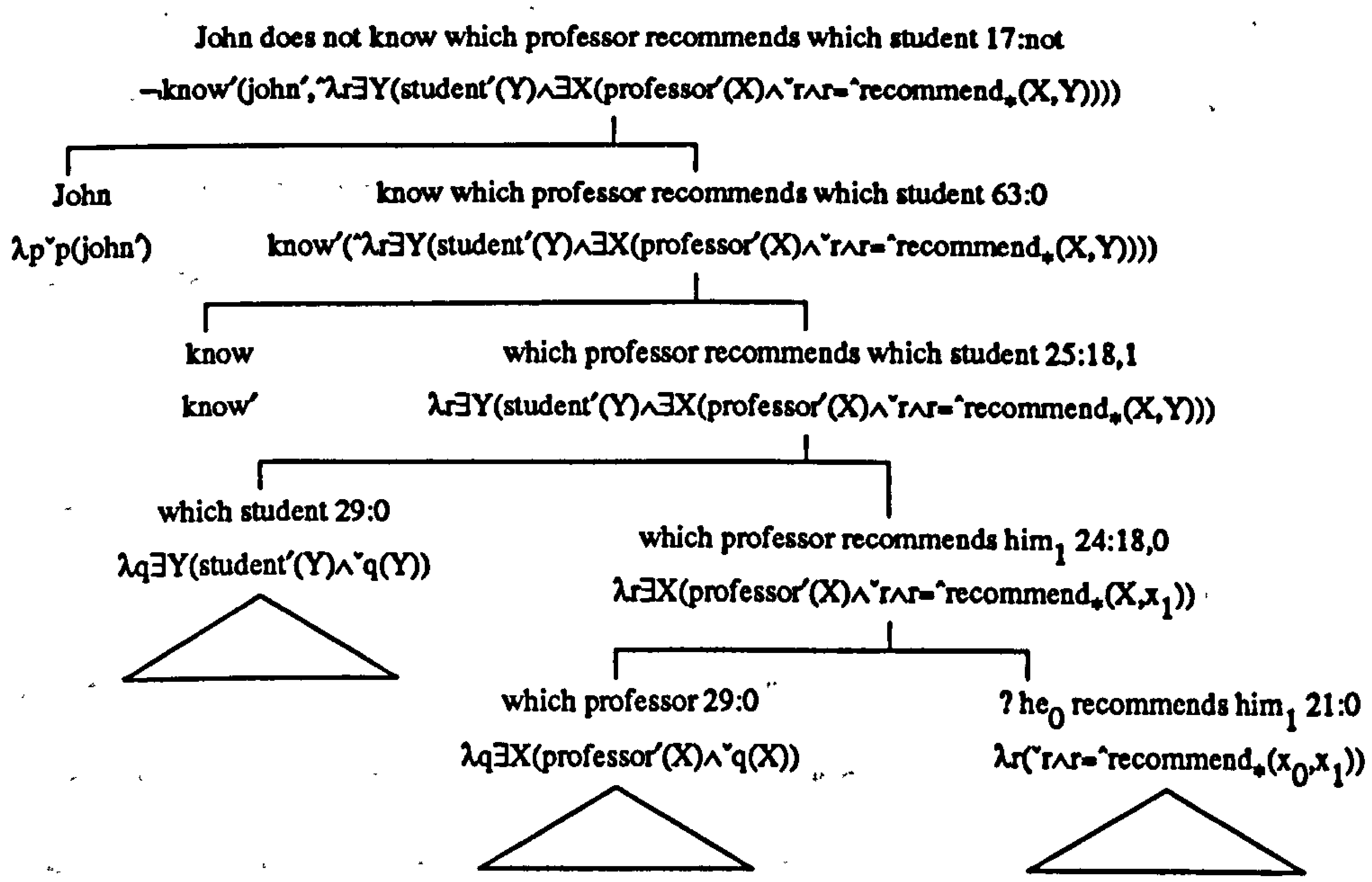


Fig 42

****

†64. This formulation expects the correct case as input.

$f_{24,n}(\alpha,\phi) = \alpha\,\xi$ where delete(n,ϕ,ψ), esub(α,n,ψ,ξ).

(T24) If $\alpha \Rightarrow \alpha'$ and $\phi \Rightarrow \phi'$ then $f_{24,n}(\alpha,\phi) \Rightarrow g_{18,n}(\alpha',\phi') = \lambda r(\alpha'(\hat{}\lambda x_n(\phi'(r))))$.

(S25) If $\alpha \in P_{WH}$ and $\phi \in P_Q$ and SA25 is fulfilled then $f_{25,n}(\alpha,\phi) \in P_Q$.

SA25: firstword(ϕ,Q), Q $\notin$ {?, whether}, leadvar(n,ϕ,V,C), α = wh([W],[G,N,C]).

$f_{25,n}(\alpha,\phi) = \psi$ where qsub(α,n,ϕ,ψ).

(T25) If $\alpha \Rightarrow \alpha'$ and $\phi \Rightarrow \phi'$ then $f_{25,n}(\alpha,\phi) \Rightarrow g_{18,n}(\alpha',\phi') = \lambda r(\alpha'(\hat{}\lambda x_n(\phi'(r))))$.

The operation of the interrogative noun phrase and search question rules is demonstrated in the examples of figs 41 & 42 which serve also to illustrate an important consideration in the employment of variables. One effect of the translation operation licenced by T24 and T25 is the reduction and subsequent replacement in wide scope of an embedded "λr": accordingly a *specific* variable "r" (= "$v_{0,<st>}$") may be used on all invocations. By contrast, successive applications of T29 must introduce *distinct* existentially quantified variables in order that illegitimate capture by an existing quantifier be avoided.

The choice of capitalised variables as surrogates for "$v_{j,e}$" (where j>0 is the next unused odd subscript) reflects a convention built into the PROLOG interpreter and to which recourse is made in the computerised grammar TMDCG. A capitalised variable will be treated by the interpreter as a PROLOG variable and will automatically be represented internally by a unique identifier not already in use, whilst a lower case variable will be regarded in PROLOG terms as an atom and will remain specific throughout a derivation.

4.3.6. Multiple Constituent Search Questions Reconsidered

Karttunen's original 1977 account of multiple constituent search questions has the unfortunate consequence that within a member of P_{WH} any non interrogative noun phrase must have narrower scope than all interrogative noun phrases: for an application of S14 must take place *either* before the initial proto question has been formed *or* after the final indirect question has been embedded in a complete sentence. Given the sentence:

(110) John does not know which professor recommends every student.

the first option allows the tree in fig 43 and the second that of fig 44, where the former represents the

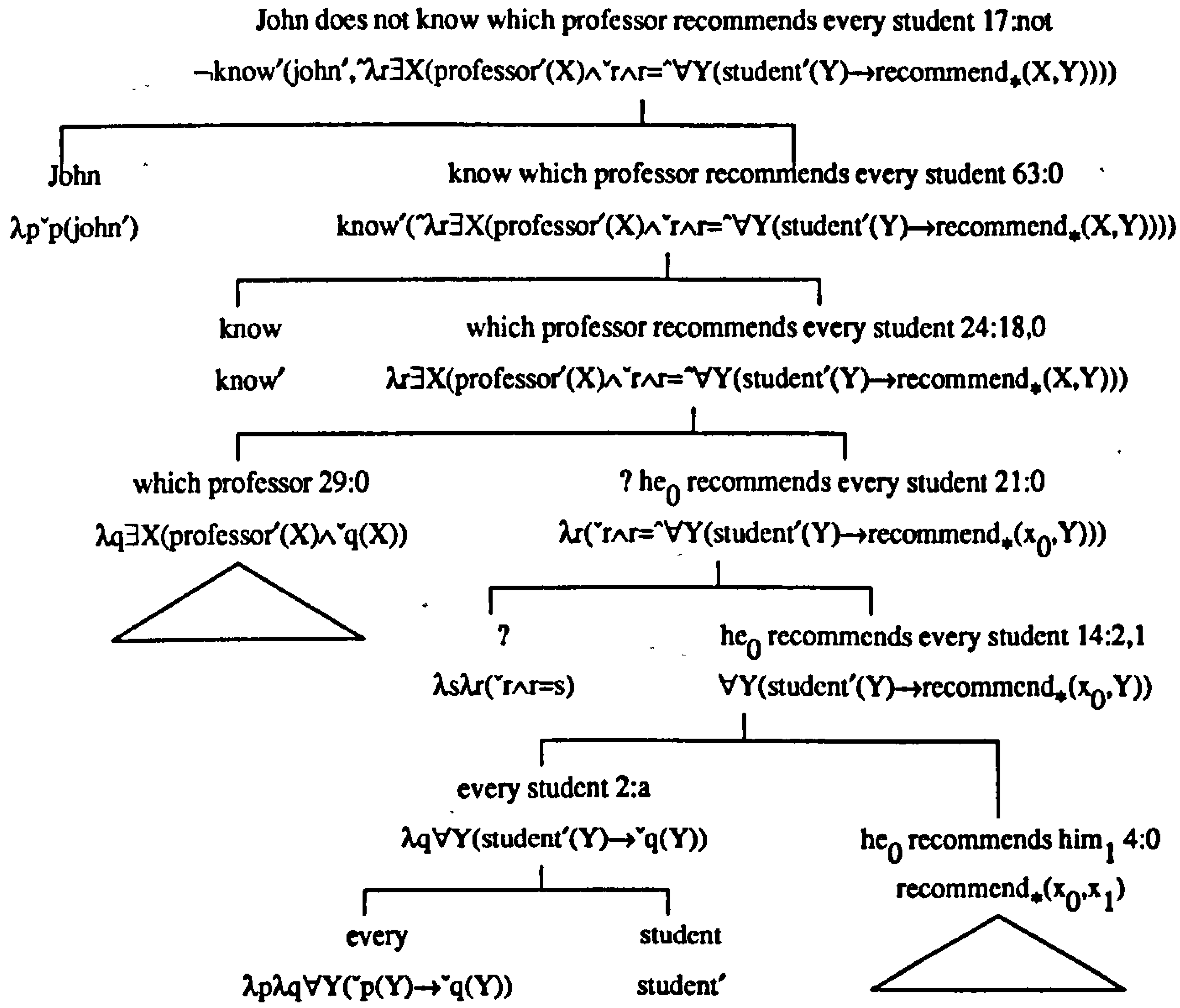

John does not know which professor recommends every student 17:not
¬know′(john′,ˆλr∃X(professor′(X)∧ˇr∧r=ˆ∀Y(student′(Y)→recommend*(X,Y))))
John
λpˇp(john′)
know which professor recommends every student 63:0
know′(ˆλr∃X(professor′(X)∧ˇr∧r=ˆ∀Y(student′(Y)→recommend*(X,Y))))
know
know′
which professor recommends every student 24:18,0
λr∃X(professor′(X)∧ˇr∧r=ˆ∀Y(student′(Y)→recommend*(X,Y)))
which professor 29:0
λq∃X(professor′(X)∧ˇq(X))
? he0 recommends every student 21:0
λr(ˇr∧r=ˆ∀Y(student′(Y)→recommend*(x0,Y)))
?
λsλr(ˇr∧r=s)
he0 recommends every student 14:2,1
∀Y(student′(Y)→recommend*(x0,Y))
every student 2:a
λq∀Y(student′(Y)→ˇq(Y))
he0 recommends him1 4:0
recommend*(x0,x1)
every
λpλq∀Y(ˇp(Y)→ˇq(Y))
student
student′

Fig 43

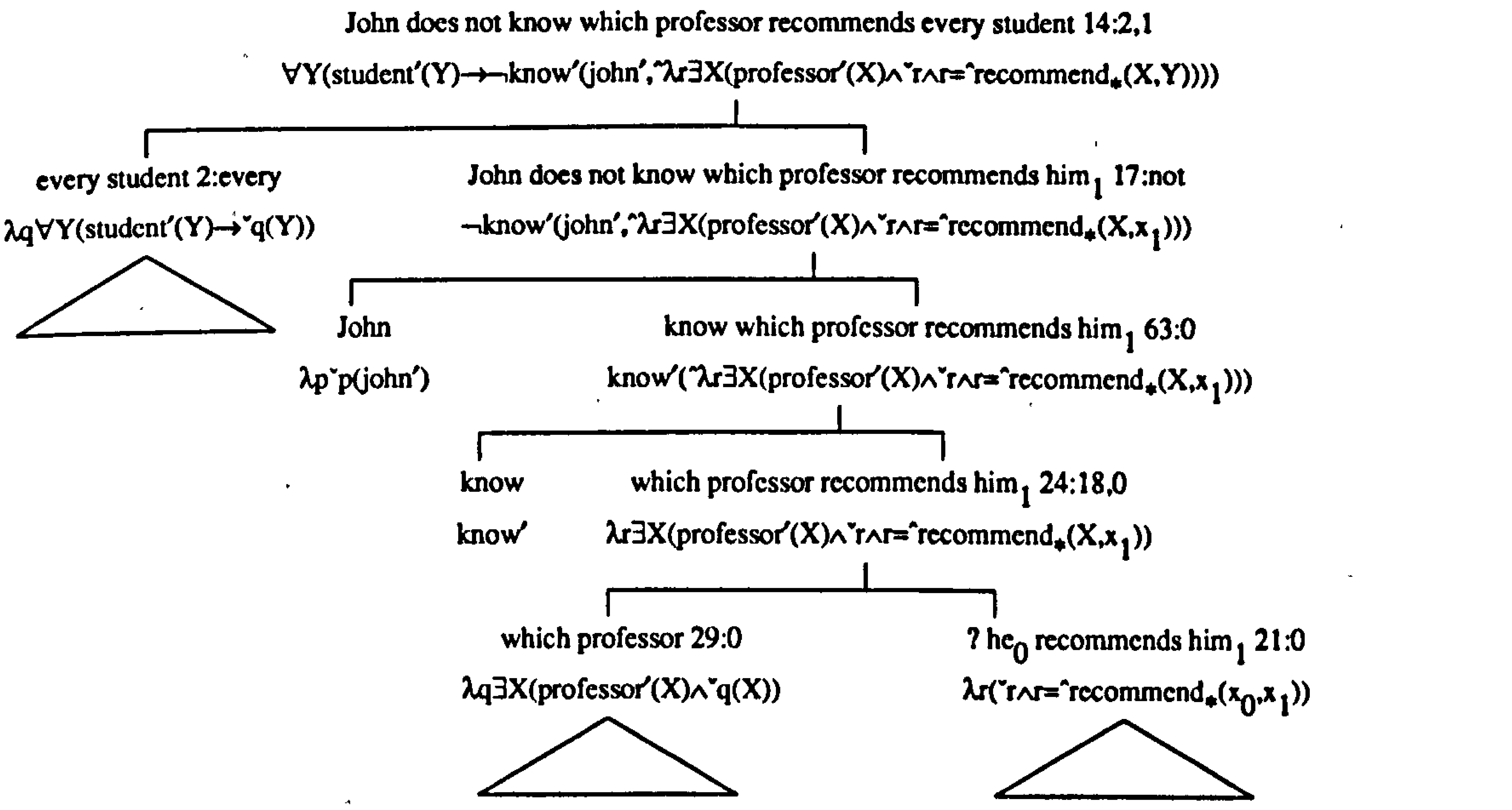

John does not know which professor recommends every student 14:2,1
∀Y(student′(Y)→¬know′(john′,ˆλr∃X(professor′(X)∧ˇr∧r=ˆrecommend*(X,Y))))
every student 2:every
λq∀Y(student′(Y)→ˇq(Y))
John does not know which professor recommends him1 17:not
¬know′(john′,ˆλr∃X(professor′(X)∧ˇr∧r=ˆrecommend*(X,x1)))
John
λpˇp(john′)
know which professor recommends him1 63:0
know′(ˆλr∃X(professor′(X)∧ˇr∧r=ˆrecommend*(X,x1)))
know
know′
which professor recommends him1 24:18,0
λr∃X(professor′(X)∧ˇr∧r=ˆrecommend*(X,x1))
which professor 29:0
λq∃X(professor′(X)∧ˇq(X))
? he0 recommends him1 21:0
λr(ˇr∧r=ˆrecommend*(x0,x1))

Fig 44

interpretation according to which John is unable to identify a generous professor who makes universal recommendations while the latter embodies the reading according to which John is unable to pair professors with *any* of their recommendees.

****

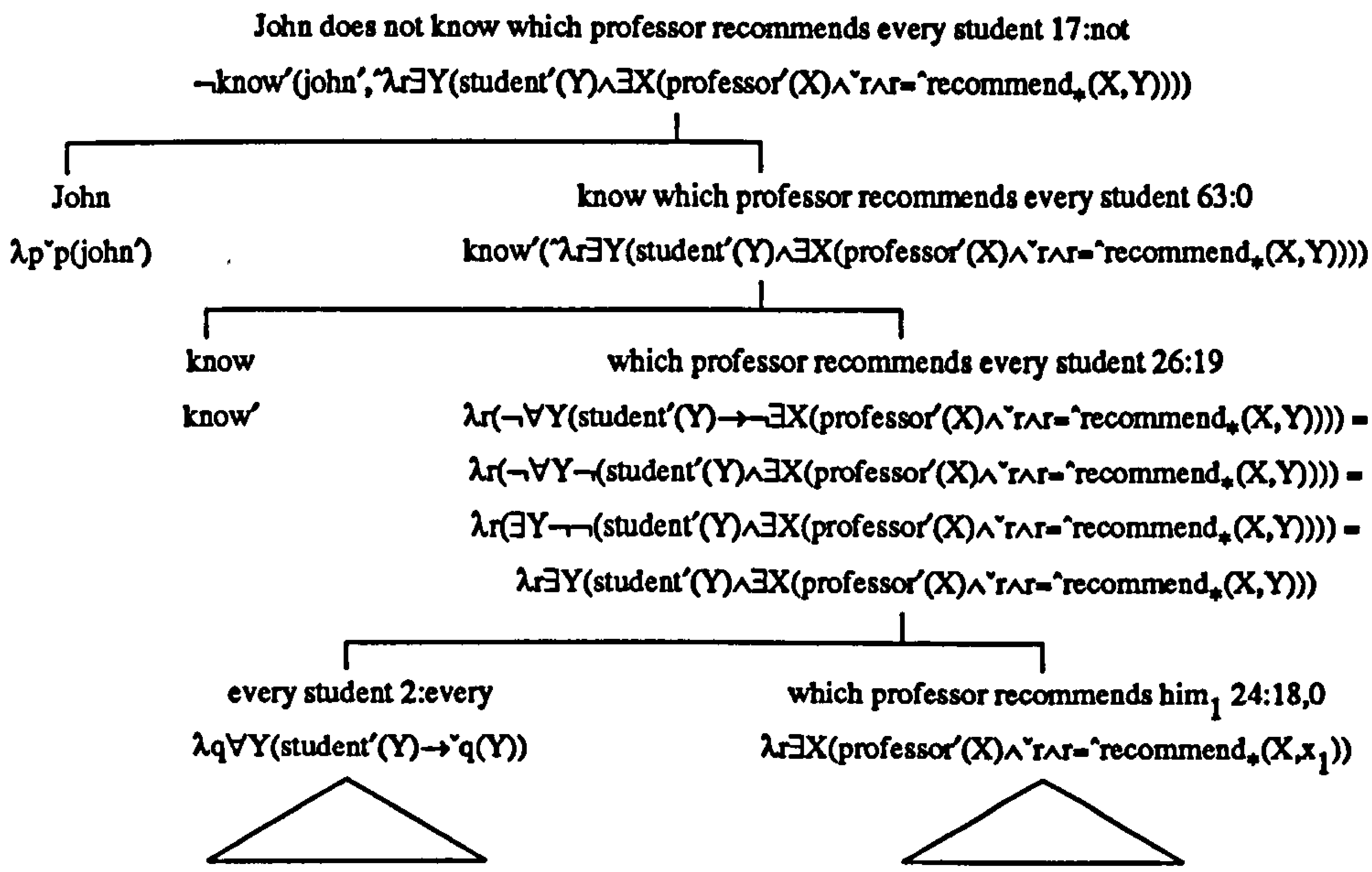


Fig 45

****

There is however a more plausible interpretation in which "every student" has wide scope with respect to "which professor", but in which (110) may be true even though there are *some* pairings of recommendee and professor which John *is* able to complete. To obtain this reading "every student" must receive narrower scope than "know", ie. it must be quantified into the indirect question "which professor recommends him_n".

In their 1980 revision Karttunen and Peters, [K5], remedy this deficiency by introducing a new semantic rule[†65] for quantifying ordinary noun phrases into questions. No syntactic counterpart is provided,

†65. Karttunen and Peters offer only semantic rules in [K5], but for purposes of comparison the following may be regarded as equivalent:

[K5](10) = WHQ,*n*(A) = S24

[K5](12) = WHQ,*n*(B) = S25

[K5](26) = new rule = S26

The need for a third scoping possibility was first noticed by Bennett, [B9p295], who considers the example:

John wonders where two unicorns live

on an interpretation wherein there is no speaker's ontological commitment to unicorns, but John's wonderment is concerning separate abodes. The example requires more revision to PTQ than has been introduce, and prompts Karttunen and Peters to devise a convoluted alternative also phrased in terms of "wonder". My own example above is essentially a simplification which achieves the same purpose.

but the complete rule may be reconstructed thus:

(S26) If $\alpha \in P_T$ and $\phi \in P_Q$ and SA26 is fulfilled then $f_{26,n}(\alpha,\phi) \in P_Q$.

SA26: $\neg$firstword(ϕ,whether), leadvar(n,ϕ,V,C).

$f_{26,n}(\alpha,\phi) = \psi$ such that qsub(α,n,ϕ,ψ).

(T26) If $\alpha \twoheadrightarrow \alpha'$ and $\phi \twoheadrightarrow \phi'$ then $f_{26,n}(\alpha,\phi) \twoheadrightarrow g_{19,n}(\alpha',\phi') = \lambda r(\neg\alpha'(\hat{}\lambda x_n(\neg\phi'(r))))$.

The effect of the new rule is to allow the tree of fig 45, the top node of which should be compared with that of the top node in fig 42. Semantically, "which professor recommends every student", with "every student" in wide scope, and "which professor recommends which student" are equivalent.

CHAPTER 5. PASSIVISATION, TENSE AND ASPECT

¶ This chapter commences with a review of the accounts of passivisation, tense and aspect presented by Bach, [B1, B2, B3] and Dowty, [D6, D8], since these treatments form the basis for the innovations to be introduced in TMG. Dowty's two dimensional system of tense logic, which is developed to support his analysis of tense operators and temporal adverbials, is considered in this context. The analysis adopted in TMG requires a multi dimensional tense logic; accordingly the chapter closes with a discussion of the multi dimensional tradition originating with Reichenbach, [R1], and developed by both Bull, [B16], and Bruce, [B15].

5.1. Bach's Account of Passivisation

In developing an exhaustive classification of verb phrases, the details of which would take us beyond the bounds of the present investigation, Bach, [B2], presents a treatment of passivisation based upon hints provided by Thomason, [T4].[†56] The rule implied, but never explicitly formulated, by Thomason would have the form:

(SP1) If $\gamma \in P_{TV}$ then $f_p(\gamma) \in P_{IV}$.

$f_p(\gamma) = EN(\gamma) = \gamma$ with the main verb replaced by its past participle.

(TP1) If $\gamma \Rightarrow \gamma'$ then $EN(\gamma) \Rightarrow \lambda X \exists Y(\gamma'(\hat{}\lambda p\check{}p(X)))(Y)$.[†57]

†56. The origins of Bach's treatment occur in an earlier paper, [B1], in which he first distinguishes complex transitive verb phrases like *persuade to go* from complex predicative intransitive verb phrases such as *promise Mary*, his claim being that only the former admit passivisation as evidenced by:

(i) John persuaded Mary to go.

(ii) Mary was persuaded to go by John.

(iii) John promised Mary to go.

(iv) * Mary was promised to go by John.

The existence of structurally analogous sentences only some of which are passivisable is taken as evidence against the classical transformational view that passivisation is a transformation applicable to complete sentences. Likewise the appearance of complex passivisable verb phrases militates against any alternative treatment of passivisation as a lexically defined phenomenon.

In the earlier paper, Bach, like Partee, represents expressions as labeled bracketed strings and introduces subfunctions in terms of which structural operations are to be defined. Were complex transitive verb phrases of the kind illustrated in (i) to be included in TMG, then simulation of some of his additional functions would become significant. His syntax rule for combining a (complex) transitive verb phrase with its object takes the form:

- If $\alpha \in P_{TV}$ and $\beta \in P_T$ then $RWRAP(\alpha,\beta) \in P_{IV}$.

where RWRAP ("*right wrap*") is defined thus:

(a) If α is simple then $RWRAP(\alpha,\beta) = \alpha\,\beta$.

(b) If $\alpha = [_{cat}\, X\, W]$ then $RWRAP(\alpha,\beta) = X\,\beta\,W$.

Constructions for which "right wrap" is required do not feature in TMG.

†57. In relational form this expression is equivalent to $\lambda X \exists Y(\gamma'(Y, \hat{}\lambda p\check{}p(X)))$.

According to this rule, the passive auxiliary is to be introduced syncategorematically, a phenomenon which as a side effect makes it impossible to isolate the passive verb phrase governed by that auxiliary: the only categories involved in passivisation are TV and IV. By contrast, Bach argues that unless a distinct intermediate category of passive verb phrases (PVP) is recognised it becomes impossible to explain the ambiguity in a sentence such as:

(111) John was attacked and bitten by a vicious dog.

If the dog is responsible for both the attack and the biting, then two transitive verb phrases must be conjoined prior to agentive passivisation as in:

(112) John $[_{IV}$ was $[_{PVP}$ $[_{TV}$ $[_{TV}$ attacked] and $[_{TV}$ bitten]] by a vicious dog]].

However if the attack was by an unspecified agent, with the dog guilty only of the biting, then one agentless and one agentive passive verb phrase must be conjoined prior to the introduction of the auxiliary; for a necessary condition of conjunction is membership of a common syntactic category. The alternative reading has the form:

(113) John $[_{IV}$ was $[_{PVP}$ $[_{PVP}$ attacked] and $[_{PVP}$ bitten by a vicious dog]]].

In order to provide for the isolation of passive verb phrases Bach replaces Thomason's rule by the following pair which handle agentless and agentive passives respectively. A new category PVP of passive verb phrases, with f(PVP) = <et>, must be presumed by these formulations.[†58]

(SP2) If $\gamma \in P_{TV}$ then $f_{p2}(\gamma) \in P_{PVP}$.

$f_{p2}(\gamma) = EN(\gamma) = \gamma$ with the main verb replaced by its past participle.

(TP2) If $\gamma \Rightarrow \gamma'$ then $EN(\gamma) \Rightarrow \lambda X \exists Y(\gamma'(\hat{}\lambda p\,\check{}p(X)))(Y)$.

(SP3) If $\alpha \in P_T$ and $\gamma \in P_{TV}$ then $f_{p3}(\alpha,\gamma) \in P_{PVP}$.

$f_{p3}(\alpha,\gamma) = EN(\gamma)$ by α.

(TP3) If $\alpha \Rightarrow \alpha'$ and $\gamma \Rightarrow \gamma'$ then $EN(\gamma) \Rightarrow \lambda X\alpha'(\hat{}\lambda Y(\gamma'(\hat{}\lambda p\,\check{}p(X)))(Y))$.

No rule for incorporating the passive auxiliary is formulated by Bach, but he suggests that (passive)

---

†58. Bach employs Montague's original system of types, not Bennett's simplified version, thus he makes PVP = $t/^{n}e$ for some n corresponding to semantic type <<se>t>. For compatibility with Bennett's system PVP must be regarded as primitive.

"*be*" should be assigned to category IV/PVP; moreover, given the translations proposed in (TP2) and (TP3), the semantic effect of the auxiliary should be vacuous, ie. the interpretation of passive "*be*" should constitute an identity function. Such a provision may be achieved by inclusion of the following rules:

(SPx) If β is a passive auxiliary and $\gamma \in P_{PVP}$ then $f_{px}(\beta,\gamma) \in P_{IV}$.

$$f_{px}(\beta,\gamma) = \beta\,\gamma.$$

(TPx) If $\beta \Rightarrow \beta'$ and $\gamma \Rightarrow \gamma'$ then $f_{px}(\beta,\gamma) \Rightarrow \beta'({}^{\wedge}\gamma')$

where the passive auxiliary is given the translation $\lambda p \lambda X({}^{\vee}p(X))$. The effect of the proposed rules is illustrated in figs 46 and 47.

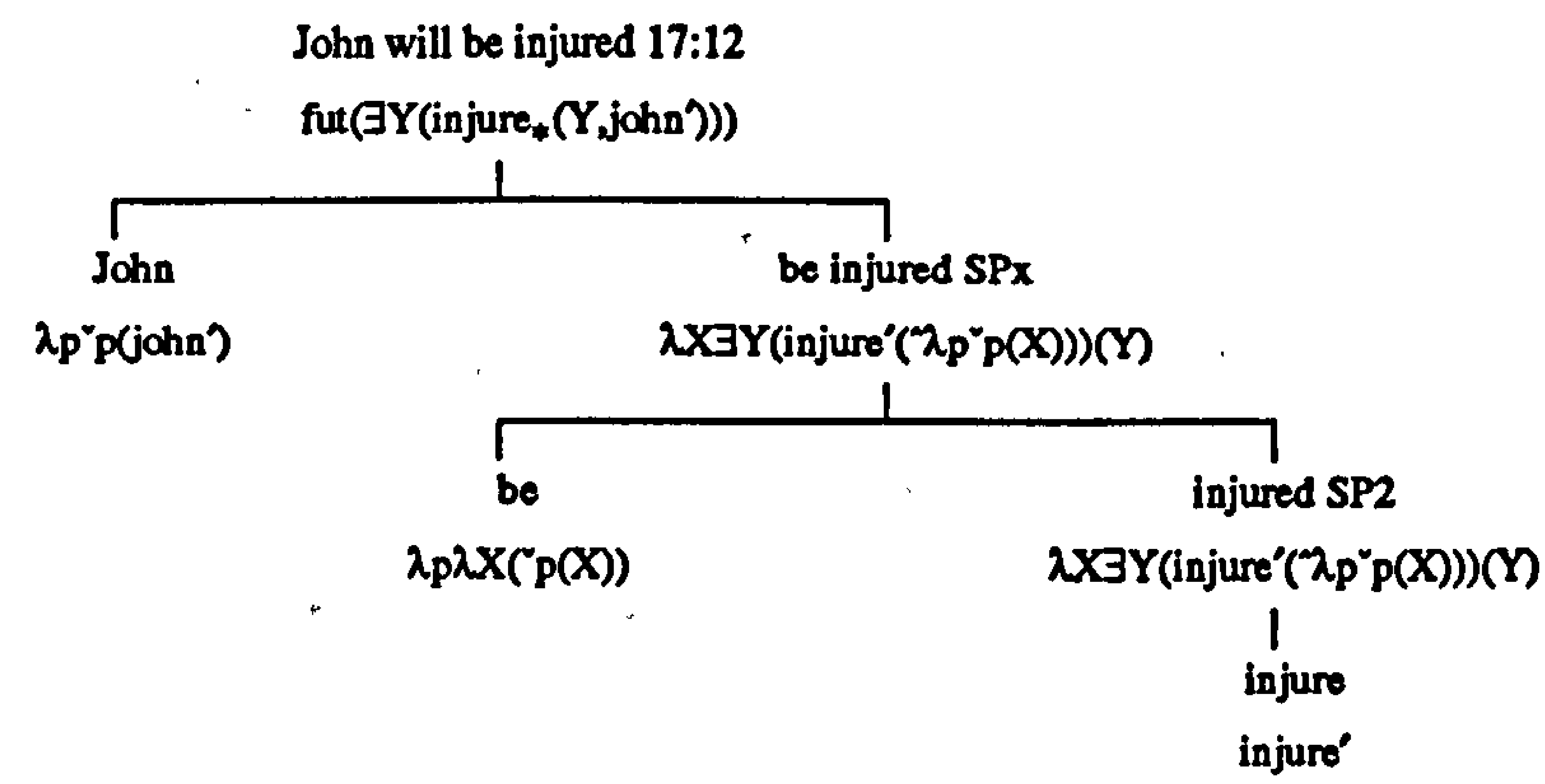


Fig 46

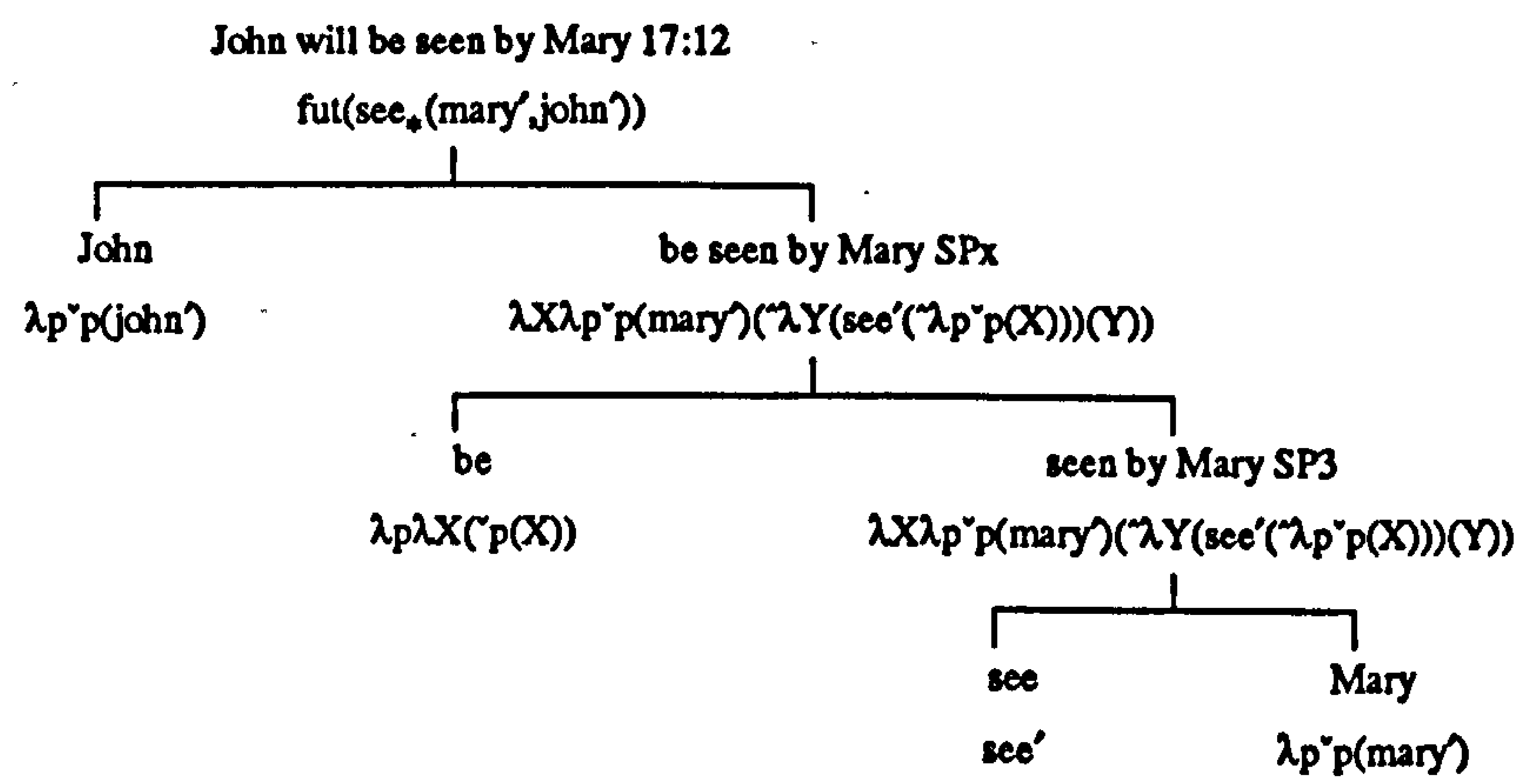


Fig 47

The conjunction test employed by Bach to illustrate the deficiency of a simple dichotomy of verb phrases into transitive and intransitive can also be used to demonstrate that the threefold classification into transitive, passive and intransitive verb phrases is not yet adequate. Arguably intransitive verb phrases commencing with a non finite auxiliary should not be conjoined:

(114) ?* John will [$_{IV}$ have run] and [$_{IV}$ be jumping].

although non finite intransitive verb phrases commencing with a *lexical* verb are undoubtedly conjoinable:

(115) John will [$_{LIV}$ kiss Lucy] and [$_{LIV}$ cuddle Mary].

If this intuition is correct, then the set of intransitive verb phrases must contain, as an identifiable sub set, members of a category LIV of lexical intransitive verb phrases. Notwithstanding such a subclassification, intransitive verb phrases of all varieties become conjoinable when finite:[†59]

(116) John [$_{FV}$ has kissed Lucy] and [$_{FV}$ is cuddling Mary].

Despite Bach's claim that the conjunction in (112) is between *transitive* verb phrases, the example cited could equally well be construed as involving a category of *passive transitive* verb phrases, as opposed to the *passive intransitive* conjuncts of example (113). The possibilities of adverbial attachment suggest to me that a class of passive transitive verb phrases should indeed be distinguished; for there can be no objection to either:

(117) John was [$_{PIV}$ [$_{PIV}$ kissed by Lucy frequently] and [$_{PIV}$ cuddled by Mary occasionally]].

or

(118) John was [$_{PIV}$ [$_{PTV}$ [$_{PTV}$ attacked in the park] and [$_{PTV}$ bitten on the leg]] by a vicious dog].

whereas adverbial qualification of an *active* transitive verb phrase appears to be inhibited:

(119) * John [$_{TV}$ [$_{TV}$ kissed in the park] and [$_{TV}$ cuddled on the sofa]] Lucy.

A system of categories including LIV $\subseteq$ IV, PIV, PTV and FV which accords with the above reflections will in due course be presented.

†59. This point is made explicitly by Bach in his later paper on tense and aspect, [B3].

5.2. Bach's Account of Tensed Verb Phrases and Auxiliaries

Montague's policy in PTQ is to interpret subject noun phrases as functions which take as their arguments the intensions of intransitive verb phrases. It has however been observed by Keenan and Faltz, [K6], that intransitive verb phrases vary in meaning with their subjects; thus "is running" is understood differently in:

(120) John is still running.

(121) The tap is still running.

(122) The play is still running.

Keenan and Faltz proceed to postulate a "functional principle (FP)":

> "The meaning we assign to expressions of functions may vary with (be conditioned by) the meaning we assign to expressions of their arguments."

which, in view of the examples above, would predict that it is the intransitive verb phrase which should be interpreted as the function with the interpretation of the subject noun phrase as argument.[†60]

The distinction between finite and non finite intransitive verb phrases, which I have advocated above, is utilised by Bach, [B3], in an attempt to formulate minimal modifications to the grammar of PTQ sufficient to guarantee conformity with Keenan and Faltz' functional principle. As Bach points out, there are no finite verb phrases in PTQ, since tense and aspect are introduced syncategorematically, hence the claim that such phrases, if introduced, should be of category t/T, with $f(t/T) = \langle\langle s, f(T)\rangle, t\rangle$, is not incompatible with the Montagovian orthodoxy that $f(IV) = \langle et\rangle$.[†61] Accordingly, in his conservative revision of PTQ, Bach introduces a new category of *tensed* intransitive verb phrases, The expressions of which are members of $P_{t/T}$. Semantically a tensed intransitive verb phrase is to be interpreted as a function from intensions of *terms* to truth values, ie:

$$f(t/T) = \langle\langle s\langle\langle s\langle et\rangle\rangle t\rangle\rangle t\rangle.$$

---

†60. Bach, [B3], produces an ingenious supporting argument based on phonology. In categorial terms, a phonological phrase is the maximal segment obtained by putting together functions with their right hand arguments; but apparently liason never occurs from a subject noun phrase to the (tensed) verb phrase.

†61. As always in this thesis, the formulations suppose the simplified Bennett system of typing.

Since subject noun phrases must now combine with members of $P_{t/T}$ and not members of P_{IV}, rule S4 must be replaced by the following:

(Ssp) If $\alpha \in P_T$ and $\gamma \in P_{t/T}$ then $f_{sp}(\alpha,\gamma) \in P_t$.

$f_{sp}(\alpha,\gamma) = NOM(\alpha)\ \gamma$[†62]

(Tsp) If $\alpha \rightsquigarrow \alpha'$ and $\gamma \rightsquigarrow \gamma'$ then $f_{sp}(\alpha,\gamma) \rightsquigarrow \gamma'(\hat{}\alpha')$.

The assumption behind this rule is that the verb phrase to be combined with the subject term will already have been made finite by the application of previous rules.

In Bach's conservative system, selected auxiliaries are allowed to appear in the lexicon as finite forms, thus a verb phrase commencing with an auxiliary cannot but be finite. However special provision must be made for inflecting verb phrases in which auxiliaries are *not* involved: hence Bach introduces a rule:

(Stn) If $\gamma \in P_{IV}$ then $f_{tn}(\gamma) \in P_{t/T}$.

$f_{tn}(\gamma)$ replaces all main verbs in γ with present tense singular forms.

(Ttn) If $\gamma \rightsquigarrow \gamma'$ then $f_{tn}(\gamma) \rightsquigarrow \lambda n(\check{}n)(\hat{}\gamma')$.

usage of which is illustrated in fig 48.

****

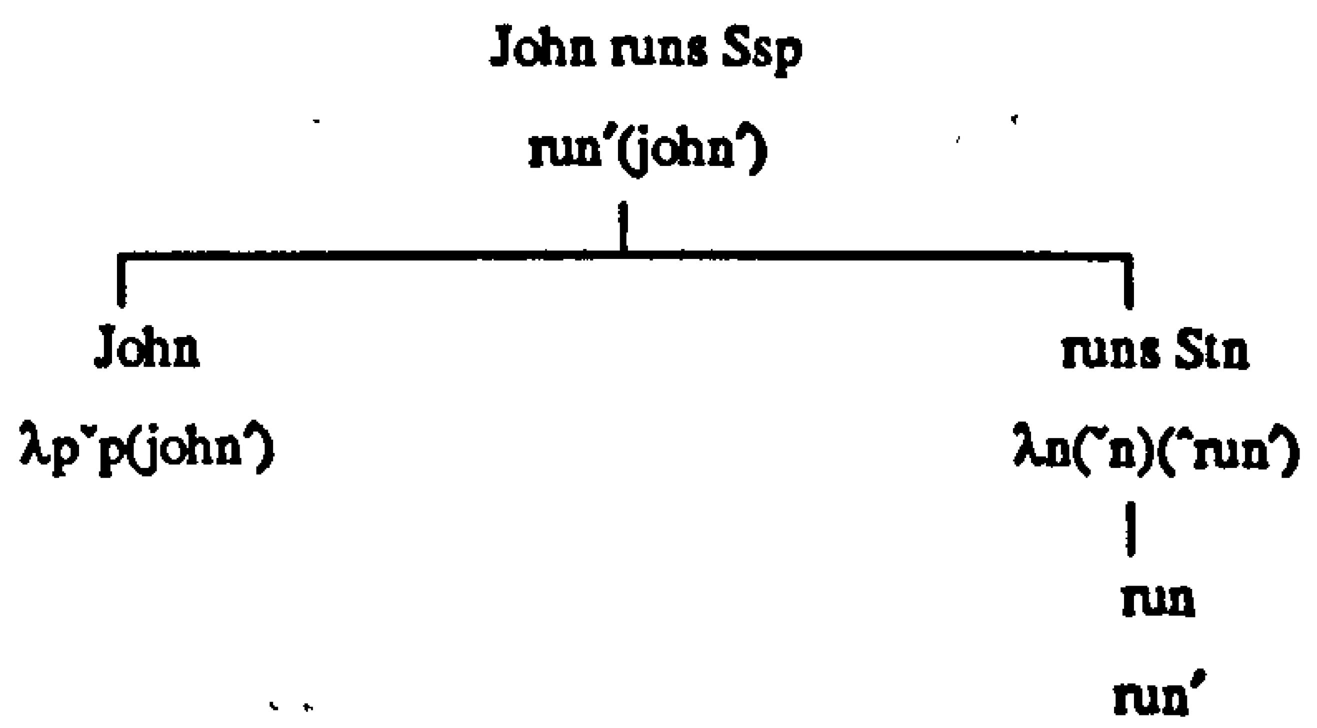


Fig 48

****

Prior to introducing auxiliary verbs, and in order to inhibit improper combinations, Bach subclassifies non finite intransitive verb phrases into *verbal* IVs and *copular* IVs. I shall refer to

†62. Bach adopts the interesting approach of introducing all syntactic variables in the *accusative* case, thus reversing Montague's practice. The result of NOM(α) will be α with all main terms of form him_n replaced by he_n. I have not employed Bach's own numbering system for his rules of tense, (Ssp is rule 106 in [B3]), since to do so might cause confusion with the rule indices of TMG.

expressions of these kinds as members of P_{VIV} and members of P_{CIV} and assume both that $P_{VIV} \subseteq P_{IV}$ and that $P_{CIV} \subseteq P_{IV}$.†63 Lexical intransitive verbs must now be assigned to B_{VIV}, while the category of transitive verb phrases becomes VIV/T. The copula is removed from the category of transitive verbs and assigned instead to $B_{CIV/T} = \{be\}$.

Auxiliaries are to accept complements of divers kinds and to make *finite* verb phrases directly. Of the auxiliaries which may find their complements in $P_{IV} = P_{VIV} \cup P_{CIV}$, Bach distinguishes those which accept an infinitival complement without modification, (Aux-m or true modals), from those which demand a participle form (Aux-h).†64 Two other classes of auxiliary are also recognised: Aux-d, the periphrastic "do", which takes complements only from P_{VIV}, and Aux-b which contains finite forms of "be" taking *terms* as complements. The full classification is illustrated in fig 49.

****

Auxiliaries: Bach's Classification				
Aux Category	Members	+VIV Complement	+CIV Complement	+T Complement
-m = (t/T)/IV	{will,wont}	John will speak	John will be a candidate	....
-h = (t/T)//IV	{has,hasn't}	John has spoken	John has been a candidate	....
-d = (t/T)/VIV	{doesn't}	John doesn't speak	....	....
-b = (t/T)/T	{is,isn't}	....	....	John is a candidate

Fig 49

****

The rules for introducing auxiliaries under this scheme are as follows:

(Scop) If $\gamma \in P_{CIV/T}$ and $\alpha \in PT$ then $f_{cop}(\gamma,\alpha) \in P_{CIV}$.

$f_{cop}(\eta,\alpha) = \gamma\, NOM(\alpha)$†65

(Tcop) If $\gamma \twoheadrightarrow \gamma'$ and $\alpha \twoheadrightarrow \alpha'$ then $f_{cop}(\gamma,\alpha) \twoheadrightarrow \gamma'(\hat{}\alpha')$.

(Saux-m) If $\gamma \in P_{Aux\text{-}m}$ and $\delta \in P_{IV}$ then $f_{aux\text{-}m}(\gamma,\delta) \in P_{t/T}$.

$f_{aux\text{-}m}(\gamma,\delta) = \gamma\,\delta$.

†63. Bach himself does not isolate P_{VIV} as a subset of P_{IV}, regarding the latter as the set of verbal intransitive verb phrases so that, on his account, $P_{CIV} \cap P_{IV} = \varnothing$. This usage however makes it impossible to state accurately the category of certain auxiliaries.

†64. Bach actually uses the code AUX-1, AUX-2, AUX-3 and AUX-4 rather than Aux-m, Aux-d, Aux-b, and Aux-h, but the grouping is perhaps misleading. The closest affinity is between AUX-1 and AUX-4.

†65. Bach defines $f_{cop}(\gamma,\alpha)$ to be equal to $\eta\ \alpha$. Where α is a syntactic variable this formulation gives incorrect results since in Bach's system such a variable is by default accusative.

(Taux-m) If $\gamma \Rightarrow \gamma'$ and $\delta \Rightarrow \delta'$ then $f_{aux\text{-}m}(\gamma,\delta) \Rightarrow \gamma'(\hat{}\delta')$.

(Saux-h) If $\gamma \in P_{Aux\text{-}h}$ and $\delta \in P_{IV}$ then $f_{aux\text{-}h}(\gamma,\delta) \in P_{t/T}$.

$f_{aux\text{-}h}(\gamma,\delta) = \gamma EN(\delta)$.

(Taux-h) If $\gamma \Rightarrow \gamma'$ and $\delta \Rightarrow \delta'$ then $f_{aux\text{-}h}(\gamma,\delta) \Rightarrow \gamma'(\hat{}\delta')$.

(Saux-d) If $\gamma \in P_{Aux\text{-}d}$ and $\delta \in P_{VIV}$ then $f_{aux\text{-}d}(\gamma,\delta) \in P_{t/T}$.

$f_{aux\text{-}d}(\gamma,\delta) = \gamma\delta$.

(Taux-d) If $\gamma \Rightarrow \gamma'$ and $\delta \Rightarrow \delta'$ then $f_{aux\text{-}d}(\gamma,\delta) \Rightarrow \gamma'(\hat{}\delta')$.

(Saux-b) If $\gamma \in P_{Aux\text{-}b}$ and $\alpha \in P_T$ then $f_{aux\text{-}b}(\gamma,\alpha) \in P_{t/T}$.

$f_{aux\text{-}b}(\gamma,\alpha) = \gamma\alpha$.

(Taux-b) If $\gamma \Rightarrow \gamma'$ and $\alpha \Rightarrow \alpha'$ then $f_{aux\text{-}b}(\gamma,\alpha) \Rightarrow \gamma'(\hat{}\alpha')$.

As translations for auxiliary verbs Bach makes suggestions which amount to:

will $\Rightarrow$ $\lambda p\lambda n fut((\check{}n)(p))$

wont $\Rightarrow$ $\lambda p\lambda n \neg fut((\check{}n)(p))$

doesn't $\Rightarrow$ $\lambda p\lambda n \neg(\check{}n)(p)$

has $\Rightarrow$ $\lambda p\lambda n past((\check{}n)(p))$

hasn't $\Rightarrow$ $\lambda p\lambda n \neg past((\check{}n)(p))$

is $\Rightarrow$ $\lambda n\lambda m(\check{}m)(\hat{}\lambda X(\check{}n)(\hat{}\lambda Y[X=Y]))$

isn't $\Rightarrow$ $\lambda n\lambda m \neg(\check{}m)(\hat{}\lambda X(\check{}n)(\hat{}\lambda Y[X=Y]))$

Thus the effects of the proposed rules may be illustrated as in figs 50...52.

Upon investigation a number of infelicities become apparent in the account of auxiliaries provided in Bach's conservative revision of PTQ. In the first place the account is incomplete since it makes no provision for the modal, passive and progressive variations of "be", nor is it obvious how such additions would be accommodated. Presumably {is,isn't} would have to appear in the lexicon four times iff all auxiliaries were to be held as finite forms.

As regards the passive "be", Bach's earlier treatment introduced it as a *non* finite form, thus for con-

****

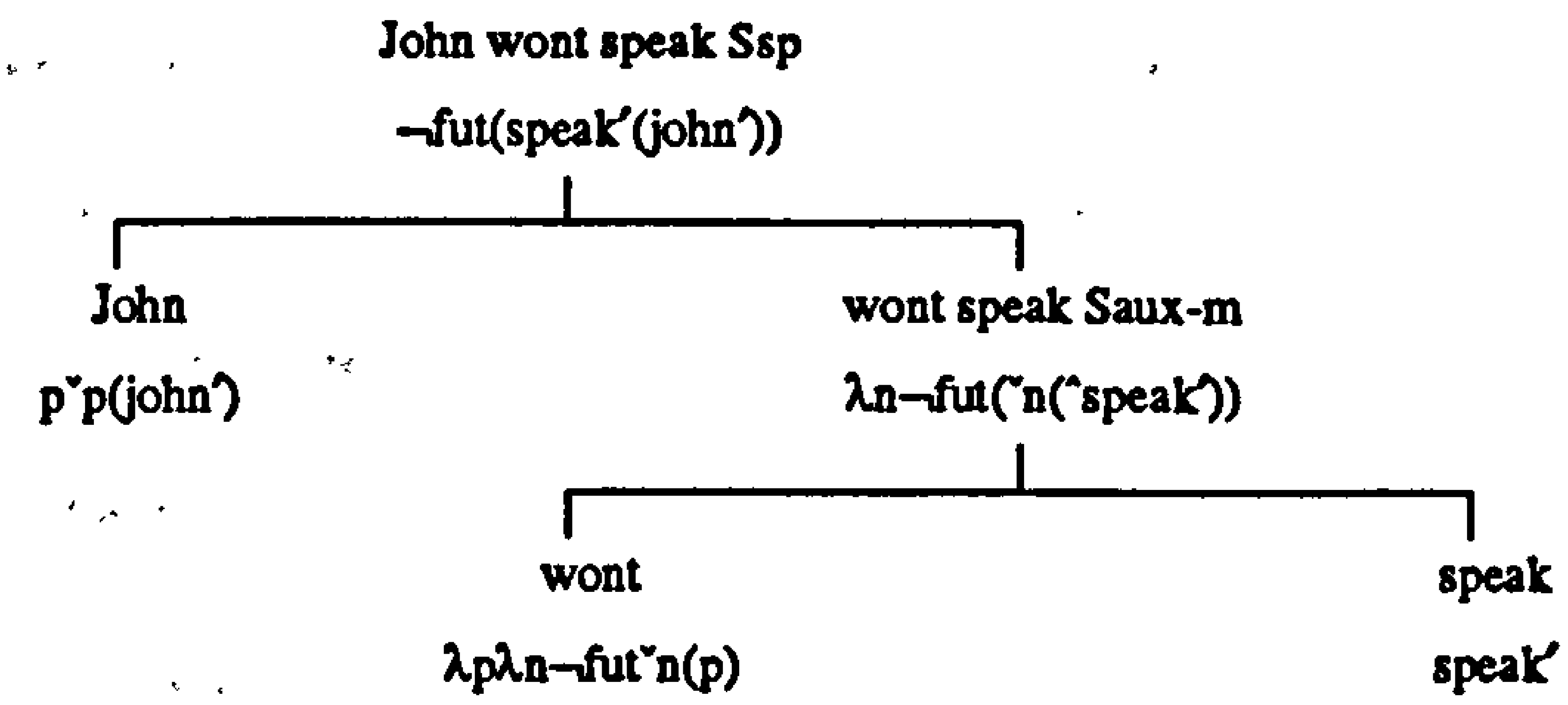


Fig 50

****

****

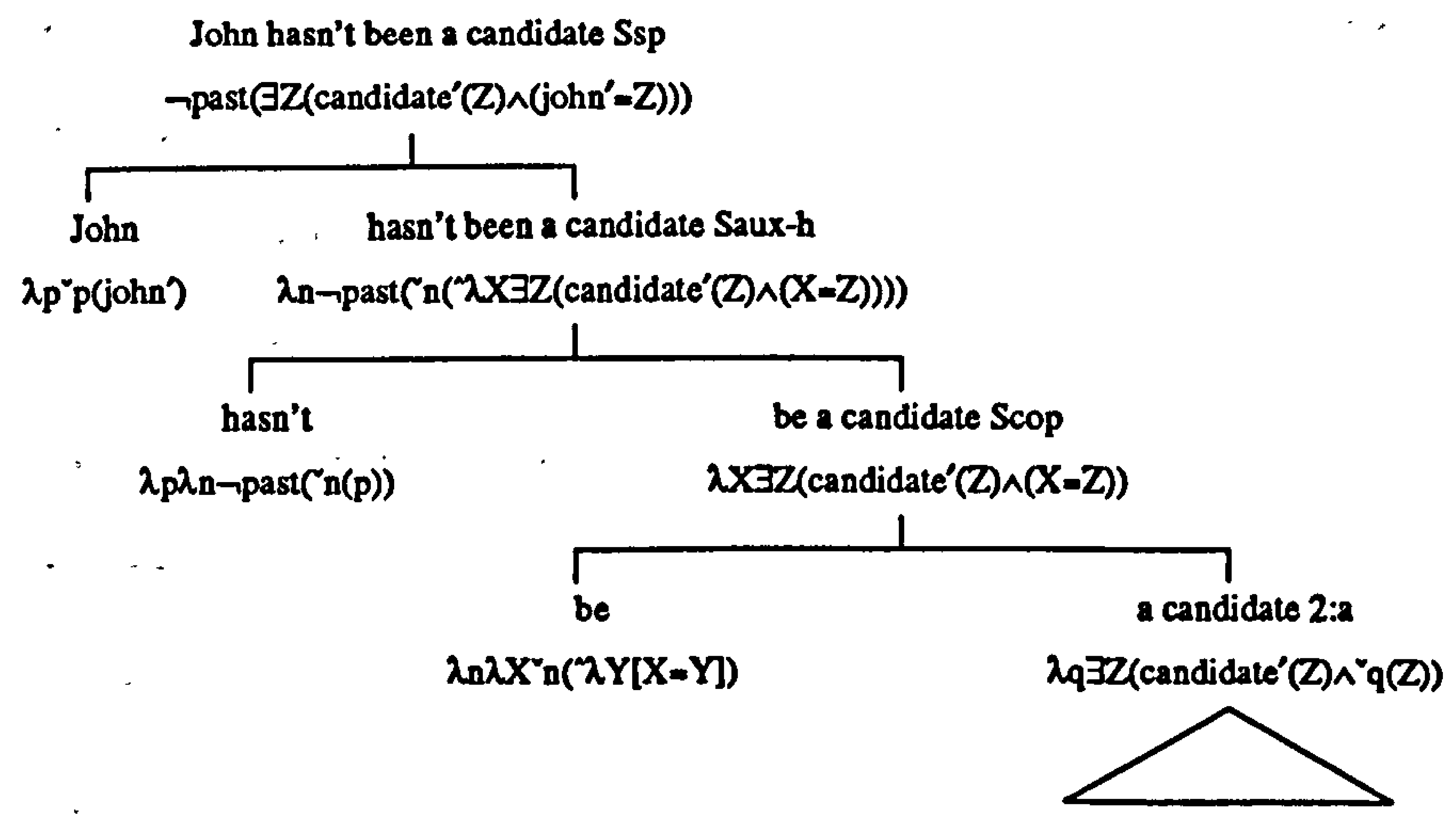


Fig 51

****

****

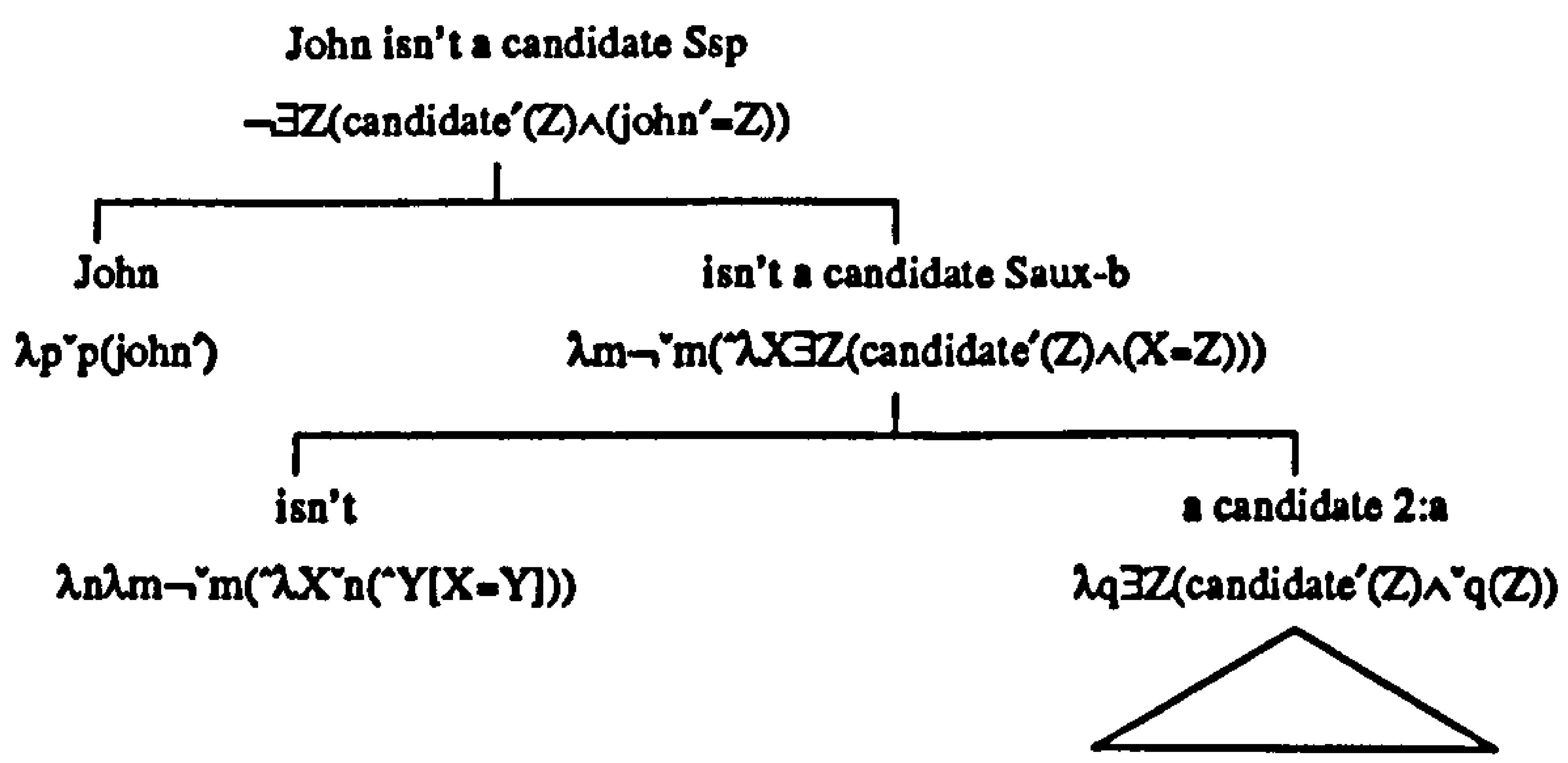


Fig 52

****

sistency some modification would be required before the two accounts could be integrated. The position with the copula is however even more curious: both finite and non finite forms are recorded in the lexicon, but with no apparent relationship between them. Comparison of the derivations in figs 51 and 52 reveals that "isn't a candidate" has not been derived from "be a candidate", the affinities in the translations of the top nodes of the respective trees being purely coincidental.

It is surely anomalous that tensed forms for lexical verbs should be introduced explicitly by rule Stn while auxiliaries are finite by default: moreover the absence of non finite forms for the auxiliaries makes auxiliary chaining impossible.

When exploring the possibility of a more radical revision of PTQ, Bach implies that in order to introduce auxiliary chaining it is necessary to map *all* intransitive verbs onto type <<s,f(T)>,t>, thus replacing the category IV by a new category t//T. Auxiliaries, given this amendment, would be assigned to category (t//T)/(t//T) corresponding to semantic type <<s,f(t//T)>,f(t//T)>.

Bach does not illustrate the translation of intransitive verbs implied by the later revision, but presumably we would require:

walk ⇝ $\lambda n walk'(n)$

where, as Bach suggests, walk′ denotes not a set as heretofore but a function from the intensions of sets of properties to truth values. The proposed translations for modal "will" and perfect "have" under the modified scheme become:

will ⇝ $\lambda f will(f)$

have ⇝ $\lambda f past(f)$

where $f = v_{0<s<<s<<s<et>>t>>t>>}$.

A rule for combining auxiliaries and intransitive verb phrases must (presumably) take the form:

(Sau) If $\gamma \in P_{(t//T)/(t//T)}$ and $\delta \in P_{t//T}$ then $f_{au}(\gamma,\delta) \in P_{t//T}$.

$f_{au}(\gamma,\delta) = \gamma\,\theta$ where $\theta = \delta$ with the appropriate inflection.

(Tau) If γ ⇝ γ' and δ ⇝ δ' then $f_{au}(\gamma,\delta)$ ⇝ $\lambda n \gamma'(\hat{}\delta'[n])$.

Thus equipped, we may illustrate the derivation of a sentence with chained auxiliaries such as "John will have walked", by a tree like that of fig 53.

****

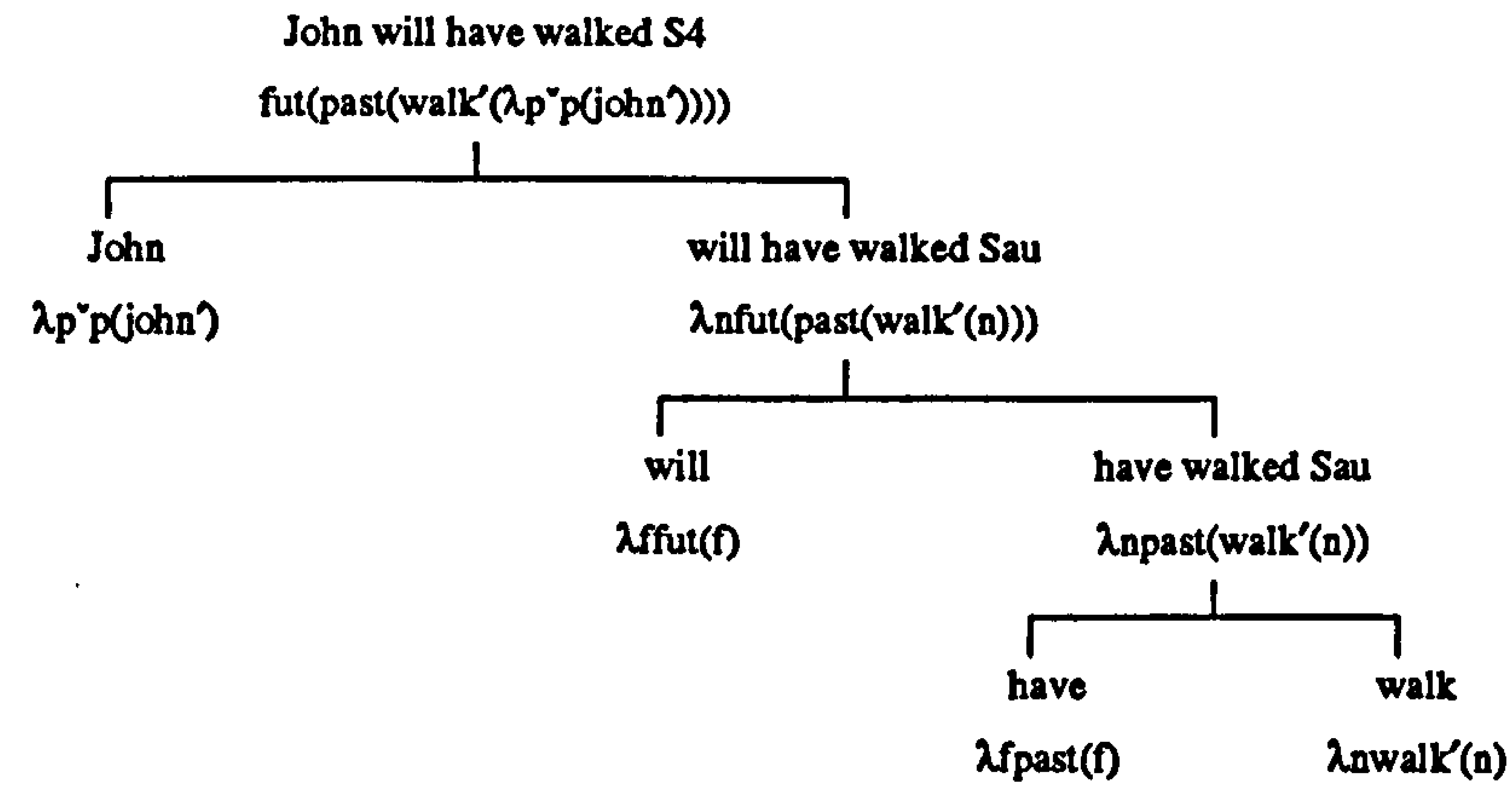


Fig 53

****

The problem now, even disregarding the cryptic translation which emerges at the top node, is that the distinction between finite and non finite verb phrases has been lost altogether and S4 in its original form resurrected. Tensed verb phrases were originally introduced by Bach in the context of converting intransitive verb phrases into a functional form, but such conversion has become otiose: indeed the distinction between the categories t/T and t//T serves no purpose.

In TMG a tactic is adopted which permits auxiliary chaining without violence to Montague's original categorial framework, but before developing this further we shall consider Dowty's treatment of the topic under discussion.

5.3. Dowty's Two Dimensional System

In *Tenses, Time Adverbials and Compositional Semantic Theory*, [D8], Dowty presents an analysis which is of interest as much for its identification of certain key problems to be solved by any adequate account of tense and aspect as for the specific partial solutions suggested. The following sample sentences, most of which are adaptations of Dowty's own examples, will serve to illustrate the nature of these problems.

(123) John left yesterday.

(124) John saw three ships come sailing by on Christmas day in the morning.

(125) A competitor who panicked cheated.

(126) A competitor who triumphed cheated.

(127) John has married a girl who is a student.

(128) John will marry a girl who is a student.

(129) A child who will be a king was born.

(130) A child who would be a king was born.

As is apparent from example (123), time references may be polymodal, ie. they may not only be latent in the choice of tense but also explicit in the employment of a time adverbial: moreover, as in sentence (124), time adverbials may themselves be nested. Such considerations vitiate any attempt to generalise upon the tactics adopted in IL, (ILt10), where "past" and "fut" are introduced as *Priorian*†66 tense operators, ie. operators which merely shift the time of evaluation of the governed sentence along a linear axis. If the interpretation of "past" is defined as in (ILt10) and "yesterday′" is treated analogously as an operator which moves the moment of evaluation to a point in the day preceding the current moment, ie. if:

$$[\![\ulcorner \text{yesterday}'(\Phi) \urcorner]\!]^{M,w,t,g} = 1 \text{ iff}$$

$$[\![\Phi]\!]^{M,w,t',g} = 1 \text{ for some } t' \in T \text{ falling in the day previous to that containing } t.$$

then:

past(yesterday′(leave′(john′)))

and:

yesterday′(past(leave′(john′)))

will result in evaluation of the embedded sentence on the day before some arbitrary moment in the past and some moment in the past *before* yesterday respectively.

Examples (125) ... (130) all relate to sequence of tense phenomena. Where one past tense is embedded within another, as in examples (125) and (126), then the events reported are not essentially ordered with respect to each other. Our gratuitous assumption that these sentences imply a causal connection,

†66. After Arthur Prior who first experimented with operators which shifted the moment of evaluation of the argument sentence, [P8,P9].

together with our extra linguistic knowledge, doubtless leads us to expect that in (125) the cheating followed the panicking whilst in (126) the cheating came first; but the tensed construction which is common to both is neutral as regards the ordering.

Although there is no ambiguity in sentence (127), where the girl in question is plainly implied to be a student at the time of *utterance*, sentence (128) is temporally ambiguous in so far that it is unclear whether the girl is a student *now*, (but may not be so on her wedding day), or whether she is predicted to be a student bride whatever her present occupation. An ambiguous temporal ordering must accordingly be allowed to arise when a present tense becomes embedded in a future matrix clause.

Likewise there is no ordering ambiguity in example (129), where the birth must be before and the coronation subsequent to the time of utterance, but an element of free ordering must be admitted in case (130). Unlike examples (125) and (126), sentence (139) does not permit free ordering between the reported events, for the coronation must be subsequent to the birth, but there *is* now a free ordering of the coronation with respect to the time of utterance.

In order to accommodate these phenomena Dowty, following a suggestion by Kamp, [K1], from whom examples (129) and (130) emanate, proposes to evaluate expressions involving temporal references relative to a *pair* of time indices <i,j>, where j is the time of utterance or "speech" time and i the time of occurrence or "valuation time".†67 Accordingly he first engineers certain modifications to the language IL of intensional logic which is to serve as intermediary for semantic interpretation.

5.3.1. Dowty's Modifications to IL

As in an earlier treatment of the language of intensional logic, [D6], Dowty expands the set *Type* of semantic types to contain a new primitive type *i*, the type of *intervals of time*. The recursive clauses in the definition of *Type* remain as before; but the lexicon will now contain the new sets:

$Var_i = \{v_{n,i}: n \geq 0\}.$

$Con_i = \{c_{n,i}: n \geq 0\}.$

†67. Dowty uses the term "reference time" not "valuation time", but in my view this usage is confusing since "reference time" already has connotations associated with the Reichenbachian theory to be discussed later.

By convention $v_{0,i}$, $v_{1,i}$, $v_{2,i}$ ... may be abbreviated to t, t′, t″ ..., while $c_{0,i}$ is written t* and will be interpreted at any interval of evaluation as denoting the interval itself.

The set *Int* of intervals of time is defined as the smallest set

$$Int \subset Power(T)$$

such that if $\mathbf{i} \in Int$ then for all $\mathbf{m}, \mathbf{m}', \mathbf{m}'' \in T$, if $\mathbf{m} \in \mathbf{i}$ and $\mathbf{m}'' \in \mathbf{i}$ and $\mathbf{m}<\mathbf{m}'<\mathbf{m}''$ then $\mathbf{m}' \in \mathbf{i}$.

We may next define

$$den(i,M) = Int.$$

ie. the possible denotations of type *i* are to be intervals or sets of contiguous moments. If we now redefine:

$$den(<s,a>,M) = den(a,M)^{W \times Int}.$$

and assume that i<i′ iff some moment in i′ succeeds all moments in i, and that i>i′ iff some moment in i precedes all moments in i′, then the original semantics for IL may be preserved by substituting interval j for moment t throughout the secondary valuation, ie:

$$[\![\alpha]\!]^{M,w,t,g} \text{ becomes } [\![\alpha]\!]^{M,w,j,g}$$

A new secondary valuation function *V′* must however be introduced for the two dimensional system such that:

$$V'(\alpha,M,w,<i,j>,g) = [\![\![\alpha]\!]\!]^{M,w,<i,j>,g}.$$

gives the extension of *a* at *w* relative to the pair *<i,j>* of intervals and with respect to *M*. To avoid excessive superscription, and where no ambiguity arises we may conveniently abbreviate so that:

$$[\![\![\alpha]\!]\!]^{M,w,<i,j>,g} \text{ becomes } [\![\![\alpha]\!]\!]^{i,j} \text{ and } [\![\alpha]\!]^{M,w,j,g} \text{ becomes } [\![\alpha]\!]^{j}.$$

Unless otherwise stated the "speech time" index *j* becomes vacuous, ie. we require a default according to which:

(Default) $[\![\![\alpha]\!]\!]^{i,j} = [\![\alpha]\!]^{i}$†68

but for expressions involving temporal references the following clauses are introduced by Dowty:

†68. If $\alpha \in Con_a$ then $[\![\![\alpha]\!]\!]^{M,w,<i,j>,g} = I(\alpha)(w,i)$. A sentence ϕ is to be true at speech point *j* iff, for some *i*, $V'(\phi, M, w, <i,j>, g) = 1$ ie. $[\![\phi]\!]^{j} = 1$ iff $\exists i [\![\![\phi]\!]\!]^{i,j} = 1$.

$[\![\text{present}(\phi)]\!]^{i,j} = 1$ iff $[\![\phi]\!]^{i,j} = 1$ and $i = j$.

$[\![\text{past}(\phi)]\!]^{i,j} = 1$ iff $[\![\phi]\!]^{i,j} = 1$ and $i < j$.

$[\![\text{perfect}(\phi)]\!]^{i,j} = 1$ iff $[\![\phi]\!]^{i',j} = 1$ for some i' of which i is a final subinterval.

$[\![\text{fut}(\phi)]\!]^{i,j} = 1$ iff $[\![\phi]\!]^{i',i'} = 1$ for some $i' > j$.

$[\![\text{would}(\phi)]\!]^{i,j} = 1$ iff $[\![\phi]\!]^{i',j} = 1$ for some $i' > i$.

$[\![\text{at}(\tau,\phi)]\!]^{i,j} = 1$ iff $[\![\phi]\!]^{i',j} = 1$ where $i' = [\![\tau]\!]^{i,j}$.

$[\![t^*]\!]^{i,j} = i$.

According to these definitions both the "present" and "past" operators have the effect of asserting that a given ordering relationship holds between speech and valuation times although the "perfect" operator retains the Priorian characteristic in so far that it shifts the valuation time of the embedded sentence to a new interval of which the original valuation time is a final subinterval. The "fut" operator in effect substitutes the valuation time of the tensed sentence for the speech time of the governed sentence, while the "would" operator shifts the valuation time of the embedded sentence in Priorian fashion to a time later than the original valuation time but otherwise unordered with respect to the speech time. As forecast, t* is defined so as to denote the current valuation time; furthermore we have a new "at" operator which repositions the valuation time of the argument sentence at τ.

5.3.2. Dowty's Grammar Rules for Tense and Aspect.

Like Bach, Dowty recognises a category t/T of tensed verb phrases, but he introduces in addition a category:

TMADV = t(t/i) such that f(TMADV) = <<it>t>.

Expressions in P_{TMADV} are to be temporal adverbials, thus basic time adverbs are members of B_{TMADV}. Possible denotations for temporal adverbials may be identified thus:

$$den(<<it>t>,M) = \{0,1\}^{\left(\{0,1\}^{Int}\right)}$$

so if $\delta \in B_{TMADV}$ then $\delta \twoheadrightarrow \delta' \in Con_{<<it>t>}$ and:

$[\![\delta']\!]^{i,j} \in \{0,1\}^{(\{0,1\}^{Int})}$.

ie. the extension of δ′ is a function from sets of intervals to truth values, or equivalently a set of sets of intervals. In particular Dowty requires:

$[\![yesterday']\!]^{i,j}$ = the set of all sets containing an interval within the day preceding the day containing the speech time *j*.

and

$[\![now']\!]^{i,j}$ = the set of all sets containing *j*.

thus these adverbials become indexical in nature, relating an event directly to the speech time.

The argument for a temporal adverbial is to take the form $\lambda t(t = t^* \wedge \phi)$ hence an expression of the form $\alpha'(\lambda t(t = t^* \wedge \phi))$ will be true iff α′ contains an interval t which is identical with the valuation time at which φ is true. Hence time adverbials, unlike Priorian operators, assert that the valuation time has a particular location.

****

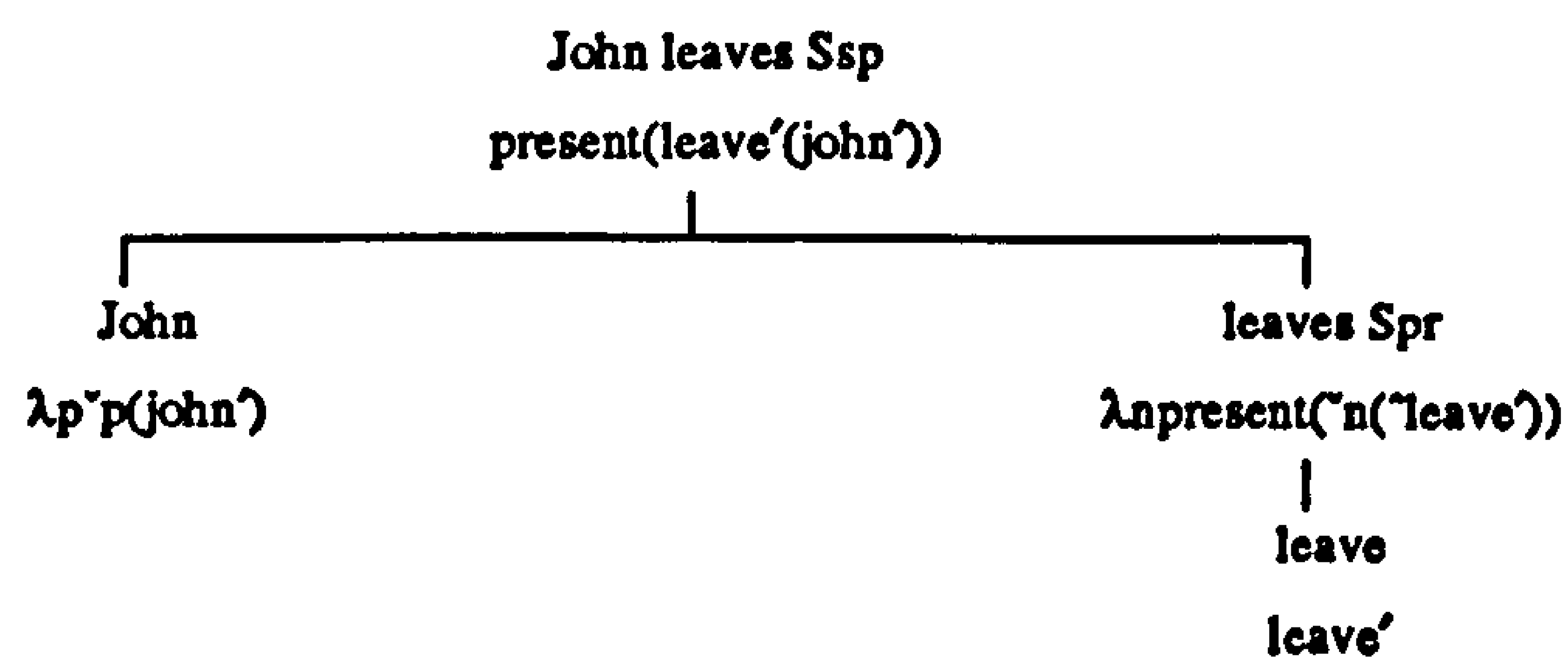


Fig 54

****

These preliminaries having been completed, we may formulate Dowty's proposed rules for tense and aspect as follows:

(Spr) If $\delta \in P_{IV}$ then $f_{pr}(\delta) \in P_{t/T}$.

$f_{pr}(\delta)$ = the third person singular present of δ.

(Tpr) If $\delta \leadsto \delta'$ then $f_{pr}(\delta) \leadsto \lambda n(present(\check{}n(\hat{}\delta)))$.

(Spst) If $\delta \in P_{IV}$ then $f_{pst}(\delta) \in P_{t/T}$.

$f_{pst}(\delta)$ = the simple past tense of δ.

****

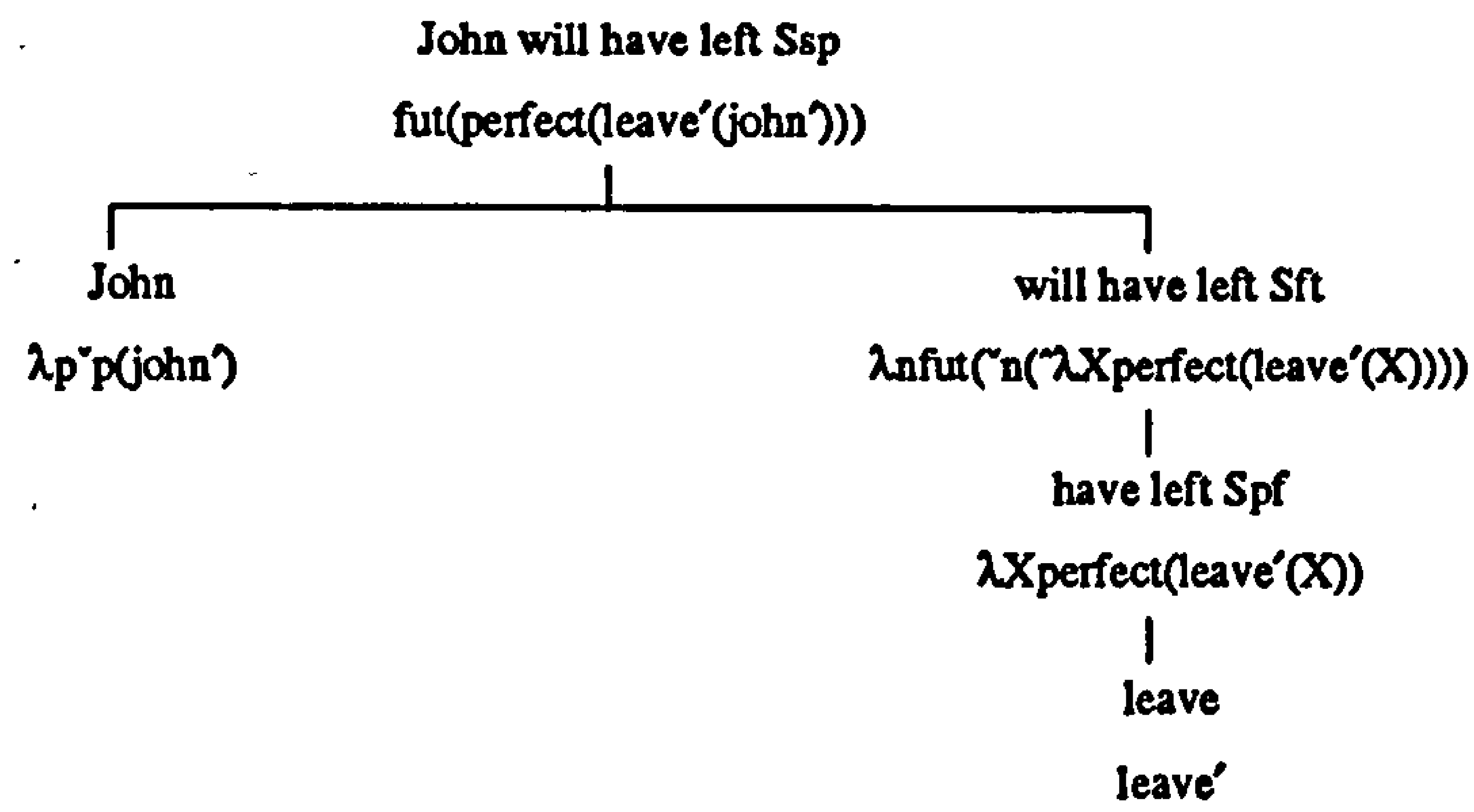


Fig 55

****

(Tpst) If $\delta \rightsquigarrow \delta'$ then $f_{pst}(\delta) \rightsquigarrow \lambda n(past(\check{}n(\hat{}\delta)))$.

(Sft) If $\delta \in P_{IV}$ then $f_{ft}(\delta) \in P_{t/T}$.

$f_{ft}(\delta) =$ will δ.

(Tft) If $\delta \rightsquigarrow \delta'$ then $f_{ft}(\delta) \rightsquigarrow \lambda n(fut(\check{}n(\hat{}\delta)))$.

(Swd) If $\delta \in P_{IV}$ then $f_{wd}(\delta) \in P_{t/T}$.

$f_{wd}(\delta) =$ would δ.

(Twd) If $\delta \rightsquigarrow \delta'$ then $f_{wd}(\delta) \rightsquigarrow \lambda n(would(\check{}n(\hat{}\delta)))$.

(Spf) If $\delta \in P_{IV}$ then $f_{pf}(\delta) \in P_{IV}$.

$f_{pf}(\delta) =$ have EN(δ).

(Twd) If $\delta \rightsquigarrow \delta'$ then $f_{pf}(\delta) \rightsquigarrow \lambda X(perfect(\delta'(X)))$.

(Stav) If $\alpha \in P_{TMAV}$ and $\phi \in P_t$ then $f_{tav}(\alpha,\phi) \in P_t$.

$f_{tav}(\alpha,\phi) = \phi\ \alpha$.

(Ttav) If $\alpha \rightsquigarrow \alpha'$ and $\phi \rightsquigarrow \phi'$ then $f_{tav}(\alpha,\phi) \rightsquigarrow \alpha'(\lambda t(t{=}t^{*} \wedge \phi'))$.

The operation of these rules in the simplest cases is illustrated in figs 54 and 55.

5.3.3. The Limitations of Two Dimensionality

Rule (Stav) combines a temporal adverbial with a sentence to make another sentence and may accordingly be reapplied to its own output as illustrated in fig 56. Thus given Dowty's semantics for time adverbials, a satisfactory treatment for sentences like (123) and (124) is immediately forthcoming, for:

****

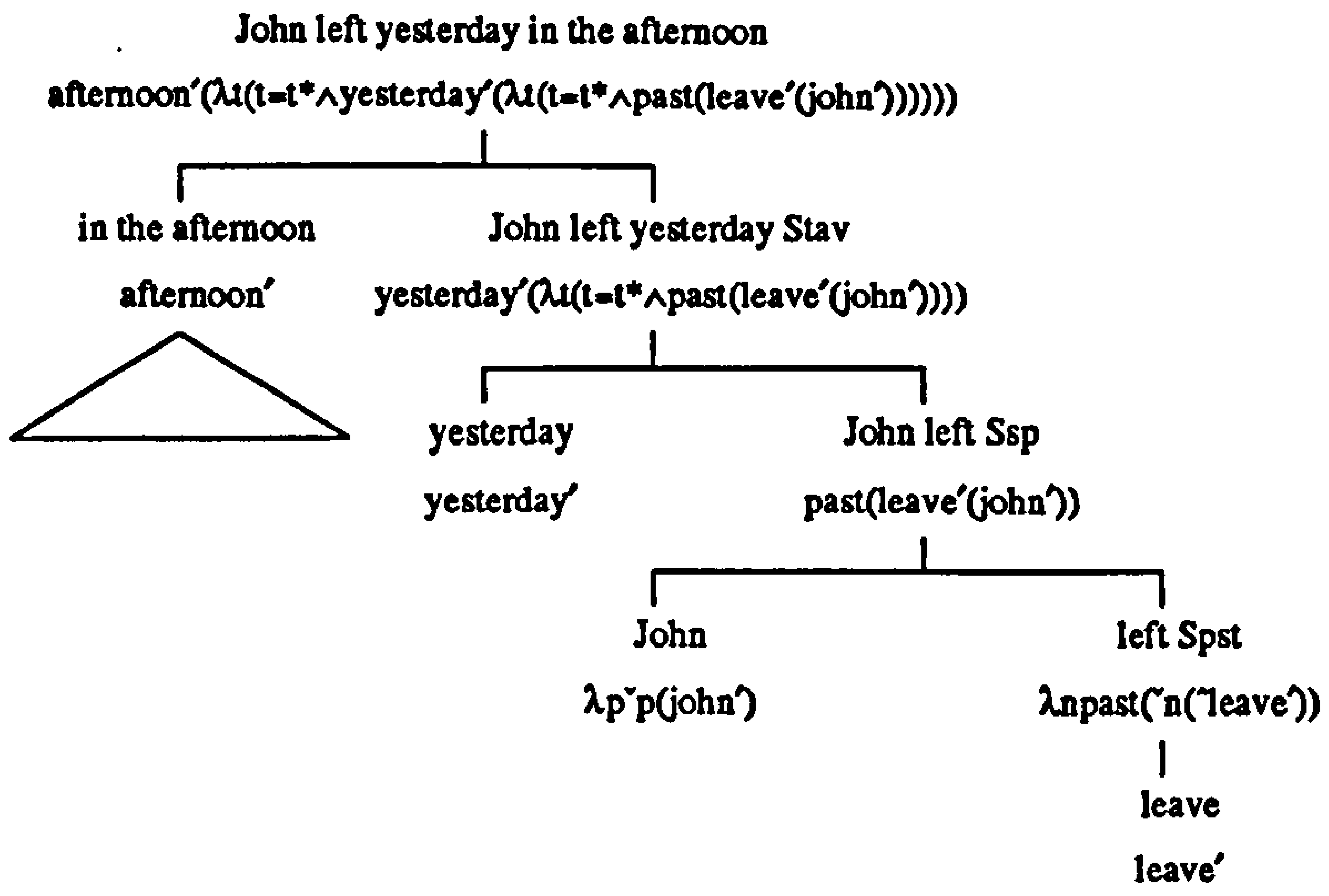


Fig 56

****

(131) $afternoon'(\lambda t(t=t^{*}\wedge yesterday'(\lambda t(t=t^{*}\wedge past(leave'(john'))))))$

gets the value 1 iff leave′(john′) is true for some pair $\langle i,j\rangle$ with $i<j$ and {i} is a member of the sets assigned to both yesterday′ and afternoon′.†69 The manoeuvre of applying temporal adverbials directly to sentences does however inhibit analysis of sentences like:

(132) John arrived yesterday and will leave tomorrow.

where it appears to be necessary to introduce the time adverb at finite verb phrase level before the conjunction. In TMG we shall adopt a modification derived ultimately from Dowty's analysis which renders examples such as (132) tractable.

Prior to considering any sequence of tense cases involving relative clauses, Dowty introduces a modification to the translation rule (T3) corresponding to Montague's syntax rule (S3):

(T3′) $f_{3,n}(\zeta,\phi) \leadsto \lambda x_n(\zeta'(x_n)\wedge\exists t(at(t,\phi')))$.

Use of this modified rule is illustrated in fig 57, which corresponds to example (125) on the analysis

---

†69. This mode of analysis would not of course be applicable to a sentence like :

- John left a fortnight ago yesterday.

where "a fortnight ago" must be measured *from* yesterday.

whereby the object noun phrase is *not* introduced by quantification.

****

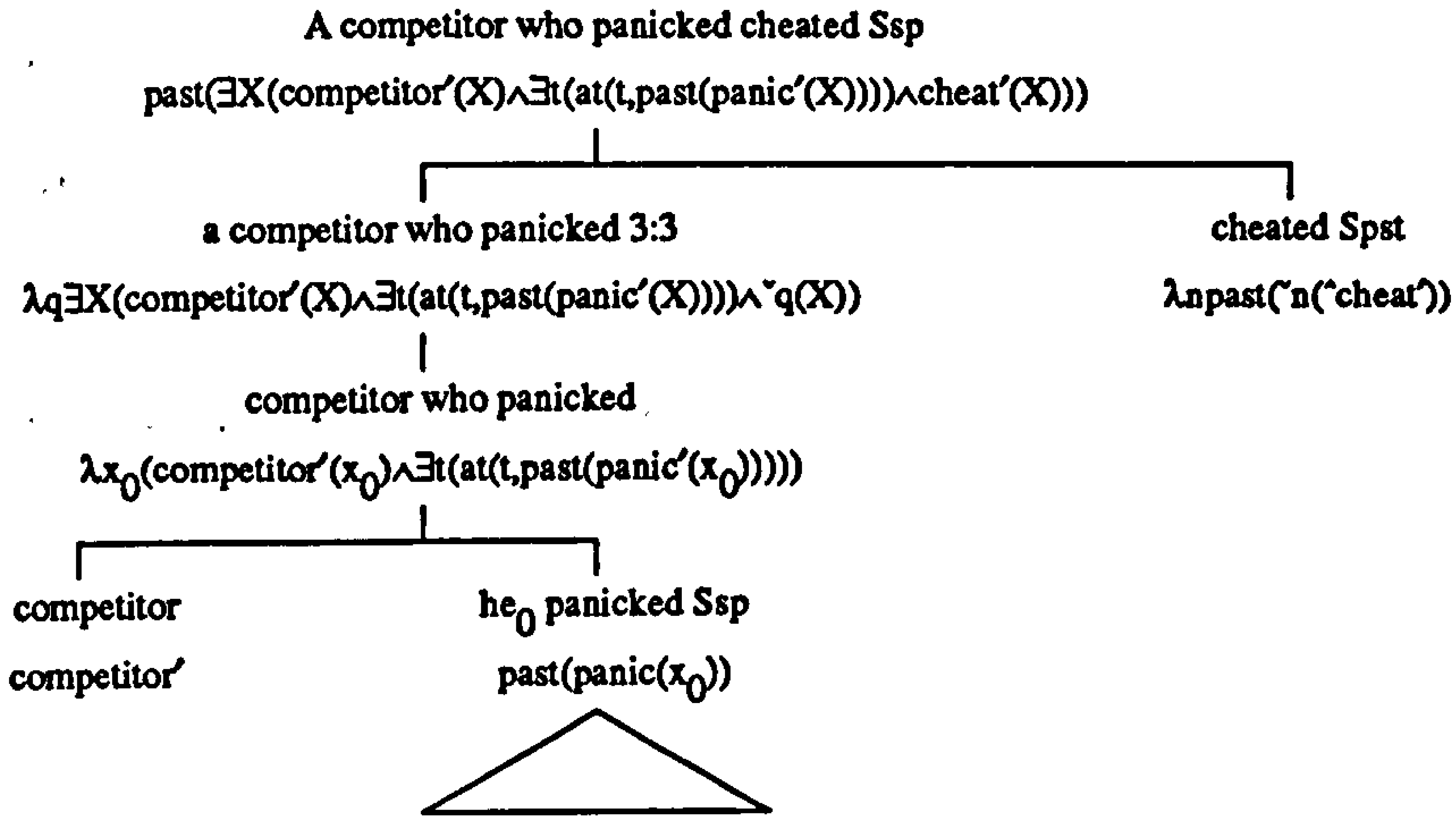


Fig 57

****

Inspection reveals that the panicking and cheating are indeed unordered for:[†70]

(133) $\llbracket \text{past}(\exists X(\text{candidate}'(X)\wedge\exists t(\text{at}(t,\text{past}(\text{panic}'(X))))\wedge\text{cheat}'(X))) \rrbracket^{i,j} = 1$ iff $i<j$ and

$\llbracket \exists X(\text{candidate}'(X)\wedge\exists t(\text{at}(t,\text{past}(\text{panic}'(X))))\wedge\text{cheat}'(X)) \rrbracket^{i,j} = 1$ iff

{for some $i' = \llbracket t \rrbracket^{i,j}$

$\llbracket \text{past}(\text{panic}'(X)) \rrbracket^{i',j} = 1$ iff $i'<j$ and

$\llbracket \text{panic}'(X) \rrbracket^{i',j} = 1$} and

$\llbracket \text{cheat}'(X) \rrbracket^{i,j} = 1$.

Plainly both i and i′ are ordered with respect to j, but *not* with respect to each other; moreover a similar result is obtained from the alternative analysis whereby the object noun phrase is introduced by quantification:

(134) $\llbracket \exists X(\text{candidate}'(X)\exists t(\text{at}(t,\text{past}(\text{panic}'(X))))\wedge\text{past}(\text{cheat}'(X))) \rrbracket^{i,j} = 1$ iff

{for some $i' = \llbracket t \rrbracket^{i,j}$

$\llbracket \text{past}(\text{panic}'(X)) \rrbracket^{i',j} = 1$ iff $i'<j$ and

$\llbracket \text{panic}'(X) \rrbracket^{i',j} = 1$} and

{ $\llbracket \text{past}(\text{cheat}'(X)) \rrbracket^{i,j} = 1$ iff $i<j$ and

†70. In this and subsequent derivations only stages relevant to the problem in hand are listed.

$[\![\text{cheat}'(X)]\!]^{i,j} = 1\}$.

Uniform results, with the girl in question determined unequivocally to be a student at the time of utterance, are likewise generated for example (127), where the choices are:

(135) $[\![\text{present(perfect}(\exists X(\text{girl}'(X) \wedge \exists t(\text{at}(t,\text{present}(\text{student}'(X)))) \wedge \text{marry}_*(\text{john}',X))))]\!]^{i,j} = 1$ iff $i=j$ and

$[\![\text{perfect}(\exists X(\text{girl}'(X) \wedge \exists t(\text{at}(t,\text{present}(\text{student}'(X)))) \wedge \text{marry}_*(\text{john}',X))]\!]^{i,j} = 1$ iff

for some i' such that i is a final subinterval of i'

$\exists X(\text{girl}'(X) \wedge \exists t(\text{at}(t,\text{present}(\text{student}'(X)))) \wedge \text{marry}_*(\text{john}',X))]\!]^{i',j} = 1$ iff

$\{$for some $i'' = [\![t]\!]^{i',j}$

$[\![\text{present}(\text{student}'(X))]\!]^{i'',j} = 1$ iff $i'' = j$ and

$[\![\text{student}'(X)]\!]^{i'',j} = 1\}$ and

$[\![\text{marry}_*(\text{john}',X)]\!]^{i',j} = 1$

or alternatively, if the object noun phrase is quantified in:

(136) $[\![\exists X(\text{girl}'(X) \wedge \exists t(\text{at}(t,\text{present}(\text{student}'(X)))) \wedge \text{present}(\text{perfect}(\text{marry}_*(\text{john}',X))))]\!]^{i,j} = 1$ iff

$\{$for some $i' = [\![t]\!]^{i,j}$

$[\![\text{present}(\text{student}'(X))]\!]^{i',j} = 1$ iff $i' = j$ and

$[\![\text{student}'(X)]\!]^{i',j} = 1\}$ and

$\{ [\![\text{present}(\text{perfect}(\text{marry}_*(\text{john}',X)))]\!]^{i,j} = 1$ iff $i=j$ and

$[\![\text{perfect}(\text{marry}_*(\text{john}',X)))]\!]^{i,j} = 1$ iff

for some i'' such that i is a final sub interval of i''

$[\![\text{marry}_*(\text{john}',X)]\!]^{i'',j} = 1\}$.

Temporal ambiguity does however arise as intended in connection with example (128); for whether the lucky girl is to become a student bride, or whether she is merely a student at present, depends upon whether or not quantification is used to introduce the object noun phrase. In the one case we have:

(137) $[\![\text{fut}(\exists X(\text{girl}'(X) \wedge \exists t(\text{at}(t,\text{present}(\text{student}'(X)))) \wedge \text{marry}_*(\text{john}',X)))]\!]^{i,j} = 1$ iff

for some $i' > j$

$[\![\exists X(\text{girl}'(X) \wedge \exists t(\text{at}(t,\text{present}(\text{student}'(X)))) \wedge \text{marry}_*(\text{john}',X)))]\!]^{i',i'} = 1$ iff

$\{$for some $i'' = [\![t]\!]^{i',i'}$

$[\![present(student'(X))]\!]^{i'',i'} = 1$ iff $i''=i'$ and

$[\![student(X)]\!]^{i'',i'} = 1\}$ and

$[\![marry_*(john',X)]\!]^{i',i'} = 1.$

which makes the girl a student at the time of the wedding. Alternatively we have:

(138) $[\![\exists X(girl'(X) \wedge \exists t(at(t,present(student'(X)))) \wedge fut(marry_*(john',X)))]\!]^{i,j} = 1$ iff

{for some $i' = [\![t]\!]^{i,j}$

$[\![present(student'(X))]\!]^{i',j} = 1$ iff $i'=j$ and

$[\![student'(X)]\!]^{i',j} = 1\}$ and

$\{ [\![fut(marry_*(john',X))]\!]^{i,j} = 1$ iff

for some $i''>j$

$[\![marry_*(john',X)]\!]^{i'',i''} = 1\}.$

as a result of which the girl is deemed to be a student at the time of utterance.

Cracks unfortunately begin to appear in Dowty's two dimensional analysis as soon as we turn our attention to example (130), for on investigation we find we may derive:

(139) $[\![past(\exists X(child'(X) \wedge \exists t(at(t,would(king'(X)))) \wedge \exists Y(bear(Y,X))))]\!]^{i,j} = 1$ iff $i<j$ and

$[\![\exists X(child'(X) \wedge \exists t(at(t,would(king'(X)))) \wedge \exists Y(bear(Y,X)))]\!]^{i,j} = 1$ iff

{for some $i' = [\![t]\!]^{i,j}$ (*with no specific relationship between i' and either i or j*)

$[\![would(king'(X))]\!]^{i',j} = 1$ iff

for some $i''>i'$

$[\![king'(X)]\!]^{i'',j} = 1\}$ and

$[\![bear(Y,X)]\!]^{i,j} = 1.$

According to this derivation, i″, the coronation date, is unordered not only with respect to the time of utterance j, but also with respect to the time i of the nativity: for no relationship between i and i′ or between j and i′ has been posited. Even if an alternative analysis were available, the existence of this one unacceptable interpretation is sufficient to constitute an objection to Dowty's two dimensional semantics as presently formulated.

The problem arises because the time t provided as valuation time for the sentence embedded in the relative clause is not constrained by (T3′) to have any particular relationship to either the speech time or the valuation time of the matrix clause. Dowty makes an abortive attempt to remedy this situation by suggesting yet another variation of (T3), viz:

$(T3'')\ f_{3,n}(\zeta,\phi) \Rightarrow \lambda x_n(\zeta'(x_n)\wedge\exists t(t\leq t^*\wedge at(t,\phi'))).$

but this turns out to be worse than useless. If in place of the italicised phrase in the last derivation we insert:

$i = [\![t^*]\!]^{i,j}$ and $i'\leq i$

it is still possible for i″ to fall between i′ and i, ie. for the coronation to precede the nativity. The only restriction in this vein that would be effective in connection with derivation (139) would be to insist that i′ and i be *equal*; but to strengthen (T3″) so that it required "t=t*" instead of "t≤t*" would reduce (T3″) to the original (T3).

Moreover neither (T3) nor (T3″) but (T3′) appears to be needed for the correct interpretation of (133) and (134).[†71] If we were to add the restriction i′≤i in either (133) or (134) we should derive a reading in which the event in the relative clause *could not be later* than the event in the matrix clause. While this would be acceptable in the case of sentence (125), an analogous derivation for example (126) would rule out the favoured chronology. Requiring i′ to be *equal* to i would of course make the events in both (125) and (126) simultaneous. Discussion of the means of escape from this impasse provided by TMG must be deferred until multi dimensional alternatives to the two dimensional system under review have been considered.

Dowty himself concedes the inadequacy of the two dimensional system since his semantic clause for "fut", which alters the *speech* time of the argument sentence, has unwelcome side effects: the interpretation of indexical time adverbs within the scope of a "fut" operator becomes corrupt. Consider a sentence such as

(140) John will marry a girl who will be a student tomorrow.

which, among others, is given the translation:

(141) $fut(\exists X(girl'(X)\wedge\exists t(at(t,tomorrow'(\lambda t(t=t^*\wedge fut(student'(X))))))\wedge marry_*(john',X)))$.

†71. Derivations (135) ... (138) would in fact survive introduction of the (T3″) condition, but the restriction would be vacuous.

If we were to evaluate (141) at <i,j>, then

$marry_*(john', X)$

would be evaluated at <i',i'> for some i'>j and the expression:

$tomorrow'(\lambda t(t{=}t^* \wedge fut(student'(X))))$

would be evaluated at <i",i'> for some i". This would make tomorrow' denote the set of all sets containing an interval within the day after the day containing i' since the semantics of indexical time adverbs relates them to the *speech* time, ie. "tomorrow" would refer to the day after the wedding.

As a patch Dowty reluctantly suggests a *three* dimensional system with evaluation relative to a triple <i,j,k>, where i is the valuation time, k an *invariant* speech time and j a "quasi speech time" which may be allowed to vary. The revised semantic clause for "fut" would be:

$$[\![fut(\phi)]\!]^{i,j,k} = 1 \text{ iff } [\![\phi]\!]^{i',i',k} = 1 \text{ for some } i'>j.$$

while indexical time adverbials would refer not to the changeable j but to the constant k. Such a solution is however disparaged by Dowty as "ad hoc", a conclusion which, in view of the multi dimensional heritage we are about to consider, is little short of amazing.

5.4. The Reichenbachian Tradition

The practice of employing multiple indices in the evaluation of tensed natural language locutions originates with Reichenbach, [R1], who observes that a system employing only two *linearly* ordered points, a point of speech S and a point of event E, can represent exactly 3 ordering relationships, viz. E<S, E=S, and S<E. Reichenbach suggests that in a tensed locution a point of reference R is first ordered with respect to S, and thereafter E is ordered with respect to this viewpoint R. He reserves the terminology "past", "present", and "future" to refer to the initial positioning of R relative to S, while "anterior", "simple", and "posterior" indicate the subsequent positioning of E relative to R.

5.4.1. Reichenbach's Nine Tense System

If we adopt the convention that {P1,P2} indicates P1 and P2 occurring simultaneously and P1—P2 indicates P1 preceding P2 then a total of thirteen permutations become possible. An initial ordering {S,R} supports E—{S,R}, {S,R,E}, and {S,R}—E, while a starting configuration of S—R gives E—S—R, {S,E}—R, S—E—R, S—{R,E} and S—R—E. Finally an initial condition of R—S will serve to generate E—R—S, {E,R}—S, R—E—S, R—{S,E} and R—S—E.

****

Reichenbach's Nine Tense System			
Code	Reichenbach	Traditional	Example
E—{S,R}	Anterior Present	Present Perfect	I have seen John today
{S,R,E}	Simple Present	Simple Present	I see John
{S,R}—E	Posterior Present	Simple Future	I shall see John today
S—E—R	Anterior Future	Future Perfect	I shall have seen John tomorrow
S—{R,E}	Simple Future	Simple Future	I shall see John tomorrow
S—R—E	Posterior Future	------?------	I shall be going to see John tomorrow
E—R—S	Anterior Past	Past Perfect	Yesterday I had seen John
{E,R}—S	Simple Past	Simple Past	I saw John yesterday
R—E—S	Posterior Past	Conditional	(I knew) I would see John yesterday

Fig 58

****

Like most tense logicians, Reichenbach is concerned to identify *possible* tense forms irrespective of whether or not they have a realisation in a particular natural language. Nevertheless not all the above combinations are considered significant. If S and E coincide then R becomes otiose thus eliminating {S,E}—R and R—{S,E}; moreover Reichenbach opines that although there may be tense forms which look first forwards and then back (future perfect) or first backwards and then forwards (conditional), there can be no utility in a configuration which places the point of speech *unequivocally*†72 in between the point of reference and the point of event. Hence E—S—R and R—S—E may both be discarded.

Accordingly Reichenbach accepts as possible tense forms the nine combinations illustrated in fig 58, which includes conventional nomenclature and examples for easy reference.

It is interesting to observe that throughout his exposition of the two dimensional system Dowty is somewhat ambivalent as to its relationship with the early Reichenbachian model. He claims that his index i "plays intuitively much the same role as Reichenbach's reference time" and that "nothing corresponds to

†72. In a linearly ordered system employing a single axis the placings are necessarily unequivocal.

Reichenbach's event time", although the affinity if any is surely between Dowty's i and Reichenbach's E.[†73]

Although Reichenbach, in attempting to accommodate progressive aspect, provides for an increase in the number of tense forms by allowing E to become an extended interval rather than a point, his system is still patently inadequate as a classification of all *possible* tenses. Not only is it parsimonious in coverage, but also it fails correctly to predict that the event reported by the "posterior past", as exemplified in example (130), should be unordered with respect to the time of utterance. Indeed two points on a single linear axis *cannot* be unordered with respect to each other.[†74]

It is accordingly the employment of a single linear axis which vitiates Reichenbach's analysis; but before turning to Bull's remedial alternative one final Reichenbachian contention deserves mention. He claims that "time determination", (as represented in indexical adverbials), must apply to the point of reference R and not to the point of event: hence, as illustrated in fig 58, "today" but not "yesterday" may qualify the anterior present, while "yesterday" is acceptable with the simple past.

5.4.2. Bull's Twelve Tense System

A temporal axis is defined in terms of a point of origin. The major innovation introduced by Bull, [B16], is the employment of a multi dimensional representation of time wherein alternative points of origin, only partially ordered with respect to each other, are permitted to govern independent axes.

From a present viewpoint or "point present", (P), an observer may anticipate a future viewpoint, the "anticipated point", (AP), or remember a previous one, the "recall point", (RP). Moreover the observer may recall that at RP a future viewpoint was anticipated which, from the point of view of P, is a "retrospective anticipated point", (ARP),[†75] The facility for generating new points of origin by recall or anticipation from existing ones is in principle open ended; but Bull is content, albeit not constrained, to consider just the four here mentioned. Plainly RP<P<AP and RP<ARP, but no ordering of ARP with respect to either P or

†73. For this reason I have eschewed the term "reference time" in my exposition of Dowty's system, preferring instead "valuation time".

†74. Dowty is able to have i'' unordered with respect to j in derivation (139) because he does *not* insist upon a single linear axis.

†75. Bull's mnemonic is RAP, but this terminology is unfortunate. It will prove convenient if, for any given point Ψ, RΨ locates an earlier viewpoint while AΨ locates a later one. Thus a point anticipated from RP should be ARP.

****

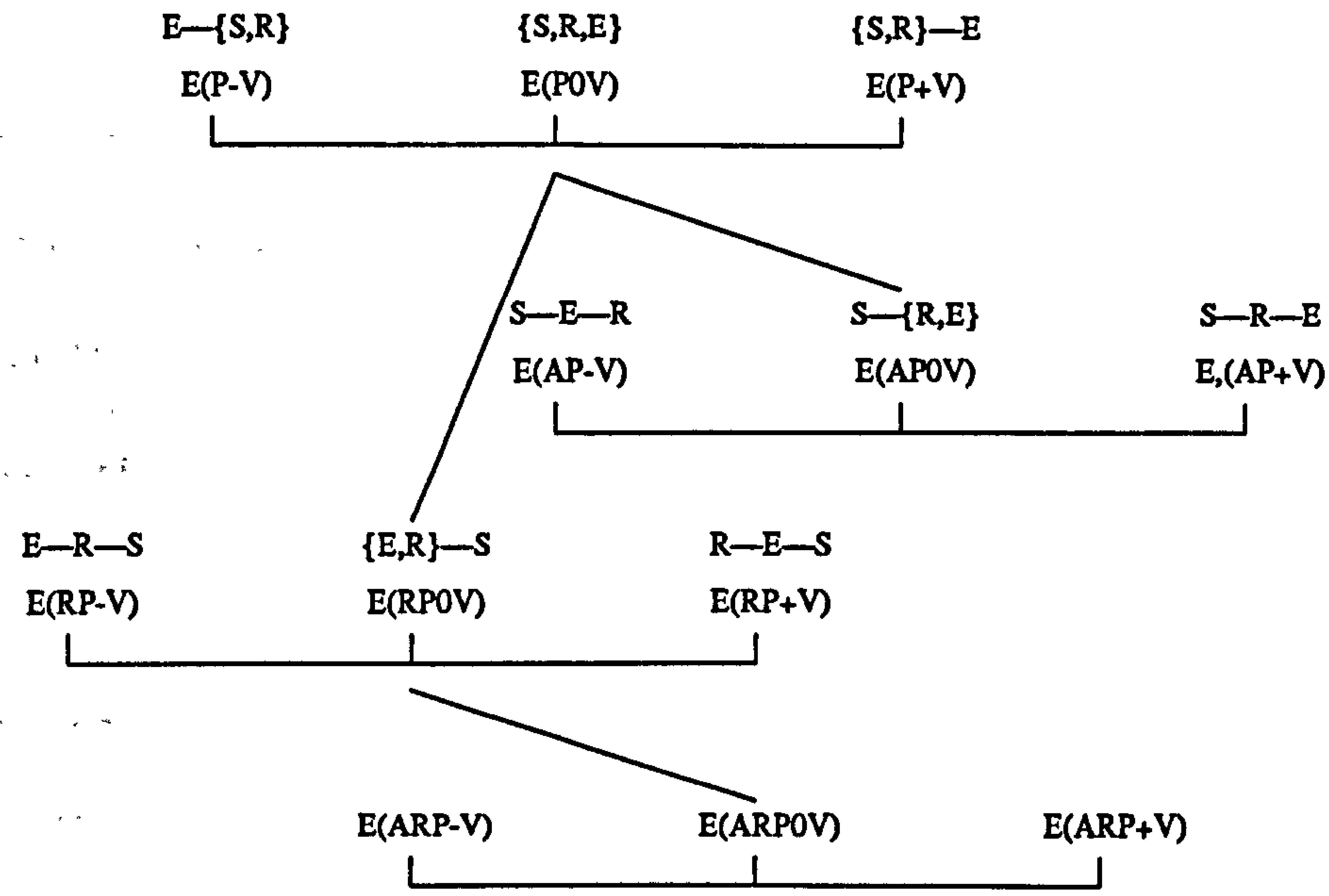


Fig 59

****

AP is prescribed. Each of these four points of origin governs an axis to which tensed constructions may make reference. Bull represents a tense by means of a *vector formula* such that where Ψ is a point of origin:

E(Ψ-V) refers to an event anterior to Ψ.

E(Ψ0V) refers to an event simultaneous with Ψ.

E(Ψ+V) refers to an event posterior to Ψ.

Given four axes and three vector formula patterns a total of twelve tenses may be identified as illustrated in fig 59 which, for convenience, also records the corresponding Reichenbach code in cases where the respective systems coincide. The only further restriction upon which Bull insists is that E(AP-V)≥PP and E(ARP-V)≥RP.

Unlike Reichenbach's tense R—E—S, the vector formula E(RP+V) does indeed leave the event time unordered with respect to the time of utterance P. Had he not been committed to a single temporal axis, Reichenbach might have achieved a similar effect by representing his anterior past by a *pair* of conditions <R—S, R—E>. With such a revised notation, Bull's E(RAP-V) would correspond to a new tense <R—S, R—R1, E—R1>, while both E(ARP0V) and E(ARP+V) would map to <R—S, R—R1, R1—E>, suggesting that one or other should be considered otiose. It is interesting to note in this connection that while Bull's E(ARP-V) has an English realisation as the perfect conditional:†76

(142) John would have sung.

and E(ARP0V) may be represented obliquely in the construction:

(143) (He would have left before) John sang.

Bull concedes that E(RAP+V) has no distinct realisation.

5.4.3. Bruce's Relational System

Given that his definition of "time segment" parallels Dowty's definition of "interval", Bruce, [B15], may be said to introduce seven binary ordering relations defined over intervals, namely:

Before: $B(i,i')$ iff $\forall m \forall m'((m \in i \wedge m' \in i') \rightarrow m < m')$.

After: $A(i,i')$ iff $B(i',i)$.

Same: $S(i,i')$ iff $i=i'$.

During: $D(i,i')$ iff $i \subset i'$.

Contains: $C(i,i')$ iff $D(i',i)$.

Overlaps: $O(i,i')$ iff $i \cap i' \neq \varnothing$

$\wedge \exists m \forall m'((m \in i \wedge m' \in i') \rightarrow m < m')$

$\wedge \exists m' \forall m((m' \in i' \wedge m \in i) \rightarrow m' > m)$.

Overlapped: $O'(i,i')$ iff $O(i',i)$.

Relations of higher degree may be defined compositionally in terms of these binary relations, for example the relation which holds if i occurred during an interval i′ before i″ is:

$D(i,i') \wedge B(i',i'')$

which Bruce proposes to identify by the shorthand:

$DB(i,i',i'')$.

---

†76. All Bull's own examples are in Spanish.

In general a relation identified by *n* letters of the alphabet must have $n+1$ argument places and represents a composition of *n* binary relations.

Defined in terms of binary ordering relations, a tense may be formulated as:

$$R_1(i_1,i_2) \wedge ... \wedge R_n(i_n,i_{n+1}).$$

where each R_j is a member of {B, A, S, D, C, O, O′}, the first argument of the first relation, ie. i_1, is the speech point, the second argument of the last relation, ie. i_{n+1}, is the event point, and each i_k for $1<k\leq n$ is an intermediate reference point.

A past tense is indicated by an initial "A", a present tense by initial "S", and a future tense by initial "B", while an "A" or an "S" in any non initial position implies perfect or simple aspect respectively. Although Bruce is cagey on this point, a terminal "D" appears to be the appropriate signal for progressive aspect. As an example of tense "BABB" Bruce cites:

(144) ?* John will have been going to be going to sing.

while I surmise that "ABBAD" would correspond to:

(145) John would be going to have been singing.

To facilitate comparison with Bull's system, I wish to propose certain modifications to the notation of the latter. Tense forms in Bull's system are represented as a combination of two phenomena, the first involving a choice of viewpoint and the second a choice of direction with respect to that point: but the manner in which these phenomena are treated lacks uniformity. Choice of a *recall* viewpoint RP always signals a past tense, while inclusion of a minus vector, -V, indicates perfect aspect. With regard to anticipated viewpoints and plus vectors however Bull tends to prevaricate. Ostensibly +V, unlike -V, is intended to represent not a somewhat nebulous "prospective aspect" but rather posteriority in order, ie. futuriety; but so too does the adoption of an anticipated viewpoint. Hence the difference between E(PP+V) and E(AP0V) is described as "basically a difference between formulations of the same facts", a conclusion which is endorsed by the observation that both of Reichenbach's corresponding forms, {S,R}—E and S—{R,E}, map to the simple future.

Arguably the functions of both plus vector and anticipated viewpoint are the same, thus I suggest a general reduction principle:

****

Twenty Four Bruce and (Revised) Bull Tenses			
Bruce	Bull	Tense Form	Example
SA	E(P-V)	Present perfect	John has sung
SAD	EI(P-V)	Present perfect progressive	John has been singing
SS	E(POV)	Present simple	John sings
SD	EI(POV)	Present progressive	John is singing
BA	E(AP-V)	Future perfect	John will have sung
BAD	EI(AP-V)	Future perfect progressive	John will have been singing
BS	E(APOV)	Future simple	John will sing
BD	EI(APOV)	Future progressive	John will be singing
BBA	E(AAP-V)	Future prospective perfect	John will be going to have sung
BBAD	EI(AAP-V)	Future prospective perfect progressive	John will be going to have been singing
BB	E(AAPOV)	Future prospective	John will be going to sing
BBD	EI(AAPOV)	Future prospective progressive	John will be going to be singing
AA	E(RP-V)	Past perfect	John had sung
AAD	EI(RP-V)	Past perfect progressive	John had been singing
AS	E(RPOV)	Past simple	John sang
AD	EI(RPOV)	Past progressive	John was singing
ABA	E(ARP-V)	Future past perfect	John would have sung
ABAD	EI(ARP-V)	Future past perfect progressive	John would have been singing
ABS	E(ARPOV)	Future past	John would sing
ABD	EI(ARPOV)	Future past progressive	John would be singing
ABBA	E(AARP-V)	Future past prospective perfect	John would be going to have sung
ABBAD	EI(AARP-V)	Future past prospective perfect progressive	John would be going to have been singing
ABB	E(AARPOV)	Future past prospective	John would be going to sing
ABBD	EI(AARPOV)	Future past prospective progressive	John would be going to be singing

Fig 60

****

$$E(\Psi+V) = E(A\Psi OV)$$

given which the plus vector becomes obsolete. At one point Bull considers but abandons the tactic of introducing an alternative vector formula of form:

****

Sixteen Further Tenses (Only Bruce Code Shown)		
Code	Tense Form	Example
SABA	Present perfect prospective perfect	John has been going to have sung
SABAD	Present perfect prospective perfect progressive	John has been going to have been singing
SAB	Present perfect prospective	John has been going to sing
SABD	Present perfect prospective progressive	John has been going to be singing
BABA	Future perfect prospective perfect	John will have been going to have sung
BABAD	Future perfect prospective perfect progressive	John will have been going to have been singing
BAB	Future perfect prospective	John will have been going to sing
BABD	Future perfect prospective progressive	John will have been going to be singing
AABA	Past perfect prospective perfect	John had been going to have sung
AABAD	Past perfect prospective perfect progressive	John had been going to have been singing
AAB	Past perfect prospective	John had been going to sing
AABD	Past perfect prospective progressive	John had been going to be singing
ABABA	Future past perfect prospective perfect	John would have been going to have sung
ABABAD	Future past perfect prospective perfect progressive	John would have been going to have been singing
ABAB	Future past perfect prospective	John would have been going to sing
ABABD	Future past perfect prospective progressive	John would have been going to be singing

Fig 61

****

EI(Ψvector)

to encapsulate imperfective (ie. progressive) aspect. If we resurrect this suggestion and in addition adopt the reduction principle formulated above then it becomes possible consistently to generate the twenty four tenses represented in fig 60 which juxtaposes both Bruce's and the revised Bull codes for ease of comparison.

In non progressive cases, Bull's "0V" and "-V" map to Bruce's "S" and "A" in final position, while "P" maps to "S", "AP" to "B", and "RP" to "A". Whereas the Bull style compound viewpoints are constructed by *prefixing* an "A" or an "R", the corresponding Bruce code is constructed by inserting "A" or "B" in *penultimate* place. Bull's progressive formulae append a "D" to the corresponding non progressive Bruce code.

Bruce's claim to greater generality than Bull is only tenable because the latter opts for a restriction to four viewpoints, an option which is not mandatory. There is however an elegance in the Bruce notation which may be employed to identify at least the further sixteen tense forms of fig 61. What gives the system its elegance is the use of a *sequence* of intermediate reference points each inheriting, modifying and passing on a relational characteristics like competitors in a relay race handing on a baton. It is this feature of Bruce's system that the tense logic facility for TMG will adopt.

CHAPTER 6. VERB PHRASES IN TMG

¶ In this chapter an integrated account of passivisation, tense and aspect is developed and incorporated in TMG. A strictly binary recursive mechanism for constructing new verb phrases from auxiliaries and existing verb phrases is introduced together with a modified tense logic necessary for interpretation of the expressions so formed. All rules finally adopted were developed, tested and verified using the computational inversion TMDCG.

6.1. Surface Oriented Syntax

Montague grammarians have, according to Dowty, [D8], been striving towards analyses which make the logical form of a sentence as close to the apparent surface syntactic form as possible. Compositionality in semantics should not, in other words, be purchased at the cost of a bizarre syntax. The syntax accompanying Dowty's own two dimensional system of interpretation is however not altogether successful in achieving the above goal, for it remains characterised by certain eccentric features.

In the case of simple present and simple past tenses, lexical verbs which originate as unmarked forms are made finite by application of the custom built rules (Spr) and (Spst); but tense forms involving auxiliaries acquire the appropriate auxiliary and finite marking simultaneously. As noted earlier, it seems anomalous to treat the marking of lexical verbs and auxiliaries in distinct fashions.

By treating (Spr) and (Spst) as *unary* rules Dowty conforms to Partee's condition C3(iii) in so far that he does not *explicitly* introduce tense markers on his trees; but the presence of such markers remains implicit in his formulations. Although overtly unary, these rules are covertly binary since both combine a tense marker and a non finite intransitive verb phrase in order to make a finite verb phrase. Likewise rules (Sft), (Swd), and (Spf) are implicitly ternary despite their binary formulations, requiring as inputs in addition to the non finite verb phrase a tense marker and an auxiliary. There seems to be little point in Partee's restriction C3(iii) if it can always be circumvented in such a way, and accordingly my policy will be to make all requisite markers explicit.

One curious consequence of Dowty's method is that "will" and "would" appear not to be related, as would have been the case had the tense marking been added separately after the introduction of a common root form. Given Dowty's two dimensional logic, it is in fact *necessary* to treat "would" as a primitive because "past(fut(ϕ))" reduces to "fut(ϕ)":

$[\![\text{past}(\text{fut}(\phi))]\!]^{i,j} = 1$ iff $i<j$ and

$[\![\text{fut}(\phi)]\!]^{i,j} = 1$ iff for some $i'>j$

$[\![\phi]\!]^{i',i'} = 1$.

Plainly the condition that $i<j$ has no bearing on the final value.

What should a natural compositional syntax for English verb phrases be like? A necessary condition must be the provision of a suitably restricted binary recursive mechanism for making new members of P_{IV} by combining auxiliaries (members of $B_{IV/nIV}$ for some *n*) and existing members of P_{IV}. Auxiliaries are verbs†77 which accept intransitive verb phrases as complements.

The negative particle in English attaches to the *leading* auxiliary in a tense form, hence further recursion must be inhibited by the introduction of "not" or "n't"; moreover there is an ordering constraint usually cited†78 as:

<modal, perfect, progressive, passive, lexical>.

which no rule for combination may be allowed to violate.

A convenient mechanism for implementing the ordering constraint is adopted in both TMG and TMDCG. The method involves specifying for each auxiliary a mandatory inflection on the leading verb of the complement and recording for each verb in the lexicon those inflections which it may possess. Some modification to the lexicon is implied; but it then becomes apparent that, for instance, the reason why the perfect auxiliary cannot be followed by a modal is simply that the complement of perfect "have" must commence with a verb marked [+enform], and modal auxiliaries lack past participles.

If a Montague grammar is to cease to rely upon an *informal* understanding of what counts as, for example, the third person singular present of a verb then the lexicon defined as in PTQ by (S1) must in any

†77. Hudson, [H8], has argued forcefully for the treatment of auxiliaries as verbs in opposition to the classical transformationalist position.

†78. The citation does not strictly speaking represent the whole truth. An auxiliary version of "go" may follow the progressive and may itself be followed by the perfect, progressive or passive, thus introducing a loop and allowing for tense forms like Bruce's BBD, ABBA, ABBAD, BABAD and ABABAD. How often this loop may be taken is uncertain, thus examples like (144) are questionable. One advantage of the mechanism to be suggested is that it handles such looping without special provision.

****

Auxiliary Inflections

```
/* verbform(Form,Inflection,Root) */

verbform(Vr,root,Vr) :- inflect(Vr,_,_,_,_,_,_).
verbform(Vo,fin(pl,pres),Vr) :- inflect(Vr,Vo,_,_,_,_,_).
verbform(Vs,fin(sg,pres),Vr) :- inflect(Vr,_,Vs,_,_,_,_).
verbform(Vd,fin(_,past),Vr)  :- inflect(Vr,_,_,Vd,_,_,_).
verbform(Vto,inf,Vr)         :- inflect(Vr,_,_,_,Vto,_,_).
verbform(Vg,ing,Vr)          :- inflect(Vr,_,_,_,_,Vg,_).
verbform(Vn,en,Vr)           :- inflect(Vr,_,_,_,_,_,Vn).

verbform(was,fin(sg,past),B) :- getnext(B,[be,bepass,beprog,bemod]).
verbform(were,fin(pl,past),B) :- getnext(B,[be,bepass,beprog,bemod]).

/* inflect(Root,Simple,Sform,Preterite,Infinitive,Ingform,Enform) */

inflect(doaux,do,does,did,[],[],[]).
inflect(bepass,are,is,[],be,being,been).
inflect(bemod,are,is,[],[],[],[]).
inflect(beprog,are,is,[],be,[],been).
inflect(goaux,[],[],[],[],going,[]).
inflect(shall,shall,shall,should,[],[],[]).
inflect(willaux,will,will,would,[],[],[]).
inflect(canaux,can,can,could,[],[],[]).
inflect(must,must,must,[],[],[],[]).
inflect(haveaux,have,has,had,have,[],[]).
```

Fig 62

****

event be augmented by details of verb inflection while noun declension seems equally fundamental and to demand recognition. Montague's naive lists of lexical items can hardly be accepted as adequate. Inflectional structure is manipulated in TMG and its computational correlate by recourse to a data structure having the form:

inflect(Root, Simple, Sform, Preterite, Infinitive, Ingform, Enform).

The clauses illustrated in fig 62, which include instances of the above available for inclusion in an augmented version of (S1), correspond to the records maintained for auxiliaries by GENLEX. The field "Root" in the "inflect" data structure is the only field instantiation of which is mandatory and empty fields are represented by the empty list [].†79

†79. Since all verbs must have a root, this cannot be identified with any surface form because no such form can be guaranteed present in all cases; thus a root so conceived constitutes an internal key field for the lexical database.

The general form of a compositional rule to handle auxiliary plus intransitive verb phrase combinations will accordingly be the following:

> If α is an auxiliary requiring a complement bearing the feature ξ and δ is a (non negative) intransitive verb phrase the leading verb(s) of which may bear feature ξ, then $\alpha\gamma$ is an intransitive verb phrase, where γ comes from δ by marking each leading verb with ξ while α remains unmarked.

Since the output from such a rule always commences with an unmarked verb specific provision must be made for converting intransitive verb phrases into finite forms. If the rules are to be explicitly binary, a tense introduction rule must combine a tense marker with an intransitive verb phrase in order to make a *finite* verb phrase: such a rule must moreover be applicable to both positive and negative inputs.

As we have already seen in example (118), there are independent reasons for acknowledging categories of passive transitive and passive intransitive verb phrases where the latter must plainly be derived from the former by introduction of an agent. A binary rule for creating passive transitive verb phrases will require as inputs a transitive verb phrase and a passive marker, while in the case of agentive passives the need is for a rule which makes passive intransitive verb phrases by combining passive transitive verb phrases with *terms*, the preposition "by" being introduced syncategorematically.

The agentless passive can be handled by a *binary* rule only by allowing an "agent" marker which may perform the role of a psuedo term in default of any specified agent. When accepting a complement, the passive auxiliary must be constrained by a rule which insists not only upon a [+enform] marking but also upon a member of the category *passive* intransitive verb phrase as input.

Devising a compositional syntax which conforms to the above conditions is not in itself problematic. The important questions are: can such a syntax support a compositional semantics and can the semantic equivalents of expressions so formed be coherently interpreted? My purpose in developing **TMG** is to demonstrate that an interpretable compositional grammar for tense and aspect answering to the above description of natural syntax is in principle possible.

6.2. Lexical Modifications in TMG

A prerequisite of the natural syntax outlined above is the availability of a number of pseudo lexical markers. Cases of such markers have already been encountered in our discussion of interrogatives; for both "?yn" and "?" have been found necessary. These, together with the markers required by the rules of tense and aspect constitute the "Pseudo Lexicon" for TMG which we identify as (S0), since by convention the standard lexicon is (S1), and which is illustrated in fig 63. The significance of the syntactic categories QIV, PIV, and LIV will be discussed in the next sub section.

****

Pseudo Lexicon (Defined by S0)			
Marker	Semantic Type	Mnemonic & Category	Translation
present	<<s<et>><<s<<s<et>>t>>t>>	TENSE = (t/T)/QIV	λpλn(pres(ˇn(p)))
past	<<s<et>><<s<<s<et>>t>>t>>	TENSE = (t/T)/QIV	λpλn(past(ˇn(p)))
?yn	<<st><<st>t>>	QMARK = (t/t)/t	λsλr(ˇr∧(r=^ˇs∨r=^¬ˇs))
?	<<st><<st>t>>	QMARK = (t/t)/t	λsλr(ˇr∧r=s)
passive	<<s<<s<<s<et>>t>><et>>> <<s<<s<et>>t>><et>>>	PASS = (PIV/T)/(LIV/T)	λbλnλX[(ˇn)^ˇb(^λpˇp(X)))]
agent	<<s<et>>t>	T = t/IV	λp∃Yˇp(Y)

Fig 63

****

The computerised translation process is simplified if the incidence of functional application in translation rules is maximised. To this end certain conventional assignments to basic expressions in the lexicon have been replaced in TMG by semantically equivalent variations.

Common nouns and intransitive verbs are translated into forms like λe(man′(e)) and λe(run′(e)), while transitive verbs map to forms like λnλe(seek′(e,n)). Sentence complement verbs map to forms like λrλe(believe′(e,r)), and infinitival complement verbs to forms like λpλe(try′(e,p)). These modifications ensure a uniform method for inserting arguments into predications, viz. lambda reduction: consequently the construction of formulae by juxtaposition or by reduction to "relational notation" becomes obsolete.

The copula is treated separately from other transitive verbs and translates as λnλeˇn(^λY(e=Y)). while auxiliary verbs are introduced with translations of the form λpλe(Ψ(ˇp(e))), where Ψ is either "perf"

****

Lexical Translations in TMG	
Common noun	$\lambda e(man'(e))$
Intransitive verb	$\lambda e(run'(e))$
Transitive verb	$\lambda n \lambda e(seek'(e,n))$
Copula	$\lambda n \lambda e\,{}^{\vee}n({}^{\wedge}\lambda Y(e=Y))$
s-comp verb	$\lambda r \lambda e(believe'(e,r))$
i-comp verb	$\lambda p \lambda e(try'(e,p))$
q-comp verb	$\lambda u \lambda e(know'(e,u))$
vp-adverb	$\lambda p \lambda e(slowly'(e,p))$
auxiliary	$\lambda p \lambda e(perf({}^{\vee}p(e)))$
temporal adverb	$\lambda f \lambda n({}^{\vee}n)[{}^{\wedge}\lambda e(at(t^*,{}^{\vee}f({}^{\wedge}\lambda p\,{}^{\vee}p(e))$ $\wedge t^* \sqsubseteq yesterday'))]$
preposition	$\lambda n \lambda p \lambda e(in'(e,{}^{\wedge}\lambda Z\,{}^{\vee}p(Z),n))$

Fig 64

****

(perfect), "prog" (progressive), "fut" (future), "nec" (necessarily) or "poss" (possibly) . Passive "be" and periphrastic "do" constitute exceptions which map to an identity operation $\lambda p \lambda e({}^{\vee}p(e))$.

Most verb phrase adverbs have translations typified by $\lambda p \lambda e(slowly'(e,p))$; but temporal adverbials have translations like:

$$\lambda f \lambda n({}^{\vee}n)[{}^{\wedge}\lambda e(at(t^*,{}^{\vee}f({}^{\wedge}\lambda p\,{}^{\vee}p(e)) \wedge t^* \sqsubseteq yesterday'))].$$

In TMG, temporal adverbials appear syntactically as modifiers of finite verb phrases, (ie. $P_{TMADV} = P_{FV/FV}$), in order to accommodate examples such as (132), but the translation guarantees that the time specification in the ultimate semantic representation applies at sentence level, as typified in:

$$at(t^*, sentence \wedge t^* \sqsubseteq yesterday').$$

These lexical changes are exemplified in fig 64, which introduces three new colloquial variables namely:

$b = v_{0<s<<s<<s<et>>t>><et>>>}$ ie $<s,f(TV)>$

$f = v_{0<s<<s<<s<et>>t>>t>>}$

$u = v_{0<s<<st>t>>}$

6.3. Subcategorising Verb Phrases

As might be expected in view of the earlier outline, TMG recognises a diversity of verb phrases, viz. TV (transitive verb), PTV (passive transitive verb), PIV (passive intransitive verb), CPV (conjoined passive intransitive verb), LIV (lexical intransitive verb), CLV (conjoined lexical intransitive verb), IV (intransitive verb), QIV (qualified intransitive verb),and FV (finite verb).

Only members of P_{PIV} may combine with the passive auxiliary, but in this case the output is in P_{IV} and may be combined with further auxiliaries to make new members of P_{IV}. Members of P_{LIV} share with members of P_{IV} the propensity for combination with auxiliaries to make members of P_{IV}, but members of P_{LIV} alone combine with periphrastic "do" or its negation whereupon the output is in P_{QIV} and can combine only with a member of P_{TENSE}. The result of any legitimate combination involving a *negated* auxiliary will always be a member of P_{QIV}.

In many cases the various categories of verb phrase recognised by TMG are related as subset and superset; for an innovative feature of the grammar is that verb phrasal categories may be subcategorised, ie:

LIV$\subseteq$ IV$\subseteq$ QIV , CLV$\subseteq$ LIV , CPV$\subseteq$ PIV. (all primitives of type <et>).

The copula alone resides in category LIV//T = COP, the categorial definition of TV for other transitive verbs becoming LIV/T, while that of PTV is PIV/T, all three having semantic type <<s<<s<et>>t>><et>>. As originally suggested by Bach, finite verb phrases are in category FV = t/T and reference to fig 63 will confirm that, as required, a TENSE is of category FV/QIV while the passive marker is of category PTV/TV. All auxiliaries are of semantic type <<s<et>><et>> which corresponds to the categories LAUX=IV/LIV ("do"), PAUX=IV/PIV (passive "be"), MAUX1=IV/IV ("will","can" etc.), MAUX2=IV//IV ("must"), TAUX=IV///IV (modal "be"), NAUX=IV////IV ("have"), and GAUX=IV/////IV (progressive "be").[†80]

Any rule defined over the superset will plainly apply to subsets but not vice versa. The rule for intro-

†80. The distinction between MAUX1 and MAUX2 is significant only in the case of a negated auxiliary. With members of MAUX1 the negation sign takes wider scope than the modal operator introduced while MAUX2 requires narrow scope negation. The semantic operations g_{15} and g_{16}, to be introduced shortly, handle the alternatives.

ducing periphrastic "do" may accordingly be restricted to taking lexical verb phrases in P_{LIV} and making qualified verb phrases. Positive auxiliaries other than the passive may be introduced by rules which both take and make category IV thus allowing inputs of type LIV. The rule for combining negative auxiliaries with members of P_{IV} may create P_{QIV} output thus inhibiting further chaining. The rule for making members of P_{FV} must combine tense markers with members of P_{QIV}, which includes both P_{IV} and P_{LIV}, while the subject predicate rule, which invokes reverse functional application, (semantic operation g_1), must expect a member of P_{FV} as the verb phrase input. The subcategories CLV$\subseteq$LIV and CPV$\subseteq$PIV are distinguished so that the verb phrase quantification rules may be restricted to cases where results are significant.

6.4. The Verb Phrase Rules

All syntax rules in TMG are to be formulable in PNF and all translation rules will make explicit reference to predefined semantic operations. Some of the PROLOG style predicates required for PNF definitions and some of the semantic operations to be employed have already been introduced in earlier chapters, but several additions will become necessary as we proceed.

For convenience the full catalogue of semantic operations to which translation rules make reference is listed in fig 65 which also cites the syntactic rules responsible for invocation.

The subject predicate rule (S4) of PTQ is to be replaced in TMG by (S40) which like Bach's (Ssp) must now be correlated with a translation rule imposing *inverse* functional application by recourse to g_1. Furthermore S40 must embody the appropriate conditions for reflexivisation which are to be expressed with the aid of the predicates "xform" and "xsub" illustrated in fig 66. Inflectional restrictions, as demanded by the auxiliary introduction rules, will be stated in terms of a *series* of predicates of which "ingform" in fig 66 is but a typical example. Others in the series are "simple", "sform", "pastform", "infform" and "enform" the definitions for which may be derived by analogy.[†81]

These predicates assume that all expressions may be represented in the form: cat(D,F), where D is a

†81. Because of the irregularity of "be", the predicate "pastform" will require a *third* argument N to be instantiated by either "fin(sg,past)" or "fin(pl,past)". The first subgoal of the second "pastform" clause will then be "verbform(Form,N,Root)". The predicate "enform" must accept both intransitive and transitive inputs.

****

Semantic Op.	Definition	Invoking Rule
$g_0(\theta',\eta')$	$\theta'(\hat{}\eta')$	Default
$g_1(\theta',\eta')$	$\eta'(\hat{}\theta')$	S40, S77
$g_{2,n}(\alpha',\phi')$	$\alpha'(\hat{}\lambda x_n\phi')$	S14
$g_{3,n}(\zeta',\phi')$	$\lambda x_n(\zeta'(x_n)\wedge\exists t(ab(t,\phi'))$	S30, S31
$g_{4,n}(\alpha',\theta')$	$\lambda Y\alpha'(\hat{}\lambda x_n[\theta'(Y)])$	S15, S16, S18
$g_5(\gamma,\delta')$	$\lambda e(\gamma(e)@\delta'(e))$	S71, S72
$g_6(\gamma,\delta')$	$\lambda e(\gamma(e)\vee\delta'(e))$	S71,S72
$g_7(\phi',\psi')$	$\lambda n(\phi'(n)@\psi'(n))$	S70
$g_8(\phi',\psi')$	$\lambda n(\phi'(n)\vee\psi'(n))$	S70
$g_9(\lambda n\lambda X[(\hat{}n)\phi],\lambda n\lambda X[(\hat{}n)\psi])$	$\lambda n\lambda X[(\hat{}n)\hat{}\lambda e(\hat{}\phi(e)@\psi(e))]$	S73
$g_{10}(\lambda n\lambda X[(\hat{}n)\phi],\lambda n\lambda X[(\hat{}n)\psi])$	$\lambda n\lambda X[(\hat{}n)\hat{}\lambda e(\hat{}\phi(e)\vee\psi(e))]$	S73
$g_{11}(\phi',\psi')$	$(\phi'\wedge\psi')$	S11
$g_{12}(\phi',\psi')$	$(\phi'\vee\psi')$	S11
$g_{13}(\alpha',\beta')$	$\lambda q(\alpha'(q)\wedge\beta'(q))$	S13
$g_{14}(\alpha',\beta')$	$\lambda q(\alpha'(q)\vee\beta'(q))$	S13
$g_{15}(\lambda p\lambda e(\phi,\psi')$	$\lambda p\lambda e\neg\phi(\hat{}\psi')$	S43, S45, S47, S49, S53, S55
$g_{16}(\lambda p\lambda e\delta(\xi),\psi')$	$\lambda p\lambda e\delta(\neg\xi)(\hat{}\psi')$	S51
$g_{17}(\phi',\psi')$	$\lambda r(\phi'(r)\vee\psi'(r))$	S22,S23
$g_{18,n}(\phi',\psi')$	$\lambda r(\phi'(\hat{}\lambda x_n(\psi'(r))))$	S24, S25
$g_{19,n}(\phi',\psi')$	$\lambda r(\neg\phi'(\hat{}\lambda x_n(\neg\psi'(r))))$	S26
$g_{20}(\delta',\phi')$	$\delta'(\hat{}\exists t(ab(t,\phi')))$	S61
$g_{21}(\delta',\phi')$	$\delta'(\hat{}\lambda Z\phi'(Z))$	S62, S74, S75
$g_{22}(\delta',\lambda n\lambda X[(\hat{}n)\hat{}\phi])$	$\lambda n\lambda X(\hat{}n)\hat{}\delta'(\hat{}\lambda Z\phi(Z))$	S76

Fig 65

****

list of daughters and F a feature list, but they are neutral as to the actual content of the feature list which, in the limiting case, could be empty. Note also that as defined "ingform" represents an overkill in so far that it both adds a feature to the list *and* changes the root to an inflected form. Arguably one or other of these changes alone would suffice, but the presence of both causes no embarrassment since the modification required to eliminate either option is obvious: removing an unwanted facility is simpler than introducing a

****

```
/* R results from reflexivising accusative variable V with index N */

xform(term([V],[acc]),N,term([R],[acc])) :-
            name(V,W),
            freestem(acc,Stem),
            append(Stem,Suffix,W),
            name(N,Suffix),integer(N),
            append(Stem,[115,101,108,102|Suffix],W1),
            name(R,W1).

/* Y is the result of reflexivising all accusative variables in X */
/* having index N */

xsub([V|T],N,[R|T1]) :- xform(V,N,R),!,xsub(T,N,T1).
xsub([H|T],N,[H1|T1]) :- xsub(H,N,H1),xsub(T,N,T1).
xsub(X,N,Y) :- lform(X,[P,X1,F]),xsub(X1,N,Y1),lform(Y,[P,Y1,F]).
xsub(X,N,X).

/* Arg2 is Arg1 with each head verb marked as [+ingform] */
/* else fail if no such form */

ingform(vp([VP1,Conj,VP2],K),vp([VP3,Conj,VP4],K1)) :- !,
            (Conj=and ; Conj=or),
            ingform(VP1,VP3),ingform(VP2,VP4),
            join(K,'+ing',K1).

ingform(vp([v([Root],F)|C],K),vp([v([Form],F)|C],K1)) :-
            verbform(Form,ing,Root),not(Form=[]),
            join(K,'+ing',K1).

/* Arg3 is Arg1 with each first variable of index N marked accusative */

accform(term([T1,Conj,T2],K),N,term([T3,Conj,T4],K1)) :-
      (Conj=and;Conj=or),
      accform(T1,N,T3),accform(T2,N,T4),
      join(K,'+acc',K1).

accform(term([V],[nom]),N,term([V1],[acc])) :-
      synvar(term([V],[nom]),N),!,
      synvar(term([V1],[acc]),N).

accform(T,N,T).
```

Fig 66

****

missing one, hence both choices are made available.

The predicate "accform" is to have the effect engineered in Friedman's reformulation of (S5) where each member of firstvar(β) of form he_n must be replaced in β by him_n. Given the present definition of "synvar", (fig 27), the predicate "accform" would lack sufficient generality for this purpose in that it would require specification of the index N *as input*. Such a restriction would not be critical were "accform" to be used only in connection with variable binding rules where ex hypothesi the index is already known; but (S5) is not a variable binding rule and accordingly in any variation thereof we should require the predicate to succeed for *arbitrary* values of N. The required generality can in fact be achieved by supplementing the definition of "synvar" as in fig 67.

****

```
/* Arg1 is a syntactic variable with index N */
/* for specified or arbitrary N              */

synvar(term([V],[Case]),N) :-
        nonvar(N),name(N,Suffix),
        varstem(Case,Stem),
        append(Stem,Suffix,Ascii),
        name(V,Ascii).

synvar(term([V],[Case]),N) :-
        name(V,Ascii),
        varstem(Case,Stem),
        append(Stem,Suffix,Ascii),
        name(N,Suffix).

/* Arg3 is Arg1 with each head noun marked with number of Arg2 */

decline(cn(CN,K),sg,cn(CN,K1)) :-
        join(K,'+sg',K1).

decline(cn([n([S],F)|C],K),pl,cn([n([P],F)|C],K1)) :-
        noun(S,P,F),
        join(K,'+pl',K1).
```

Fig 67

****

Some plural terms have been introduced experimentally into TMG, so fig 67 also includes a "decline" predicate which returns a common noun phrase marked for number. If all expressions are held in the form cat(D,F), I assume that in need the definitions of procedures such as "features(CAT,F)" which isolates the feature list in the structured list "CAT", "number(CAT,N)" which identifies the number feature from the feature list, "case(CAT,C)" which determines the case, and "gender(CAT,G)" which pinpoints the gender are trivial.

6.4.1. Plural Terms and the Subject Predicate Rule

The rules for combining determiners with common noun phrases and for matching terms with concordant finite verb phrases are as follows:

(S2) If $\alpha \in P_{DET}$ and $\zeta \in P_{CN}$ then $f_2(\alpha,\zeta) \in P_T$.

$f_2(\alpha,\zeta)$ = term([α,ξ],F) where number(α,N), decline(ζ,N,ξ), features(ξ,F).

(T2) If $\alpha \Rightarrow \alpha'$ and $\zeta \Rightarrow \zeta'$ then $f_2(\alpha,\zeta) \Rightarrow g_0(\alpha',\zeta')$.

(S40) If $\alpha \in P_T$ and $\delta \in P_{FV}$ and SA40 is fulfilled then $f_{40}(\alpha,\delta) \in P_t$.

SA40: number(α,N), number(δ,M), N=M.

$f_{40}(\alpha,\delta)$ = t([α,γ],[dcl]) where if $\alpha = he_k$ then xsub(δ,k,η), rsub(η,γ) else rsub(δ,γ).

(T40) If $\alpha \Rightarrow \alpha'$ and $\delta \Rightarrow \delta'$ then $f_{40}(\alpha,\delta) \Rightarrow g_1(\alpha',\delta')$.

Reference to appendix A, which contains a full listing of TMG rules, will serve to confirm that rule (S13) for forming compound noun phrases has been amended to reflect the divers possibilities of combination. A *conjunction* of noun phrases receives the number feature [+plural] while the gender of the compound, which proves immaterial, is left open. Example (H-26) in appendix H illustrates the incorporation of conjoined noun phrases in TMDCG.

When two noun phrases are *disjoined*, the gender of the disjunction is marked as [+whichever] so that anaphoric reference to the whole disjunct may be accommodated in the manner suggested at the end of section §3.2: example (H-23) illustrates the handling of the phenomenon in question. The number feature for a disjoined noun phrase is read from the final disjunct since normative grammarians appear to opine, [Q5], that a "principle of proximity" imposes concord between the final disjunct and the verb as in:

(146) A cow and the horses are running.

This sentence is parsed in TMDCG as in example (H-24), while (H-25) offers an analysis of the more dubious case:

(147) ? The horses or the cow is running.

6.4.2. Finite Verb Phrase Rules

Present and past markers from the pseudo lexicon are combined with members of P_{QIV} by the following two rules:

(S41) If α = present and $\delta \in P_{QIV}$ then $f_{41.1}(\alpha,\delta) \in P_{FV}$ and $f_{41.2}(\alpha,\delta) \in P_{FV}$.

$f_{41.1}(\alpha,\delta) = \xi$ where simple(δ,ξ) and $f_{41.2}(\alpha,\delta) = \xi$ where sform(δ,ξ).

(T41) If $\alpha \twoheadrightarrow \alpha'$ and $\delta \twoheadrightarrow \delta'$ then $f_{41.n}(\alpha,\delta) \twoheadrightarrow g_0(\alpha',\delta')$.

(S42) If α = past and $\delta \in P_{QIV}$ then $f_{42.1}(\alpha,\delta) \in P_{FV}$ and $f_{42.2}(\alpha,\delta) \in P_{FV}$.

$f_{42.1}(\alpha,\delta) = \xi$ where pastform(δ,ξ,fin(sg,past)) and $f_{42.2}(\alpha,\delta) = \xi$ where pastform(δ,ξ,fin(pl,past)).

(T42) If $\alpha \twoheadrightarrow \alpha'$ and $\delta \twoheadrightarrow \delta'$ then $f_{42.n}(\alpha,\delta) \twoheadrightarrow g_0(\alpha',\delta')$.

6.4.3. The Auxiliary System

Rule for introducing auxiliaries come in *pairs*, the first member of each pair handling the negative and the second the positive option.

(S43) If $\delta \in P_{LAUX}$ and $\gamma \in P_{LIV}$ then $f_{43}(\delta,\gamma) \in P_{QIV}$.

$f_{43}(\delta,\gamma)$ = iv([δ,not,ξ],[laux]) where infform(γ,ξ).

(T43) If $\delta \twoheadrightarrow \delta'$ and $\gamma \twoheadrightarrow \gamma'$ then $f_{43}(\delta,\gamma) \twoheadrightarrow g_{15}(\delta',\gamma')$.

(S44) If $\delta \in P_{LAUX}$ and $\gamma \in P_{LIV}$ then $f_{44}(\delta,\gamma) \in P_{IV}$.

$f_{44}(\delta,\gamma)$ = iv([δ,ξ],[laux]) where infform(γ,ξ).

(T44) If $\delta \twoheadrightarrow \delta'$ and $\gamma \twoheadrightarrow \gamma'$ then $f_{44}(\delta,\gamma) \twoheadrightarrow g_0(\delta',\gamma')$.

(S45) If $\delta \in P_{PAUX}$ and $\gamma \in P_{PIV}$ then $f_{45}(\delta,\gamma) \in P_{QIV}$.

$f_{45}(\delta,\gamma)$ = iv([δ,not,ξ],[paux]) .

(T45) If $\delta \twoheadrightarrow \delta'$ and $\gamma \twoheadrightarrow \gamma'$ then $f_{45}(\delta,\gamma) \twoheadrightarrow g_{15}(\delta',\gamma')$.

(S46) If $\delta \in P_{PAUX}$ and $\gamma \in P_{PIV}$ then $f_{46}(\delta,\gamma) \in P_{IV}$.

$f_{46}(\delta,\gamma)$ = iv([δ,ξ],[paux]) .

(T46) If $\delta \twoheadrightarrow \delta'$ and $\gamma \twoheadrightarrow \gamma'$ then $f_{46}(\delta,\gamma) \twoheadrightarrow g_0(\delta',\gamma')$.

(S47) If $\delta \in P_{TAUX}$ and $\gamma \in P_{IV}$ and SA47 is fulfilled then $f_{47}(\delta,\gamma) \in P_{QIV}$.

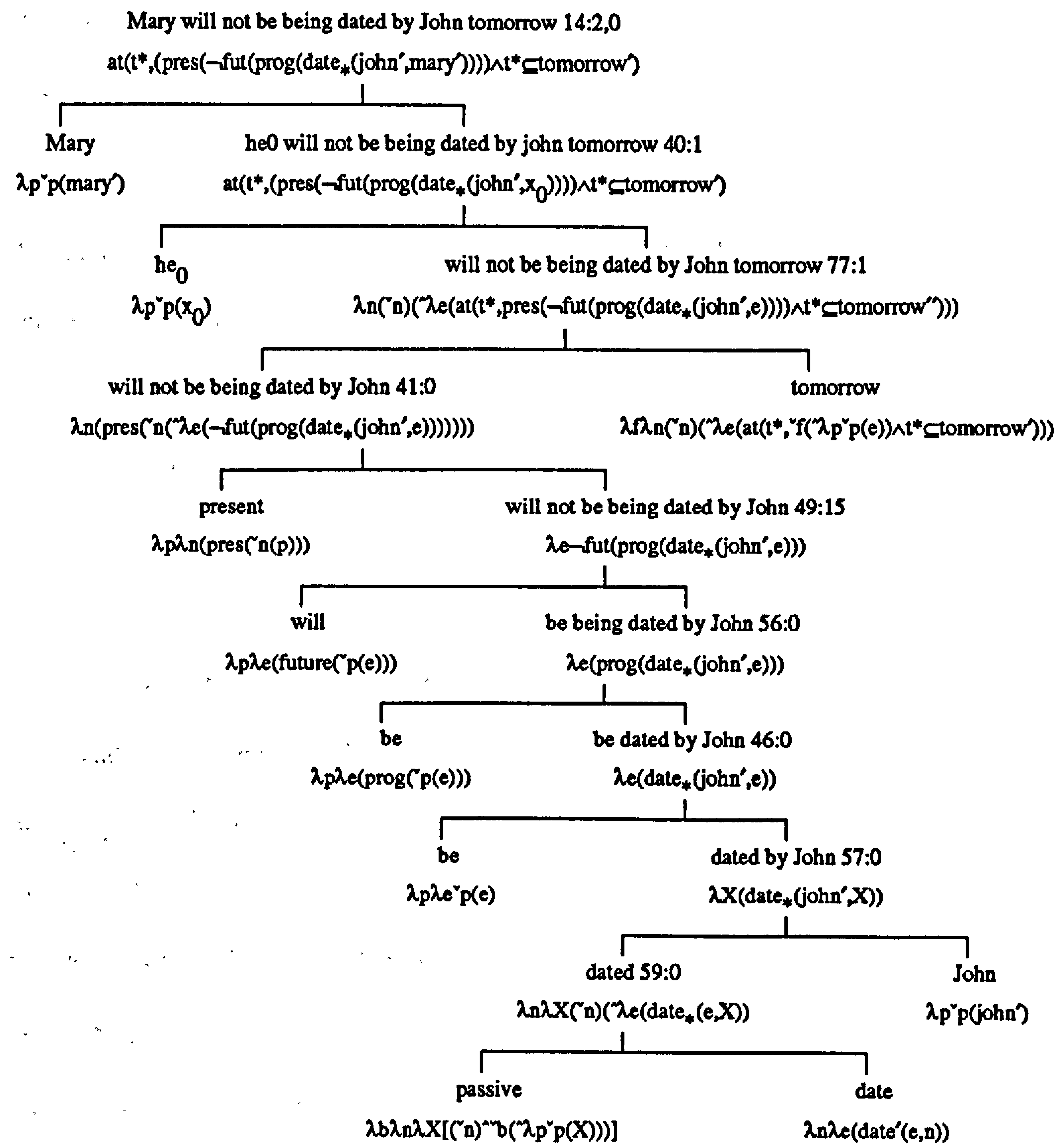


Fig 68

SA47: infform(γ,ξ) succeeds

$f_{47}(\delta,\gamma)$ = iv([δ,not,to,ξ],[taux]) where infform(γ,ξ).

(T47) If $\delta \Rightarrow \delta'$ and $\gamma \Rightarrow \gamma'$ then $f_{47}(\delta,\gamma) \Rightarrow g_{15}(\delta',\gamma')$.

(S48) If $\delta \in P_{TAUX}$ and $\gamma \in P_{IV}$ and SA48 is fulfilled then $f_{48}(\delta,\gamma) \in P_{IV}$.

SA48: infform(γ,ξ) succeeds

$f_{48}(\delta,\gamma)$ = iv([δ,to,ξ],[taux]) where infform(γ,ξ).

(T48) If $\delta \Rightarrow \delta'$ and $\gamma \Rightarrow \gamma'$ then $f_{48}(\delta,\gamma) \Rightarrow g_0(\delta',\gamma')$.

(S49) If $\delta \in P_{MAUX1}$ and $\gamma \in P_{IV}$ and SA49 is fulfilled then $f_{49}(\delta,\gamma) \in P_{QIV}$.

SA49: infform(γ,ξ) succeeds

$f_{49}(\delta,\gamma)$ = iv([δ,not,ξ],[maux1]) where infform(γ,ξ).

(T49) If $\delta \Rightarrow \delta'$ and $\gamma \Rightarrow \gamma'$ then $f_{49}(\delta,\gamma) \Rightarrow g_{15}(\delta',\gamma')$.

(S50) If $\delta \in P_{MAUX1}$ and $\gamma \in P_{IV}$ and SA50 is fulfilled then $f_{50}(\delta,\gamma) \in P_{IV}$.

SA50: infform(γ,ξ) succeeds

$f_{50}(\delta,\gamma)$ = iv([δ,ξ],[maux1]) where infform(γ,ξ).

(T50) If $\delta \Rightarrow \delta'$ and $\gamma \Rightarrow \gamma'$ then $f_{50}(\delta,\gamma) \Rightarrow g_{0}(\delta',\gamma')$.

(S51) If $\delta \in P_{MAUX2}$ and $\gamma \in P_{IV}$ and SA51 is fulfilled then $f_{51}(\delta,\gamma) \in P_{QIV}$.

SA51: infform(γ,ξ) succeeds

$f_{51}(\delta,\gamma)$ = iv([δ,not,ξ],[maux2]) where infform(γ,ξ).

(T51) If $\delta \Rightarrow \delta'$ and $\gamma \Rightarrow \gamma'$ then $f_{51}(\delta,\gamma) \Rightarrow g_{16}(\delta',\gamma')$.

(S52) If $\delta \in P_{MAUX2}$ and $\gamma \in P_{IV}$ and SA52 is fulfilled then $f_{52}(\delta,\gamma) \in P_{IV}$.

SA52: infform(γ,ξ) succeeds

$f_{52}(\delta,\gamma)$ = iv([δ,ξ],[maux2]) where infform(γ,ξ).

(T52) If $\delta \Rightarrow \delta'$ and $\gamma \Rightarrow \gamma'$ then $f_{52}(\delta,\gamma) \Rightarrow g_{0}(\delta',\gamma')$.

(S53) If $\delta \in P_{NAUX}$ and $\gamma \in P_{IV}$ and SA53 is fulfilled then $f_{53}(\delta,\gamma) \in P_{QIV}$.

SA53: enform(γ,ξ) succeeds

$f_{53}(\delta,\gamma)$ = iv([δ,not,ξ],[naux]) where enform(γ,ξ).

(T53) If $\delta \Rightarrow \delta'$ and $\gamma \Rightarrow \gamma'$ then $f_{53}(\delta,\gamma) \Rightarrow g_{15}(\delta',\gamma')$.

(S54) If $\delta \in P_{NAUX}$ and $\gamma \in P_{IV}$ and SA54 is fulfilled then $f_{54}(\delta,\gamma) \in P_{IV}$.

SA54: enform(γ,ξ) succeeds

$f_{54}(\delta,\gamma)$ = iv([δ,ξ],[naux]) where enform(γ,ξ).

(T54) If $\delta \Rightarrow \delta'$ and $\gamma \Rightarrow \gamma'$ then $f_{54}(\delta,\gamma) \Rightarrow g_{0}(\delta',\gamma')$.

(S55) If $\delta \in P_{GAUX}$ and $\gamma \in P_{IV}$ and SA55 is fulfilled then $f_{55}(\delta,\gamma) \in P_{QIV}$.

SA55: ingform(γ,ξ) succeeds

****

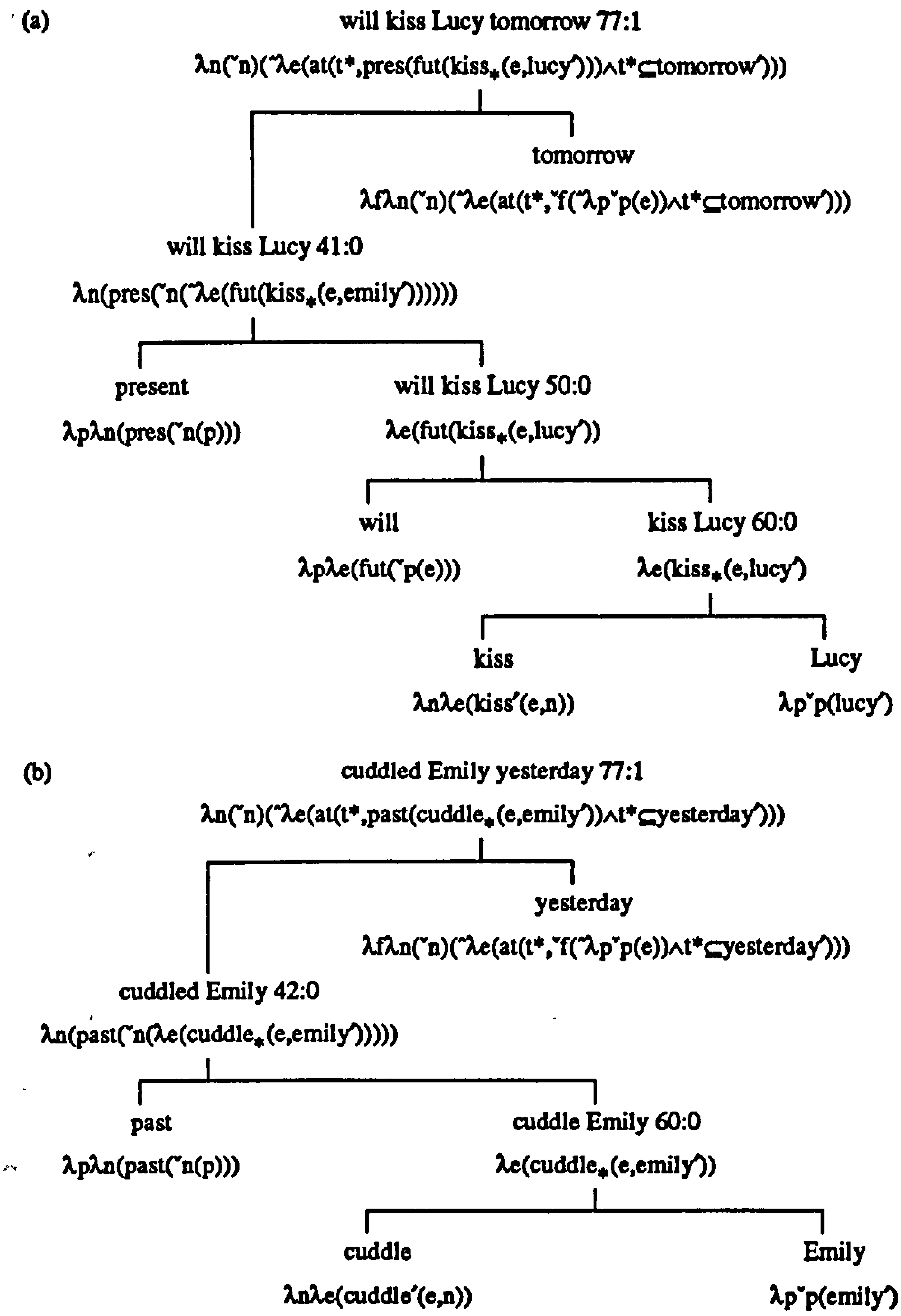


Fig 69

****

$f_{55}(\delta,\gamma) = iv([\delta,not,\xi],[gaux])$ where $ingform(\gamma,\xi)$.

(T55) If $\delta \Rightarrow \delta'$ and $\gamma \Rightarrow \gamma'$ then $f_{55}(\delta,\gamma) \Rightarrow g_{15}(\delta',\gamma')$.

(S56) If $\delta \in P_{GAUX}$ and $\gamma \in P_{IV}$ and SA56 is fulfilled then $f_{56}(\delta,\gamma) \in P_{IV}$.

SA56: $ingform(\gamma,\xi)$ succeeds

$f_{56}(\delta,\gamma) = iv([\delta,\xi],[gaux])$ where $ingform(\gamma,\xi)$.

(T56) If $\delta \Rightarrow \delta'$ and $\gamma \Rightarrow \gamma'$ then $f_{56}(\delta,\gamma) \Rightarrow g_0(\delta',\gamma')$.

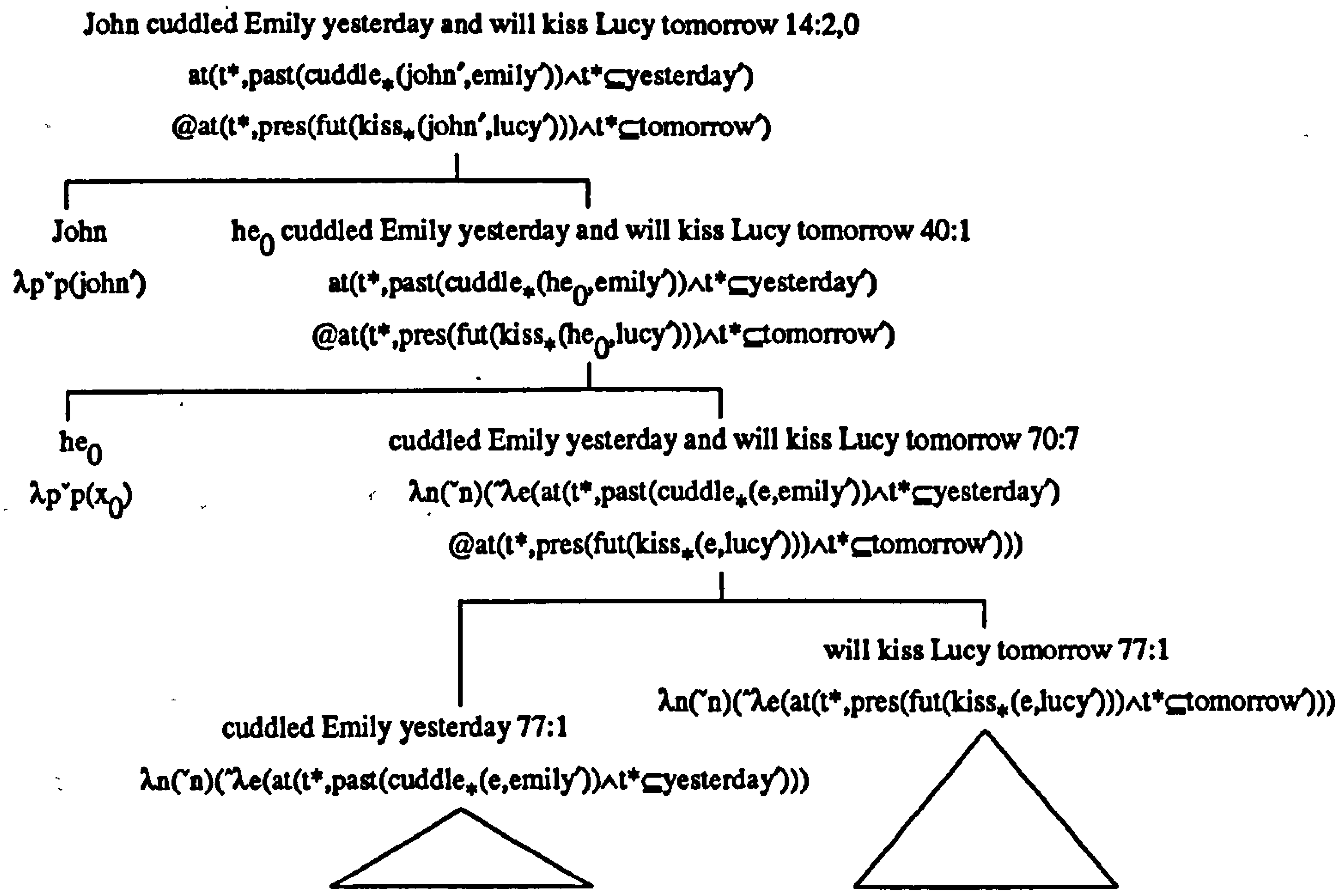


Fig 70

6.4.4. Rules of Passivisation

Three rules handle the creation of agentive passive intransitive verb phrases, agentless passive intransitive verb phrases and passive transitive verb phrases respectively.

(S57) If $\delta \in P_{PTV}$ and $\beta \in P_T$ and SA57 is fulfilled then $f_{57}(\delta,\beta) \in P_{PIV}$.

SA57: $\beta \neq$ agent

$f_{57}(\delta,\beta) = $ piv([δ,by,ξ],F) where accform(β,ξ,_) and features(δ,F).

(T57) If $\delta \leadsto \delta'$ and $\beta \leadsto \beta'$ then $f_{57}(\delta,\beta) \leadsto g_0(\delta',\beta')$.

(S58) If $\delta \in P_{PTV}$ and $\beta \in P_T$ and SA58 is fulfilled then $f_{58}(\delta,\beta) \in P_{PIV}$.

SA58: $\beta =$ agent

$f_{58}(\delta,\beta) = $ piv([δ],F) where features(δ,F).

(T58) If $\delta \leadsto \delta'$ and $\beta \leadsto \beta'$ then $f_{58}(\delta,\beta) \leadsto g_0(\delta',\beta')$.

(S59) If $\alpha =$ passive and $\delta \in P_{TV}$ then $f_{59}(\alpha,\delta) \in P_{PTV}$.

$f_{59}(\alpha,\delta) = $ ptv([ξ],F) where enform(δ,ξ) and features(δ,F).

(T59) If $\delta \leadsto \delta'$ and $\beta \leadsto \beta'$ then $f_{59}(\delta,\beta) \leadsto g_0(\delta',\beta')$.

6.4.5. Rules of Complementation

Montague's original rules (S5), (S7), and (S8) are replaced in TMG by (S60), (S61) and (S62) while an additional rule (S64) is introduced to handle the copula. Rule (S63) has already been encountered in connection with enbedded question complements.

(S60) If $\delta \in P_{TV}$ and $\beta \in P_T$ then $f_{60}(\delta,\beta) \in P_{LIV}$.

$f_{60}(\delta,\beta) = iv([\delta,\xi],[liv])$ where accform($\beta,\xi,\_$).

(T60) If $\delta \Rightarrow \delta'$ and $\beta \Rightarrow \beta'$ then $f_{60}(\delta,\beta) \Rightarrow g_0(\delta',\beta')$.

(S61) If $\delta \in P_{SCVERB}$ and $\phi \in P_t$ then $f_{61}(\delta,\phi) \in P_{LIV}$.

$f_{61}(\delta,\phi) = iv([\delta,\phi],[liv])$.

(T61) If $\delta \Rightarrow \delta'$ and $\phi \Rightarrow \phi'$ then $f_{61}(\delta,\phi) \Rightarrow g_{20}(\delta',\phi')$.

(S62) If $\delta \in P_{ICVERB}$ and $\gamma \in P_{IV}$ then $f_{62}(\delta,\gamma) \in P_{LIV}$.

$f_{62}(\delta,\gamma) = iv([\delta,\gamma],[liv])$.

(T62) If $\delta \Rightarrow \delta'$ and $\gamma \Rightarrow \gamma'$ then $f_{62}(\delta,\gamma) \Rightarrow g_{21}(\delta',\gamma')$.

(S64) If $\delta \in P_{COP}$ and $\beta \in P_T$ then $f_{64}(\delta,\beta) \in P_{LIV}$.

$f_{64}(\delta,\beta) = iv([\delta,\beta],[iv])$.

(T64) If $\delta \Rightarrow \delta'$ and $\beta \Rightarrow \beta'$ then $f_{64}(\delta,\beta) \Rightarrow g_0(\delta',\beta')$.

6.4.6. Verb Phrase Conjunctions

Finite intransitive verb phrases, lexical intransitive verb phrases, passive intransitive verb phrases and passive transitive verb phrases may all appear in conjunctions, hence Montague's (S12) must be replaced by four separate rules S70 ... S73.

(S70) If $\gamma, \delta \in P_{FV}$ then $f_{70.1}(\gamma,\delta) \in P_{FV}$ and $f_{70.2}(\gamma,\delta) \in P_{FV}$.

$f_{70.1}(\gamma,\delta) = fv([\gamma,and,\delta],[fv])$ and $f_{70.2}(\gamma,\delta) = fv([\gamma,or,\delta],[fv])$.

(T70) If $\gamma \Rightarrow \gamma'$ and $\delta \Rightarrow \delta'$ then $f_{70.1}(\gamma,\delta) \Rightarrow g_7(\gamma',\delta')$ and $f_{70.2}(\gamma,\delta) \Rightarrow g_8(\gamma',\delta')$.

(S71) If $\gamma, \delta \in P_{LIV}$ then $f_{71.1}(\gamma,\delta) \in P_{CLV}$ and $f_{71.2}(\gamma,\delta) \in P_{CLV}$.

$f_{71.1}(\gamma,\delta) = iv([\gamma,and,\delta],[clv])$ and $f_{71.2}(\gamma,\delta) = iv([\gamma,or,\delta],[clv])$.

(T71) If $\gamma \Rightarrow \gamma'$ and $\delta \Rightarrow \delta'$ then $f_{71.1}(\gamma,\delta) \Rightarrow g_5(\gamma',\delta')$ and $f_{71.2}(\gamma,\delta) \Rightarrow g_6(\gamma',\delta')$.

(S72) If $\gamma, \delta \in P_{PIV}$ then $f_{72.1}(\gamma,\delta) \in P_{CPV}$ and $f_{72.2}(\gamma,\delta) \in P_{CPV}$.

$f_{72.1}(\gamma,\delta)$ = iv([γ,and,δ],[cpv]) and $f_{72.2}(\gamma,\delta)$ = iv([γ,or,δ],[cpv]).

(T72) If $\gamma \Rightarrow \gamma'$ and $\delta \Rightarrow \delta'$ then $f_{72.1}(\gamma,\delta) \Rightarrow g_5(\gamma',\delta')$ and $f_{72.2}(\gamma,\delta) \Rightarrow g_6(\gamma',\delta')$.

(S73) If $\gamma, \delta \in P_{PTV}$ then $f_{73.1}(\gamma,\delta) \in P_{PTV}$ and $f_{73.2}(\gamma,\delta) \in P_{PTV}$.

$f_{73.1}(\gamma,\delta)$ = iv([γ,and,δ],[ptv]) and $f_{73.2}(\gamma,\delta)$ = iv([γ,or,δ],[ptv]).

(T73) If $\gamma \Rightarrow \gamma'$ and $\delta \Rightarrow \delta'$ then $f_{73.1}(\gamma,\delta) \Rightarrow g_9(\gamma',\delta')$ and $f_{73.2}(\gamma,\delta) \Rightarrow g_{10}(\gamma',\delta')$.

6.4.7. Adverbial Qualification Rules

Lexical intransitive verb phrases, passive intransitive verb phrases and passive transitive verb phrases all admit adverbial qualification while temporal adverbials may qualify finite intransitive verb phrases.

(S74) If $\gamma \in P_{IAV}$ and $\delta \in P_{LIV}$ then $f_{74}(\gamma,\delta) \in P_{LIV}$.

$f_{74}(\gamma,\delta)$ = iv([γ,δ],[liv]).

(T74) If $\gamma \Rightarrow \gamma'$ and $\delta \Rightarrow \delta'$ then $f_{74}(\gamma,\delta) \Rightarrow g_{21}(\gamma',\delta')$.

(S75) If $\gamma \in P_{IAV}$ and $\delta \in P_{PIV}$ then $f_{75}(\gamma,\delta) \in P_{PIV}$.

$f_{75}(\gamma,\delta)$ = piv([γ,δ],[piv]).

(T75) If $\gamma \Rightarrow \gamma'$ and $\delta \Rightarrow \delta'$ then $f_{75}(\gamma,\delta) \Rightarrow g_{21}(\gamma',\delta')$.

(S76) If $\gamma \in P_{IAV}$ and $\delta \in P_{PTV}$ then $f_{76}(\gamma,\delta) \in P_{PTV}$.

$f_{76}(\gamma,\delta)$ = ptv([γ,δ],[ptv]).

(T76) If $\gamma \Rightarrow \gamma'$ and $\delta \Rightarrow \delta'$ then $f_{76}(\gamma,\delta) \Rightarrow g_{22}(\gamma',\delta')$.

(S77) If $\gamma \in P_{FV}$ and $\delta \in P_{TMADV}$ then $f_{77}(\gamma,\delta) \in P_{FV}$.

$f_{77}(\gamma,\delta)$ = iv([γ,δ],F) where features(γ,F).

(T77) If $\gamma \Rightarrow \gamma'$ and $\delta \Rightarrow \delta'$ then $f_{77}(\gamma,\delta) \Rightarrow g_1(\gamma',\delta')$.

****

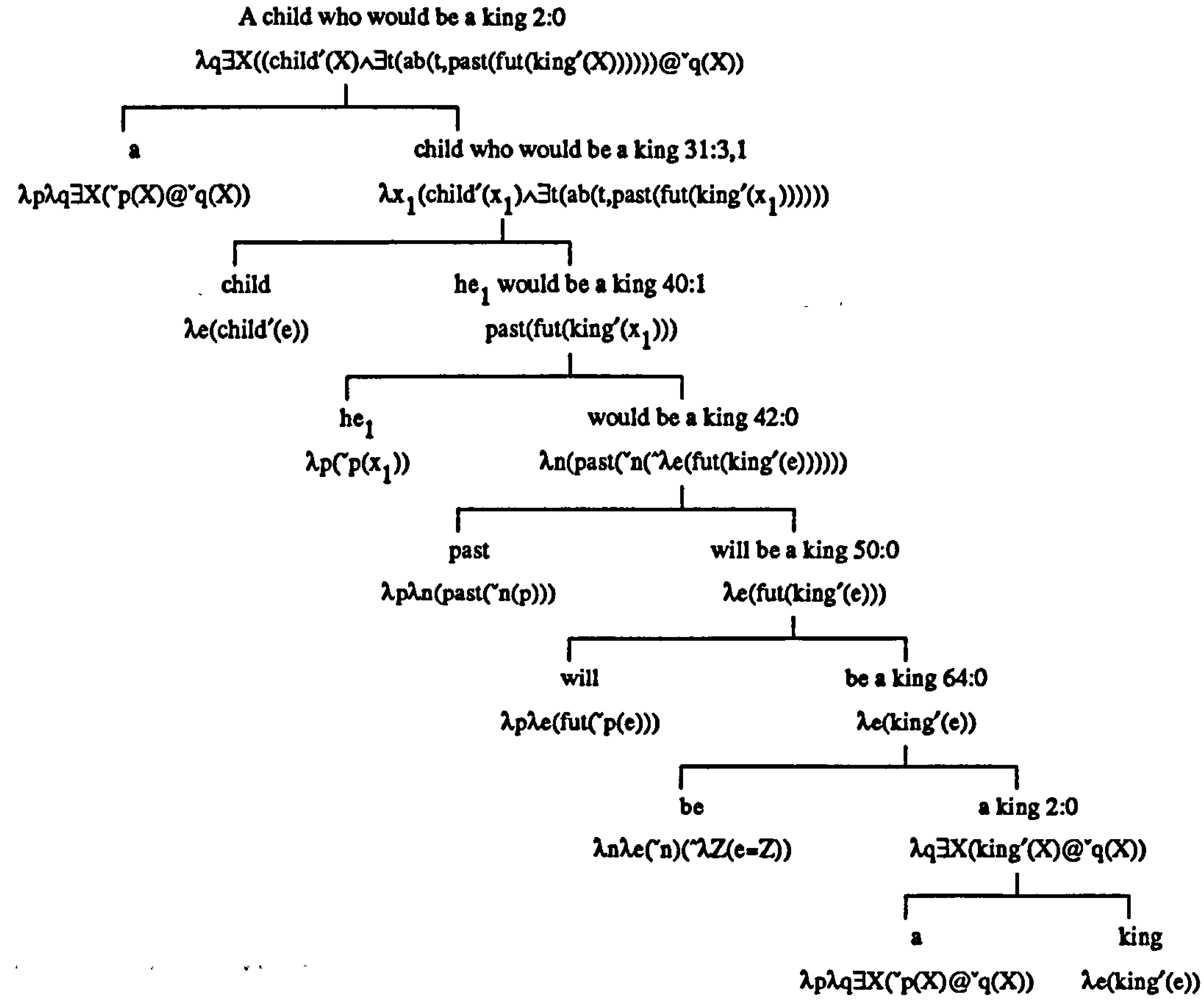


Fig 71

****

6.4.8. Quantified Verb Phrase Rules

Verb phrase quantification involves a processing premium; hence it is imperative to restrict it to situations where it is not semantically vacuous. Montague's original (S16) was designed to handle sentences like example (50) which involve *conjunctions*, nothing being achieved by quantification into elementary verb phrases. Verb phrase quantification in TMG is limited to conjoined lexical intransitive verb phrases and conjoined passive intransitive verb phrases, ie. to categories CLV and CPV.

(S16) If $\alpha \in P_T$ and $\delta \in P_{CLV}$ and SA16 is fulfilled then $f_{16,n}(\alpha,\delta) \in P_{CLV}$.

SA16: qsub(α,n,δ,γ) succeeds.

$f_{16,n}(\alpha,\delta) = \gamma$ such that qsub(α,n,δ,γ).

(T16) If $\alpha \Rightarrow \alpha'$ and $\delta \Rightarrow \delta'$ then $f_{16,n}(\alpha,\delta) \Rightarrow g_4(\alpha',\delta')$.

(S18) If $\alpha \in P_T$ and $\delta \in P_{CPV}$ and SA18 is fulfilled then $f_{18,n}(\alpha,\delta) \in P_{CPV}$.

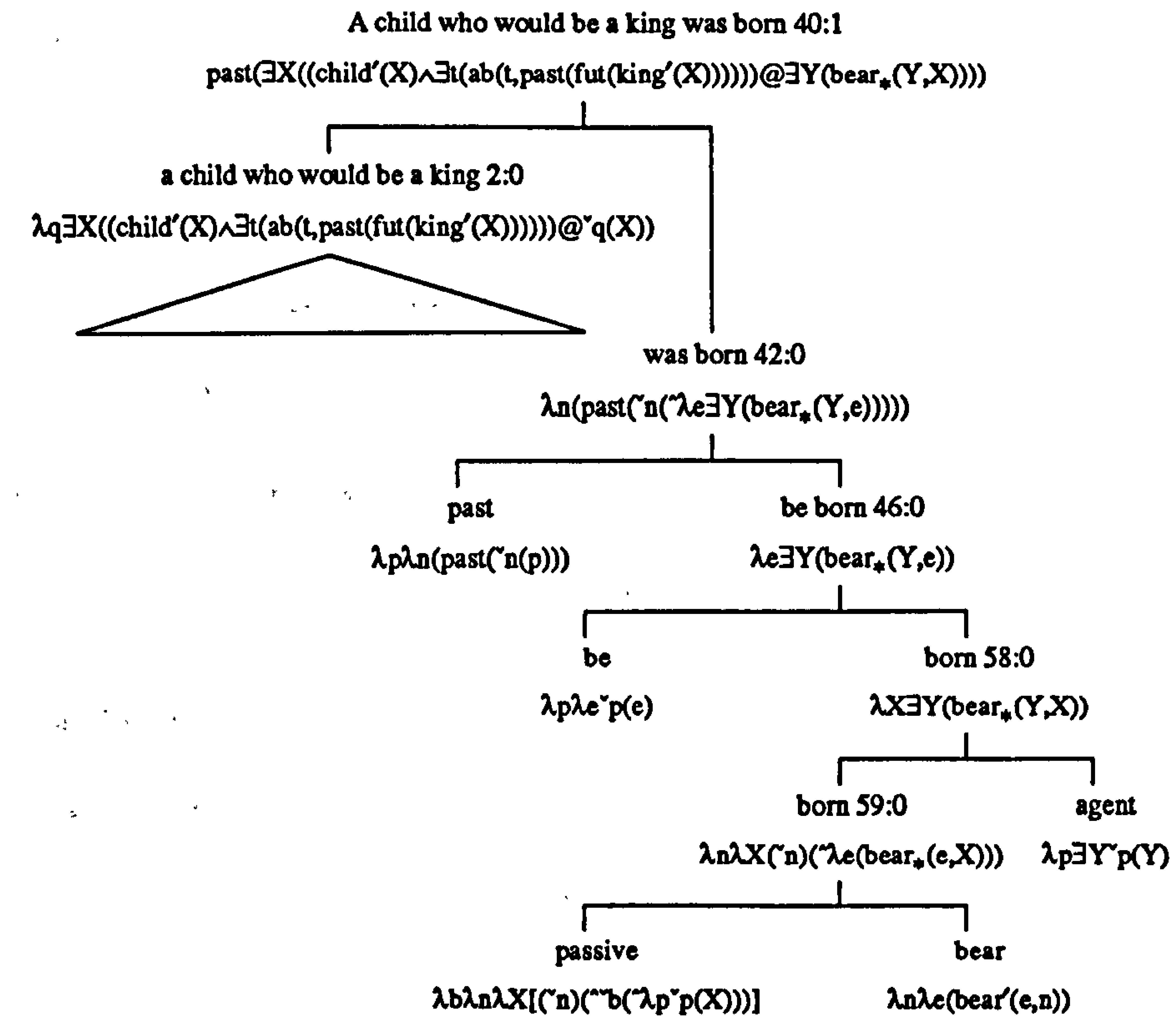


Fig 72

SA18: qsub(α,n,δ,γ) succeeds.

$f_{18,n}(\alpha,\delta) = \gamma$ such that qsub(α,n,δ,γ).

(T18) If $\alpha \Rightarrow \alpha'$ and $\delta \Rightarrow \delta'$ then $f_{18,n}(\alpha,\delta) \Rightarrow g_4(\alpha',\delta')$.

6.4.9. The Rules in Operation

All the rules of TMG are listed in appendix A, while appendix H contains illustrations of the vast majority as they appear in the output from the computational implementation. Sample TMG trees involving the new rules are also included as figs 68 ... 72.

Selected auxiliaries, passivisation, tense incorporation and temporal adverbial qualification appear in fig 68, which corresponds to appendix H example (H12), while fig 70, which is simulated by example (H13), illustrates the conjunction of the finite verb phrases introduced in fig 69. One tree for the test case (130) is represented in fig 72 which incorporates the complex noun phrase of fig 71: the final tree corresponds to the first TMDCG parse of (H-31).†82 Permitted quantification into verb phrases is shown in

†82. Certain trees in H-31 (eg. parse 5) require the complement of the copula to be introduced by quantification thus generating

(H-6) and (H-7), alternative modes of conjunction and adverbial qualification in (H-8) ... (H11).

Meaning postulates MP1 and MP8 survive inviolate in TMG; but the lexical modifications outlined in fig 64 necessitate the following amendments:

(MP4) $\forall n\Box[\lambda e\delta(e,n)\leftrightarrow\lambda e(\check{}n[\hat{}\lambda Y(\delta_*(e,Y))])]$

where $\lambda n\lambda e\delta(e,n)$ translates an extensional transitive verb and

$\delta_* =_{df} \lambda Y\lambda e(\delta(e,\hat{}\lambda p(\check{}p(Y))))$.

Typical triggering of MP4 is now at verb phrase and not sentence level (vide figs 68 ... 72).

(MP9) $\forall n\Box[\lambda e(seek'(e,n))\leftrightarrow\lambda e(try'(e,\hat{}\lambda Z(\check{}n[\hat{}\lambda Y(find_*(Z,Y))])))]$.

6.5. Multi-Indexed Tensed Logic.

The tensed logic referenced by TMG is no more than a modification of Dowty's extension to IL, [D6]; but for ease of consultation, and despite some inevitable repetition, this language TIL will now be formulated in full.

6.5.1. Semantic Types for TIL

The set *Type* of semantic types is the smallest set satisfying the recursive definition:

e ∈ *Type*.

t ∈ *Type*.

i ∈ *Type*.

If a, b ∈ *Type* then <ab> ∈ *Type*.

If a ∈ *Type* then <sa> ∈ *Type*.

6.5.2. Lexicon for TIL

For each type a ∈ *Type*:

(TILs1) $Var_a = \{v_{n,a} : n \geq 0\}$

(TILs2) $Con_a = \{c_{n,a} : n \geq 0\}$.

a somewhat bizarre interpretation that there exists a separately identifiable king which the child born was destined to be. Such a possibility must however be tolerated if we are to accept sentences such as "Samuel anointed a king and Saul was he".

where Var_a is the set of variables of type **a**, and Con_a is the set of (non logical) constants of type **a**. By convention $v_{0,i}, v_{1,i}, v_{2,i}$ ... may be abbreviated to t, t′, t″ ..., while $c_{0,i}$ is written as t* and $c_{1,i}$ as t@.

6.5.3. Syntax for TIL

The meaningful expressions of TIL are the members of $\cup_{a \in Type} ME_a$. defined recursively as follows:

(TILs3) If $\alpha \in Var_a$ then $\alpha \in ME_a$.

(TILs4) If $\alpha \in Con_a$ then $\alpha \in ME_a$.

(TILs5) If $\alpha \in ME_a$ and $\upsilon \in Var_b$ then $\lambda\upsilon\alpha \in ME_{<ba>}$.

(TILs6) If $\alpha \in ME_{<ab>}$ and $\beta \in ME_a$ then $\alpha(\beta) \in ME_b$.

(TILs7) If $\alpha, \beta \in ME_a$ then $\alpha=\beta \in ME_t$.

(TILs8) If $\zeta,\xi \in ME_i$ then $\ulcorner \zeta \subseteq \xi \urcorner$ and $\ulcorner \zeta < \xi \urcorner \in ME_t$.

(TILs9) If $\alpha \in ME_a$ then $\ulcorner {}^\wedge\alpha \urcorner \in ME_{<sa>}$.

(TILs10) If $\alpha \in ME_{<sa>}$ then $\ulcorner {}^\vee\alpha \urcorner \in ME_a$.

(TILs11) If $\Phi, \Psi \in ME_t$ then:

$\ulcorner \neg\Phi \urcorner \in ME_t$.

$\ulcorner (\Phi \wedge \Psi) \urcorner \in ME_t$.

$\ulcorner (\Phi @ \Psi) \urcorner \in ME_t$.

$\ulcorner (\Phi \vee \Psi) \urcorner \in ME_t$.

$\ulcorner (\Phi \rightarrow \Psi) \urcorner \in ME_t$.

$\ulcorner (\Phi \leftrightarrow \Psi) \urcorner \in ME_t$.

(TILs12) If $\Phi \in ME_t$ and for some a, $\upsilon \in Var_a$ then:

$\ulcorner \forall\upsilon\Phi \urcorner \in ME_t$.

$\ulcorner \exists\upsilon\Phi \urcorner \in ME_t$.

(TILs13) If $\Phi \in ME_t$ then:

$\ulcorner \Psi(\Phi) \urcorner \in ME_t$

where Ψ = *nec, poss, pres, past, fut, perf,* or *prog.*

(ILs14) If $\tau \in ME_i$ and $\Phi \in ME_t$ then

$\ulcorner at(\tau,\Phi) \urcorner \in ME_t$.

$\ulcorner ab(\tau,\Phi) \urcorner \in ME_t$.

There are no other members of ME_a besides those so defined.

6.5.4. Denotations for TIL

An intensional model has the form:

$$M = \langle D, W, T, Int, \leq, I \rangle$$

where D is a domain of possible individuals, W a set of possible worlds, T a set of moments in time ordered by $\leq$, $Int \subset Power(T)$ such that if $\mathbf{i} \in Int$ then for all $\mathbf{m}, \mathbf{m}', \mathbf{m}'' \in T$, if $\mathbf{m} \in \mathbf{i}$ and $\mathbf{m}'' \in \mathbf{i}$ and $\mathbf{m} < \mathbf{m}' < \mathbf{m}''$ then $\mathbf{m}' \in \mathbf{i}$, and I an interpretation function. For each type a the set *den*(a,*M*) of *possible denotations* of type a relative to *M* is:

$den(e,M) = D$.

$den(t,M) = \{0,1\}$.

$den(i,M) = Int$.

$den(\langle ab \rangle,M) = den(b,M)^{den(a,M)}$.

$den(\langle s,a \rangle,M) = den(a,M)^{W \times Int}$.

The set *sen*(a,*M*) of *senses* of type a relative to a model *M* is defined as:

$sen(a,M) = den(\langle sa \rangle,M)$

6.5.5. Multi Indexed Evaluation

A primary valuation structure $\langle M, G \rangle$ together with a secondary valuation structure $\langle V', \langle M, G \rangle\rangle$ are defined such that for any expression α:

$$V'(\alpha,M,w,\langle i,j,k,(h)\rangle,g) = [\![\alpha]\!]^{M,w,\langle i,j,k,(h)\rangle,g}$$

$$= \text{the extension of } \alpha \text{ at } \langle w,i \rangle \text{ with respect to } M.$$

The quadruple $\langle i, j, k, (h) \rangle$ represents four intervals of which k is the speech point and i the event point. At the nth stage of an evaluation the index j^n is to constitute the current intermediate reference point, its relationship to its immediate predecessor having been specified by an appropriate semantic rule.

Evaluation must commence in an "utterable context"†83 $<i, j, k, (h)>$ such that $j = k$, ie. the intermediate reference point on initiation must be equated with the speech point. Thereafter successive stages in the interpretation of a time sensitive formula will introduce alternative intermediate reference points, each relatively defined, until a nucleus, (untensed), predication is discovered. When this nucleus is evaluated, the current intermediate reference point is identified as the interval containing the event point i which thus becomes determined.

Although in the simplest cases k will behave as an input parameter, i as an output parameter and j as a local cursor, the order in which values become crystalised is in fact immaterial, ie. the variables may be conceived as polymodal. A successful evaluation involves discovering sequences of compatible values for $<i, j, k>$ triples: moreover at any given stage the constants t* and t@ will denote i and k respectively.

The parenthetical index (h) is an *associated* event point which becomes set only in the context of the Cresswellian conjunctive "@", to which we shall presently revert. As will be seen from rule (TILt11), the parenthetical index is used to "remember" the event time of a coordinate sentence, information which proves invaluable in handling subordinate clause time relativisation.

6.5.6. Semantics for TIL

(TILt1) If $\upsilon \in Var_a$ and $g \in G$ then $g(\upsilon) \in den(a,M)$.

(TILt2) If $\alpha \in Con_a$ then $I(\alpha) \in sen(a,M) = den(a,M)^{W \times Int}$.

In particular $I(t^*)(w,i) = i$.

(TILt3) If $\upsilon \in Var_a$ then $[\![\upsilon]\!]^{M,w,<i,j,k,(h)>,g} = g(\upsilon)$.

(TILt4) $[\![t@]\!]^{M,w,<i,j,k,(h)>,g} = [\![t^*]\!]^{M,w,<k,k,k,(h)>,g}$ otherwise

If $\alpha \in Con_a$ then $[\![\alpha]\!]^{M,w,<i,j,k,(h)>,g} = I(\alpha)(w,i)$.

(TILt5) If $\alpha \in ME_a$ and $\upsilon \in Var_b$ then $[\![\lambda\upsilon\alpha]\!]^{M,w,<i,j,k,(h)>,g} = \eta \in den(a,M)^{den(b,M)}$

such that for all $\beta \in den(b,M)$, $\eta(\beta) = [\![\alpha]\!]^{M,w,<i,j,k,(h)>,g'}$

where g and g' differ *at most* in that $g'(\upsilon) = \beta$.

†83. This concept is introduced by Dowty, [D8], in his tentative development of a triple indexed system.

(TILt6) If $\alpha \in ME_{\langle ab\rangle}$ and $\beta \in M_a$ then, for $i \subseteq j$, $[\![\alpha(\beta)]\!]^{M,w,\langle i,j,k,(h)\rangle,g} =$

$[\![\alpha]\!]^{M,w,\langle i,j,k,(h)\rangle,g}([\![\beta]\!]^{M,w,\langle i,j,k,(h)\rangle,g})$.

(TILt7) If α and $\beta \in ME_a$ then $[\![\ulcorner \alpha=\beta \urcorner]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$

iff $[\![\alpha]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = [\![\beta]\!]^{M,w,\langle i,j,k,(h)\rangle,g}$, 0 otherwise.

(TILt8) $[\![\ulcorner \zeta\subseteq\xi \urcorner]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$ iff $[\![\zeta]\!]^{M,w,\langle i,j,k,(h)\rangle,g} \subseteq [\![\xi]\!]^{M,w,\langle i,j,k,(h)\rangle,g}$.

$[\![\ulcorner \zeta<\xi \urcorner]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$ iff for all $m \in [\![\zeta]\!]^{M,w,\langle i,j,k,(h)\rangle,g}$

and all $m' \in [\![\xi]\!]^{M,w,\langle i,j,k,(h)\rangle,g}$, $m<m'$.

(TILt9) If $\alpha \in ME_a$ then $[\![\ulcorner {}^{\wedge}\alpha \urcorner]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = \eta \in den(a,M)^{W\times Int}$ such that,

for all $\langle w,i\rangle \in W\times Int$, $\eta(w,i) = [\![\alpha]\!]^{M,w,\langle i,j,k,(h)\rangle,g}$.

(TILt10) If $\alpha \in ME_{\langle sa\rangle}$ then $[\![\ulcorner {}^{\vee}\alpha \urcorner]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = [\![\alpha]\!]^{M,w,\langle i,j,k,(h)\rangle,g}(w,i)$.

(TILt11) If Φ and $\Psi \in ME_t$ then:

$[\![\ulcorner \neg\Phi \urcorner]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$ iff $[\![\Phi]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 0$.

$[\![\ulcorner (\Phi\wedge\Psi) \urcorner]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$ iff both $[\![\Phi]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$

and $[\![\psi]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$.

$[\![\ulcorner (\Phi@\Psi) \urcorner]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$ iff for some i′,i″ $[\![\Phi]\!]^{M,w,\langle i',j,k,(i'')\rangle,g} = 1$

and $[\![\Psi]\!]^{M,w,\langle i'',j,k,(i')\rangle,g} = 1$ and $i' \subseteq i$ and $i'' \subseteq i$.

$[\![\ulcorner (\Phi\vee\Psi) \urcorner]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$ iff either $[\![\Phi]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$

or $[\![\psi]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$.

$[\![\ulcorner (\Phi\rightarrow\Psi) \urcorner]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$ iff either $[\![\Phi]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 0$

or $[\![\psi]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$.

$[\![\ulcorner (\Phi\leftrightarrow\Psi) \urcorner]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$ iff

$[\![\Phi]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = [\![\Psi]\!]^{M,w,\langle i,j,k,(h)\rangle,g}$.

(TILt12) If $\Phi \in ME_t$ and for some a, $\upsilon \in Var_a$ then:

$[\![\ulcorner \forall\upsilon\Phi \urcorner]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$ iff

$[\![\Phi]\!]^{M,w,\langle i,j,k,(h)\rangle,g'} = 1$ for all g' that are υ-variant to g.

$[\![\ulcorner \exists\upsilon\Phi \urcorner]\!]^{M,w,\langle i,j,k,(h)\rangle,g} = 1$ iff

$[\![\Phi]\!]^{M,w,<i,j,k,(h)>,g'} = 1$ for some g' that is υ-variant to g.

(TILt13) If $\Phi \in ME_t$ then:

$[\![\ulcorner nec\Phi)\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ iff $[\![\Phi]\!]^{M,w',<i,j,k,(h)>',g} = 1$

for all $w' \in W$ and all i', j', k' and $h' \in Int$.

$[\![\ulcorner poss\Phi)\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ iff $[\![\Phi]\!]^{M,w',<i,j,k,(h)>',g} = 1$

for some $w' \in W$ and some i', j', k' and $h' \in Int$.

$[\![\ulcorner pres(\Phi)\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ iff $[\![\Phi]\!]^{M,w,<i,j',k,(h)>,g} = 1$ and $j'=j$.

$[\![\ulcorner past(\Phi)\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ iff $[\![\Phi]\!]^{M,w,<i,j',k,(h)>,g} = 1$ and $j'<j$.

$[\![\ulcorner fut(\Phi)\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ iff $[\![P]\!]^{M,w,<i,j',k,(h)>,g} = 1$ and $j'>j$.

$[\![\ulcorner perf(\Phi)\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ iff $[\![\Phi]\!]^{M,w,<i,j',k,(h)>,g} = 1$

and j is a final subinterval of j'.

$[\![\ulcorner prog(\Phi)\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ iff $[\![\Phi]\!]^{M,w,<i,j',k,(h)>,g} = 1$ and $j \subseteq j'$.

(TILt14) $[\![\ulcorner at(\tau,\Phi)\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ iff $[\![\Phi]\!]^{M,w,<' j,k,(h)>,g} = 1$

and $i' = [\![t]\!]^{M,w,<i,j,k,(h)>,g}$.

$[\![\ulcorner ab(\tau,X)\urcorner]\!]^{M,w,<i,j,k,(h)>,g}$ is defined by the *first* matching case from the following:

$[\![\ulcorner ab(\tau,\Phi\wedge\Psi)\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ iff $[\![\ulcorner ab(\tau,\Phi)\wedge ab(\tau,\Psi)\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$.

$[\![\ulcorner ab(\tau,\Phi@\Psi)\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ iff $[\![\exists t(ab(\tau,\Phi))\wedge\exists t(ab(\tau,\Psi))\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$.

$[\![\ulcorner ab(\tau,\Sigma(\Phi))\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ iff $[\![\Sigma(ab(\tau,\Phi))\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ where Σ is a quantifier.

$[\![\ulcorner ab(\tau,at(\tau',\Phi))\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ iff $[\![\ulcorner ab(\tau',\Phi)\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$.

$[\![\ulcorner ab(\tau,pres(\Phi))\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ iff $[\![\ulcorner ab(\tau,\Phi)\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$.

$[\![\ulcorner ab(\tau,fut(\Phi)\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ iff $[\![\ulcorner fut(\Phi)\urcorner]\!]^{M,w,<i',j',k,(h)>,g} = 1$

where $j' = [\![\tau]\!]^{M,w,<i,j,k,(h)>,g} \geq [\![t@]\!]^{M,w,<i,j,k,(h)>,g}$.

$[\![\ulcorner ab(\tau,past(\Phi))\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1$ iff either

(i) $h<k$ and $[\![\Phi]\!]^{M,w,<i',j',k,(h)>,g} = 1$ with $h \subseteq j' = [\![\tau]\!]^{M,w,<i,j,k,(h)>,g}$

(ii) $h \geq k$ and $[\![\ulcorner past(\Phi)\urcorner]\!]^{M,w,<i',j',k,(h)>,g} = 1$ where $j' = [\![\tau]\!]^{M,w,<i,j,k(h)>,g}$ $\leq [\![t@]\!]^{M,w,<i,j,k,(h)>,g}$.

$$[\![\ulcorner ab(\tau,\text{Other})\urcorner]\!]^{M,w,<i,j,k,(h)>,g} = 1 \text{ iff } [\![\text{Other}]\!]^{M,w,<i',j,k,(h)>,g} = 1$$

$$\text{where } j = [\![\tau]\!]^{M,w,<i,j,k,(h)>,g}.$$

Finally a formula Φ is true at speech point k iff for some i $[\![\Phi]\!]^{M,w,<i,j,k,(h)>,g} = 1$.

The symbol "@" introduced in (TILt11) represents Cresswell's alternative to the conventional Boolean conjunctive "$\wedge$". Cresswell observes, [C12], that the Boolean "and" is ill equipped to handle tensed constructions since its normal semantics would require each conjunct in a conjoined sentence to be true for simultaneous event times. In an example such as:

(148) One day last week John came and Mary went.

the only constraint imposed upon the coming and going is that both occur within the interval "one day last week". Hence, according to Cresswell's semantics, $\ulcorner(\Phi @ \Psi)\urcorner$ should be true at an interval t iff Φ is true at t', Ψ is true at t'', and the smallest m such that $t' \subseteq m$ and $t'' \subseteq m$ is t. The appropriate sub clause of (TILt11) may be seen as a multi indexed variation of this insight.

Singular determiners in TMG are defined in terms of the Cresswellian rather than the Boolean conjunctive so that in need the evaluation times for the restriction "$^{\vee}p(X)$" and the body "$^{\vee}q(X)$" may differ, ie:

a $\twoheadrightarrow$ $\lambda p \lambda q \exists X(^{\vee}p(X) @ ^{\vee}q(X))$.

the $\twoheadrightarrow$ $\lambda p \lambda q \exists Y(\forall X(^{\vee}p(X) \leftrightarrow X = Y) @ ^{\vee}q(Y))$.

Since conjoined verb phrases also admit temporal tolerance of the kind associated with example (148), ie. they may refer to non simultaneous times within implied or specified limits, the semantic operations invoked by (S70), (S71), (S72) and (S73) are likewise defined in Cresswellian terms.

6.5.7. Subordinate Time Relativisation

Although still suitable for crystallising the event point for sentences with time specified by an adverbial, Dowty's "at" operator will no longer serve to reset the reference time in subordinate clauses since it determines an ultimate event point not an intermediate reference point. Reference time for subordinate clauses is handled by a new "ab" (ab initio) operator introduced via semantic operation $g_{3,n}$ which is invoked by the relative clause translation rules (T30) and (T31). This operator resets the current intermediate reference point j subject to certain limitations designed to eliminate unacceptable orderings, and to contribute to the observation of sequence of tense restrictions. The first argument to "ab" is an interval variable which becomes set to an appropriate value for *j* and the second a subordinate clause representation.

Of the clauses which define the semantics for "ab" in (TILt14), the majority serve merely to pass the operator down to a level where it will obtain purchase or to eliminate it altogether when it becomes vacuous. The clauses of substance are those in which the second argument of "ab" is governed by a "fut" or a "past" operator.

The requirement if the subordinate structure is governed by "fut"†84 is that the intermediate reference point be initialised to some interval equal to or subsequent to the point of speech: in this way the restriction required by sentences such as (129) is maintained.

When the embedded formula is governed by "past" two situations may arise. The first, which is exemplified by example (130), occurs when the matrix clause is also in a past tense in which case the parenthetical index *h* will acquire a value earlier than the point of speech *k*. The requirement in this case is that the intermediate reference point be initialised to an interval including the matrix event point. Alternatively the matrix clause may be governed by a "pres" operator as in:

(149) A man who was a traitor will die.†85

If the parenthetical index indicates an interval equal to or subsequent to the speech point, the intermediate reference point must be initialised to no later than the speech point identifiable as t@. Were *j* to commence later than t@ then the treachery reported in (149) might turn out to be anticipated.

The capabilities of the system may be illustrated by reference to example (130) when parsed according to the tree of fig 72. In the following derivation only the salient time indices are indicated, the balance of the superscription being omitted for clarity:

$[\![\mathrm{past}(\exists X((\mathrm{child}'(X)\wedge\exists t(\mathrm{ab}(t,\mathrm{past}(\mathrm{fut}(\mathrm{king}'(X))))))@\exists Y(\mathrm{bear}_*(Y,X)))]\!]^{i,j,k,(h)} = 1$ iff

$[\![\exists X((\mathrm{child}'(X)\wedge\exists t(\mathrm{ab}(t,\mathrm{past}(\mathrm{fut}(\mathrm{king}'(X))))))@\exists Y(\mathrm{bear}_*(Y,X))]\!]^{i,j',k,(h)} = 1$ with $j'<j$ iff

for suitable variable assignments:

†84. The case in question actually involves "fut" within the scope of "pres", but the "ab" is simply passed over any "pres" operator.

†85. There is of course strictly speaking no future tense in English. The matrix clause here has a modal auxiliary "will" marked [+present].

****

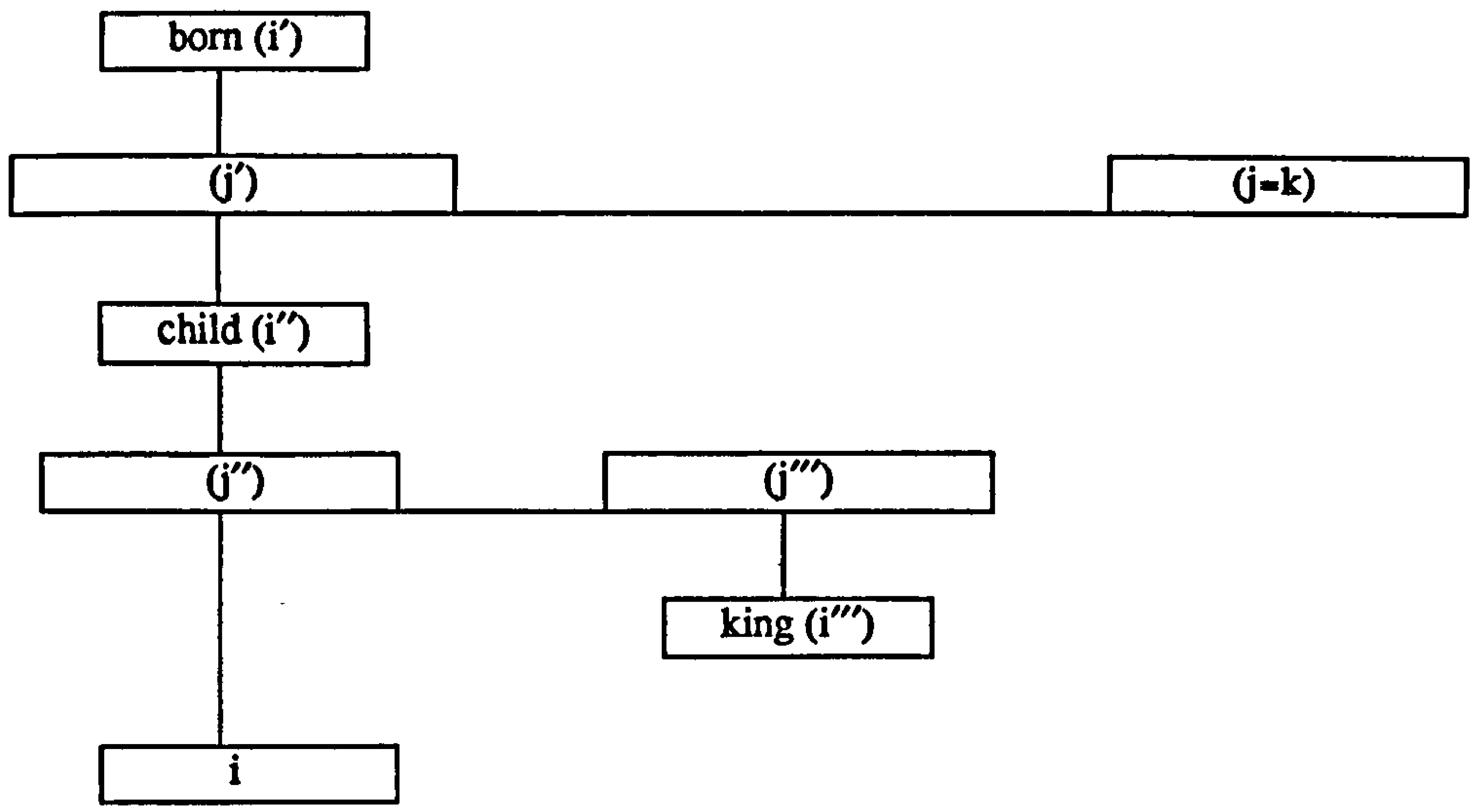


Fig 73

****

{ $[\![\text{bear}_*(Y,X)]\!]^{i',j',k,(i')} = 1$ with $i' \subseteq i$ and $i' \subseteq j'$ and

$[\![\text{child}'(X) \wedge \exists t(\text{ab}(t,\text{past}(\text{fut}(\text{king}'(X)))))]\!]^{i'',j',k,(i')} = 1$ with $i'' \subseteq i$ iff

{ $[\![\text{child}'(X)]\!]^{i'',j',k,(i')} = 1$ with $i'' \subseteq j'$ and

for a suitable $t = j''$:

$[\![\text{ab}(t,\text{past}(\text{fut}(\text{king}'(X))))]\!]^{i'',j',k,(i')} = 1$ iff

CASE (i): $i' \subseteq j' < j = k$:

$[\![\text{fut}(\text{king}'(X))]\!]^{i''',j'',k,(i')} = 1$ with $i' \subseteq j''$ iff

$[\![\text{king}'(X)]\!]^{i''',j''',k,(i')} = 1$ with $j''' > j''$ and $i''' \subseteq j'''$ } }.

The relationships between i, i', i'', i''', j, j', j'', j''' and k identified in this derivation are depicted on the "time map" in fig 73.