Issues of Semantics in a Semantic-Network Representation of Belief

by
Robin K. Hill
May 7, 1994

A dissertation submitted to the
Faculty of the Graduate School of the
State University of New York at Buffalo
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Dedicated to
Eric S. Dietrich

Acknowledgments

My gratitude for help in the development of this work is due primarily to Bill Rapaport, whose steady encouragement made all the difference, and to Eric Dietrich, always ready to share trials and tribulations, and provide advice and motivation. I would also like to sincerely thank Stu Shapiro and Ken Regan for significant contributions at critical points. Along with them, there are other mentors and inspirations who deserve a mention—Yorick Wilks, who first mentioned non-well-founded set theory to me, Lawrence S. Moss, who make some helpful suggestions in his review, and my family and friends, who will be glad to go on to topics of conversation other than my lurching progress toward a Ph.D, and, for aesthetic inspiration, Carmina Burana and Craig Langager.

Abstract

Knowledge representation and reasoning systems that are used for cognitive modeling must capture mental intensions, i.e., *senses* of linguistic or other semantic constructions, as opposed to (or, rather, in addition to) *references*, such as physical objects, of those constructions. A semantic-network system used for this purpose needs a semantics in which the nodes of a network are terms in the language of thought of the modeled cognitive agent and represent entities in the agent's mental universe. SNePS ("Semantic Network Processing System") is such a system.

With SNePS as the framework, specific issues studied include how different types of nodes (concepts) are to be interpreted and used by the cognitive agent and the different types of computations required to support these uses; how nodes come to be distinct or similar, from both inter-agent and intra-agent perspectives; how (and why) circularity of meaning may be accommodated in a semantic network; and to what extent the coherence of the cognitive structure of the agent may be maintained under such circularity.

Non-well-founded set theory, with its legitimization of truly circular structures, is of special interest to providing a semantics for SNePS in accordance with its design principles. A certain category of SNePS node, the *base node*, representing a discrete concept, is given a semantics that is both influenced by and influences its dominating compound (*molecular*) nodes, lending a controlled cyclicity to networks. A semantic function μ is defined that assigns a "hyperset", or non-well-founded set, formed from the outermost *sensory* nodes of the cognitive agent, to each other node of that agent. Investigation shows that this hyperset semantics supports representational principles of SNePS, such as the conceptual uniqueness of every node, and allows no node to be circular to the point of vacuity. Further discussion shows how this contribution from SNePS may illuminate the role that semantic networks and graphical representation in general have to play in artificial intelligence.

Contents

List of Definitions

List of Theorems

List of Figures

Chapter 1

The Problem and Its Significance

1.1 Introduction

Is knowledge computational? Assuming so, how is the computation done? These questions are central to artificial intelligence and to some areas of theoretical computer science; they are affected by *every* area of computer science that participates in defining 'computation'. One way to investigate possible answers is to ask the questions in terms of a particular knowledge/belief representation scheme, or cognitive model. To investigate knowledge in terms of such a model is partly to investigate the *semantics* of the model, the meanings of the structures in it. If we can formulate the features of knowledge as we understand it, then a model's claim to worthiness can be measured by its incorporation of those features. Conversely, if we can see what semantics are appropriate for the model, then we see what sort of "knowledge" it can capture. The intent of this chapter is to pose those questions in terms of a particular cognitive model, the semantic network, and in terms of a specific semantic network, SNePS ("Semantic Network Processing System").

1.2 Knowledge and Intensionality

Knowledge as we understand it (so to speak) is no simple matter. Human cognitive capacity does not shirk at complex interactions among the real, the unreal, and the dubious. Consider the classical fable of Mentor/Athena and Telemachus, part of the *Odyssey*:

> In Greek mythology, [Mentor was] the faithful friend of Odysseus, king of Ithaca, who entrusted to Mentor the care of his household during his absence in the Trojan War. Mentor was a guardian and tutor of Telemachus, Odysseus' son, whom the goddess Athena (assuming Mentor's form and acting as guide and prudent adviser) accompanied in the search for Odysseus after the war. [Encyclopedia Americana, 1980, "Mentor" (vol. 18, page 651)]

From direct allusions to the modern use of the term 'mentor', we can see that this story and its fellows have a rich psychological role to play in classically influenced western civilization. AI researchers would be interested to know, therefore, what sort of mental model can compute things such as the following:

- robust yet *different* concepts of Athena and Mentor, where the latter is the former in disguise, not to be confused with the independent concept of Mentor, the original individual.
- Athena's perception of Telemachus's concept of Mentor, necessary for her successful guidance of him.

- indeed, any respectable concept of Athena at all, who is both mythical and a deity, or even of Telemachus, who is merely mythical.

To require the model to be *computational*—and stretching the notion of computation to reach cognitive science—is to require the capacity to employ these concepts in thoughts, actions, and in the development of further concepts. Classical formal systems such as the predicate calculus are not computational. They are not concerned with the algorithms that would be employed, in this example, to answer questions such as: (1) Why does Telemachus follow Mentor? (2) How is the Greek conception of Athena, and the qualities that she is presumed to embody, enriched by this myth? (3) How would Telemachus react if Mentor revealed herself to be Athena? (4) What lessons does the story hold? A computational model must answer these questions in the face of the complications of non-existence, hypothesis, and meta-conceptualization outlined above. Since human cognition manages this, viable artificial cognition must also do so.

It must do so, that is, on the assumption that complex interactions among the real, the unreal, and the dubious are critical to intelligence. The consensus of the AI community appears to be that a major goal of artificial intelligence is the achievement of all of the effects of human intelligence, no matter whether or not the methods that produce these effects are the same as those employed by humans. Yet sometimes it is not clear if some aspect of human cognition is an effect or a method. Consider the phenomena above. Is the conceptualization of Telemachus's concept of Mentor an integral part of intelligence, and therefore required by AI, or, rather, a method of achieving some intelligent behavior that could be achieved in some other way, without any such conceptualization? (Is conceptualization an unnecessary middleman?) The lack of decisive evidence as to whether conceptualization is a means or an end justifies the working assumption that it is indeed an end, an assumption that frees us from interminable debate over the utility of capturing this (or various other) aspects of human cognition. This work makes that assumption.

The use of the knowledge representation and reasoning system SNePS ("Semantic Network Processing System") for cognitive modeling takes as its fundamental constructive principle that representation of all concepts (things, propositions, and all other phenomena) is done through nodes in a network, thereby providing a circumscribed set of meaning-bearing objects as defined in [Shapiro, 1979; Shapiro and Rapaport, 1987; Shapiro and Rapaport, 1991]. It embraces *intensionality*, as opposed to otherwise similar approaches, as discussed briefly in the introduction to [Maida and Shapiro, 1982] in [Brachman and Levesque, 1985, page 169]. Some other well-known knowledge representation systems, like KL-ONE [Brachman and Schmolze, 1985], also incorporate intensionality (as discussed below).

Intensionality in philosophy of mind is, very roughly,

> that set of features which concerns the meaning of a term as against the things to which it applies [Lacey, 1976, page 98]

For example, the distinction between Mentor and Athena above is intensional only (not *extensional*, which would be "the things to which that term applies"). *Intension* is close to Frege's *sense*, and *extension* to his *reference*.

Intensionality in a knowledge representation system can be described as the treatment of all concepts as first-class objects [Hirst, 1991]. In programming language theory, a first-class object is one that is subject to the widest possible variety of operations, including storage and retrieval, communication as a parameter, binding to a variable, use of its value in computations, and so forth. Whereas a traditional approach to knowledge would regard imaginary, hypothetical, or contradictory concepts as somehow pathological, and thus limit their use to special contexts, an intensional model grants them the full range of considerations. Nodes that represent imaginary, hypothetical, or contradictory concepts are peers of nodes that represent real (or possible) and extensional objects. If something can be talked or thought about, then it is a concept, even if not "in" any cognitive agent.

The proper interpretation of a SNePS node is therefore an intension, or concept. The arcs that connect nodes contribute no conceptual semantics; they are "punctuation" only. Furthermore, the interpretation of a node is affected by connecting nodes (other concepts), which may in turn influence it. Relief from this circularity is provided by sensory nodes, which are external inputs like lexemes and mechanical outputs like actions [Shapiro and Rapaport, 1987; Rapaport, 1988b; Shapiro and Rapaport, 1991].

To say that a node represents a concept is not sufficient. We need to talk about *which* concept, how the concept can be described, and how it relates to others, and how it can be distinguished from them. We need to know something about the meaning of the node in ordinary words suitable for discourse with other cognitive agents, but we need to find this meaning without subverting intensionality by placing the burden wholly on reference. We see, then, that the incorporation of intensionality is not a quick fix motivated by a few cognitive riddles, and that it both illuminates and complicates semantic issues.

1.3 Knowledge Representation and Reasoning; SNePS

1.3.1 History

For many years, logic, especially the first-order predicate calculus, was the method of choice for representing cognitive activity. A growing dissatisfaction with the results led some researchers to blame the method, as expressed, for example, by Marvin Minsky [Minsky, 1981]. A popular alternative was the graphical model, instantiated in AI as the "semantic network" with its apparent greater flexibility and its presumed analogy to neural structure. Undisciplined application of this approach raised the problems discusssed by William A. Woods, who called on semantic network modelers to provide a theory grounding their use of the constructs [Woods, 1975].

No matter what the model chosen, standard extensional ways of assigning meanings to the terms involved have proven unsatisfactory. Their interpretation as external objects and phenomena in the real world, for example, does not allow direct representation of fictional, imaginary, or impossible entities.

The development of SNePS has specifically addressed these challenges. The question of what exactly the structuring devices represent was answered by Stuart C. Shapiro [Shapiro, 1979]: the labeled, directed arcs mean nothing, since all representation is in the nodes; arcs have *structural* significance only. The domain being represented is the *intensional* world—those things that are conceivable, regardless of their status in reality [Maida and Shapiro, 1982; Rapaport, 1985b; Shapiro and Rapaport, 1987; Shapiro, 1991; Shapiro and Rapaport, 1991; Rapaport, forthcoming]. This perspective defines a mental model relative to a particular cognitive agent at some point in time. Of particular interest here is that the meaning of a node follows the paradigm given by M. Ross Quillian in an important early paper [Quillian, 1968], which defines the "full concept" of a word node as the entire network connected to it.

But logic is indispensable, and it is also an integral facility of SNePS. The treatment of quantifiers defined by Shapiro [Shapiro, 1979] enables the representation of the predicate calculus within the mind of the cognitive agent, and therefore its full use for reasoning by that agent. Woods [Woods, 1975] stated that the forerunner of SNePS was unique (at the time) in its correct incorporation of the facilities of the predicate calculus. SNePS uses generalized logical connectives such as *and/or* (which, applied to a set of sentences, specifies how many are true) and *thresh* (for "threshold") to do the work of standard and non-standard inference rules [Shapiro, 1978; Martins and Shapiro, 1988]. The mechanism that allows for inference rules conceptualized as nodes to be used in reasoning is called *node-based inference*.

Ernesto Morgado [Morgado, 1986] provided a theoretical grounding for an earlier version of SNePS in terms of abstract data types. This rigorous mathematical definition, using abstract algebras, gives us recursive definitions of SNePS constructs such as cables and networks. The exploration of SNePS continues as described below. Fundamental definitions and mechanisms are given in § 2.5.

1.3.2 Relevance to current complementary work in AI

Much of the historical discussion above is manifest in current SNePS research. To serve as a model of cognition, SNePS must have the capacity to draw conclusions, which involves building new proposition nodes and/or asserting existent ones. The means available to do this in implementation software currently include forward and backward inference with the SNePS Inference Package [Shapiro *et al.*, 1982; Hull, 1986] and context-relative belief revision through SNeBR [Martins and Shapiro, 1988]. Some research treats concepts as actions to be taken [Kumar *et al.*, 1988], leading to the question whether the *act* of a cognitive agent is a special semantic manifestation. (In other words, will *act* nodes require special interpretation mechanisms?)

Recent work concerning subconscious reasoning and the assertional status of propositional nodes suggests that a possible semantics of a network is "derived by taking all the atomic formulae conjunctively, and path-based inference rules are straight-forward conditionals" [Shapiro, 1991, page 3], and considers the inference mechanisms appropriate to this view. Meta-predicates are used to control deduction, separating, for instance, subconscious from conscious belief. The paper cited shows that SNePS is not reducible to first-order logic.

Individual nodes are meant to have intensional interpretations, so that everything, including imaginary and impossible objects, can have a place in a mental model [Shapiro and Rapaport, 1987; Shapiro and Rapaport, 1991]. An appropriate domain is developed in the work of the philosopher Alexius Meinong, who called it *Aussersein*. Its use for the semantics of semantic network representation systems in general, and SNePS in particular has been explored in [Rapaport, 1978; Rapaport, 1981; Rapaport, 1985b; Rapaport, forthcoming].

Another well-known approach to knowledge representation and reasoning with semantic networks is the KL-ONE family. Dependent on taxonomy and subsumption, it makes properties, rather than propositions, primary. The basic unit of cognition is the "structured conceptual object", or simply "Concept", the components of which are the "superConcepts" that subsume it, and the "Roles" and "Structural Description" that together constitute restrictions on the superConcepts. KL-ONE is also intended to represent intensionally [Brachman and Schmolze, 1985].

Any results derived herein that depend on the intensionality (only) of SNePS nodes would provide a rigorous distinction between it and non-intensional representations. Indeed, all considerations here could affect our understanding of the semantic network model in general, insofar as types of semantic networks contrast with each other, and as semantic networks contrast with other cognitive models. For instance, a traditional argument in this area concerns the existence and degree of qualitative differences between the various forms of representation used in artificial intelligence. In particular, the claim that a semantic network is redundant to first-order logic has been revived by Lenhart Schubert:

> *All* KR schemes I have lately encountered, which aspire to cope with a large, general propositional knowledge base, qualify as semantic nets, appropriately viewed. [Schubert, 1991, page 96; italics his]

He proposes to reserve the term 'semantic net' for models that are truly dependent on graph-theoretic characteristics—topological, dynamic, or other. The study described here seeks such characteristics of SNePS. The question is of great importance for the entire field of knowledge representation.

Brian Cantwell Smith captures the common view that knowledge must be representational in his *Knowledge Representation Hypothesis*, which suggests that success in designing a cognitive model—perhaps a semantic network; perhaps, indeed, SNePS—is both necessary and sufficient to the endeavor of artificial intelligence [Smith, 1982]. From Smith also comes a provocative paper calling for a new focus, in AI research, on the middle ground between theory and prototype, to be discussed later in this chapter. Schank and Rieger take as the goal for their knowledge representation offering, Conceptual Dependency Theory, the capturing of natural-language input to a degree sufficient to draw appropriate conclusions from it [Schank and Rieger, 1974]. These works, only a tiny sampling from an active area of research (for a survey, see [Brachman and Levesque, 1985]), show the importance to artificial intelligence of knowledge/belief representation in general, and of the representation of knowledge conveyed through natural language in particular.

SNePS has a commitment to natural language as the interface and therefore as a *de facto* functional definition of the system [Shapiro and Rapaport, 1987; Shapiro and Rapaport, 1991]. The interpretation and generation of natural languages is a vast and active area of artificial intelligence research. We need to address the following aspect of it: How might it affect the computational semantics of an arbitrary node?

One tradition in the study of language acquisition uses a computer as the vehicle, implementing some algorithm to establish an association between input and output groups of structures, which are variously considered syntactic and/or semantic. A modern exemplar is Jeffrey Siskind, who presents an operational system that acquires some semantics along with syntax, building structures embodying restricted word meanings from a stream of both linguistic and restricted empirical input, both in symbolic form [Siskind, 1990]. This type of study is complementary, rather than parallel, to the present concern.

1.4 Some Specific Motivational Questions

Suppose we intend to use a knowledge representation and reasoning system such as SNePS to model the mind of a cognitive agent, as in [Shapiro and Rapaport, 1987]. Consider a SNePS node n, and consider its meaning $[\![n]\!]$, where "meaning" is to be taken here in the sense of [Shapiro, 1991] (which is the source of the notation) as an undefined intuitive term. If $[\![\texttt{n47}]\!]$ is "the trouble with the New York State Legislature", then $[\![\text{the trouble with the New York State Legislature}]\!]^{-1}$ is `n47` (employing the standard notation for the inverse function).[1] The assertion of a proposition node n, signifying the cognitive agent's belief in the truth of $[\![n]\!]$, is given by '!', a unary predicate; so $n!$ is the agent's believing in $[\![n]\!]$. Some of the particular questions to be addressed are given in the examples that follow. Let C and D be any two cognitive agents implemented in SNePS.

1. Suppose $[\![n]\!]$ is C's concept of Perdita's old car. Suppose $[\![m]\!]$ is D's concept of Perdita's old car. What makes $[\![n]\!]$ and $[\![m]\!]$ the same thing, extensionally—the same external object? Remember that n and m themselves are NOT the same thing, since they are different nodes—about as different as they can be, in completely separate networks.

2. Logical contradictions are allowed (as required by intensionality); nothing rules out the co-existence of two proposition nodes $p!$ and $(\neg p)!$. A cognitive agent C can have concepts regarding belief, assertion, and so forth, naturally represented by nodes. For nodes n and m in the cognitive agent C, suppose $[\![n]\!]$ is that C likes Violet, $[\![m]\!]$ is that C believes that C likes Violet, and that m is asserted (i.e., $m!$) and n is not. (See footnote 7 in [Shapiro and Rapaport, 1991].) What kind of contradiction is this?

3. Suppose $[\![n]\!]$ is C's concept of Hugo's dog and that this interchange takes place between C and D:

   C: She always hated Hugo's dog.
   D: But Hugo doesn't have a dog.

[1] The inverse function exists because the correspondence between nodes and concepts is one-to-one, as mandated by the Uniqueness Principle for SNePS, treated in § 2.5.

C: Well, maybe it was Manuel's dog.
D: Oh, you mean Manuel's *cat*.

What happens to $[\![n]\!]$?

The semantics of a SNePS node needs to be clearly expressed so that questions of identity of concepts and results of computations can be resolved. These issues in reconciling human capacity with the SNePS representation provide the impetus to examine the computations necessary to the determination of $[\![n]\!]$.

1.5 Some Broad Motivational Questions

Brian Cantwell Smith has proposed twelve foundational questions to be asked of a reasoning or inference system, as a means of placing it in the theoretical space of AI [Smith, 1991]. His twelve questions provide something of a new manifesto for artificial intelligence, and are worth applying to any knowledge/belief representation system; here we apply it to SNePS. He claims that traditional methods for furthering the goals of AI through knowledge representation clump together at a point somewhat distant from actual knowledge, meaning human knowledge. The implication is that simulated cognition cannot be achieved in a system lacking the vagaries of human cognition.

Smith answers his twelve questions in terms of (1) the traditional approach of predicate logic, (2) the CYC system proposed by Dougles B. Lenat and Edward Feigenbaum [Lenat and Feigenbaum, 1991], and (3) his admittedly ill-defined "minimum standard for an AI system," called *embedded computation*, or EC [Smith, 1991, page 259]. The questions appear below, with Smith's answers provided in square brackets, followed by preliminary answers for SNePS, to be reviewed in the conclusion.

1. **Does the system focus primarily on explicit representation?** [Logic—Yes; CYC—Yes; EC—No.] Possibly no, because Smith considers compositionality a consequent of "explicit," and some aspects of SNePS semantics are apparently not compositional. Such is the topic of Chapter 6.

2. **Is representational content contextual (situated)?** [Logic—No; CYC—No; EC—Yes.] Yes; SNePS's cognitive agents rely on indexicals, as opposed to systems that work without a point of view [Rapaport, 1986].

3. **Does meaning depend on use?** [Logic—No; CYC—No; EC—Yes.] Yes, insofar as Smith's dynamic notion of meaning as "the whole complex of inferential, conversational, social, and other purposes to which it is put" is the same as the "holistic" view of the meaning of a SNePS node (see §2.7.2).

4. **Is consistency mandated?** [Logic—Yes; CYC—No; EC—No.] No; a cognitive agent may have logically contradictory intensions.

5. **Does the system use a single representational scheme?** [Logic—Yes; CYC—Yes; EC—No.] Maybe yes, insofar as all concepts are in nodes; maybe no, insofar as SNePS users employ different case frames.

6. **Are there only discrete propositions (no continuous representation, images, ...)?** [Logic—Yes; CYC—Yes; EC—No.] Hard to say. Smith wants to have "lots of clouds", without counting the clouds. If that's all it is, then no; a system (such as SNePS) that embraces a "round square" won't shrink from "lots of clouds" [Cho, 1992].

7. **Do the representations capture all that matters?** [Logic—Yes; CYC—Yes; EC—No.] Hard to say. Smith apparently wants to capture feedback, instinctual physical reactions, etc.

8. **Are reasoning and inference central?** [Logic—Yes, mostly; CYC—Yes; EC—Yes.] Yes.

9. **Are participation and action crucial?** [Logic—No; CYC—No; EC—Yes.] No, insofar as this is not required for a use or implementation of SNePS to be judged valid or successful. A system under current development is aimed at making active capacity available [Kumar *et al.*, 1988]; that still falls short, however, of making it "crucial".

10. **Is physical embodiment important?** [Logic—No; CYC—No; EC—Yes.] Yes, insofar as sensory nodes are crucial to grounding the meanings of nodes; or — No, insofar as these sensory nodes are under the control of the user, and not forced to be dependent on the embodying device.

11. **Does the system support "original" semantics?** [Logic—No; CYC—No; EC—Yes.] Yes, because Smith means that the system's constructs should mean something regardless of interpretation by outsiders, and SNePS nodes are concepts, structures internal to the agent, and in need of no observer to confer first-class status.

12. **Room for a divergence between the representational capacities of theorist and agent?** [Logic—No; CYC—No; EC—Yes.] Presumably yes, because a SNePS cognitive agent can have different concepts than does its creator.

Wherever they stand on the virtues of this analysis, AI researchers would do well to regard another message from Smith—that providing the two extremes of "broad intuition" and "detailed proposal" is methodologically bankrupt; it is high time to focus on "the middle ground of conceptual analysis and carefully laid-out details" [page 253]. It seems clear that this middle ground, once assumed to be a matter of concentrated architecture and engineering, has proved elusive. Even Smith himself does not reach it; to define an alternative like embedded computation solely in terms of its lack of others' faults is not to offer the "intermediating conceptual structure" for which he calls [page 252].

The gap between toy AI programs, productive but so small, and the going theories of intelligence, too general to implement, is discouraging. SNePS is a disciplined attempt to build an down-to-earth cognitive model, an application of artificial intelligence principles, and this study an attempt to narrow that gap.

Chapter 2

Background

2.1 Introduction

An understanding of some formal theories and the fundamentals of SNePS is necessary to understanding the non-well-founded set semantics to be proposed. A survey of the relevant areas is provided in this chapter, along with a discussion of the current issues in SNePS.

2.2 The Meaning of 'Semantics'

A formal **semantics** is a relation between two domains. To define a semantic relation, three specifications must be given:

- A class $\mathcal{Y}$, called the *domain*, conventionally regarded as the set of syntactic structures.
- A class $\mathcal{Z}$, called the *range*, conventionally regarded as the set of semantic structures.
- A relation $R \subseteq \mathcal{Y} \times \mathcal{Z}$, where the Cartesian product $\mathcal{Y} \times \mathcal{Z}$ is the set of ordered pairs $\{ \langle y, z \rangle \,|\, y \in \mathcal{Y} \text{ and } z \in \mathcal{Z} \}$

Most of the time, semantics is considered to be a function, so instead of R, we would include:

- A function $\mu : \mathcal{Y} \longrightarrow \mathcal{Z}$

A "syntactic" domain $\mathcal{Y}$ may, of course, give rise to several different semantics of interest, specified by different sets z playing the role of $\mathcal{Z}$.

In the absence of further specification, let us take the syntactic and semantic domains $\mathcal{Y}$ and $\mathcal{Z}$ to be *classes*, although they commonly turn out also to be *sets* (see Chapter 3). Appropriate questions about these domains are: Are $\mathcal{Y}$ and $\mathcal{Z}$ finite? If not, are they countably or uncountably infinite?

Unqualified references to 'semantics' herein, using μ, refer to the function version. In its broadest construal, the semantic relation R or function μ is simply a set of ordered pairs. If it is a function, however, we can further ask: Is μ a *total* function? Is it injective, surjective, bijective? Is it compositional (in a sense to be defined)? Is it a *computable* function?

2.3 Functions

A function $f : D \longrightarrow R$ is a set of ordered pairs, a subset of the Cartesian product $D \times R$, where D is called the *domain* and R the *range*.[1] A function has the special restriction on the ordered pairs that if $\langle x, y\rangle \in f$, then $[\forall z \neq y]\ \langle x, z\rangle \notin f$. A function is *injective* if it is one-to-one, that is, if $\langle x, y\rangle \in f$, then $[\forall w \neq x]\ \langle w, y\rangle \notin f$.

The critical point is that functions are sets. To say that $f(x) = y$ is to say that $\langle x, y\rangle \in f$. The *successor* function, for example, consists of all ordered pairs $\langle x, x+1\rangle$, hence is a countably infinite set, having the cardinality of the natural numbers $\mathcal{N}$, namely, $\aleph_0$. There are finite and infinite sets of ordered pairs over the natural numbers $\mathcal{N}$, so there are $2^{\aleph_0}$ functions, the cardinality of the set of subsets of $\mathcal{N} \times \mathcal{N}$. Other set-theoretical concepts are also useful, such as subsets of functions, and the use of the binary set operations—union, intersection, and so on—to define new functions.

A given function $f : D \longrightarrow R$ is either *defined* or *undefined* on every possible argument $x \in D$, depending on whether f includes an ordered pair $\langle x, y\rangle$ for some $y \in R$. If f is defined on x, we write $f(x)\downarrow$, or, to indicate the particular value y of the function on that argument, $f(x)\downarrow y$. If f is undefined on x, we write $f(x)\uparrow$. So if $\forall x f(x)\uparrow$, then $f = \emptyset$, or, in lambda-notation, $f = \lambda x.\uparrow$, the totally undefined function.

A function is *computable* (or *recursive*) if there is a Turing Machine that computes it. A function that is not defined (does not return a value) on some arguments corresponds to a Turing Machine that does not halt on those arguments. Even $\lambda x.\uparrow$ is computable, as witnessed by any Turing Machine that does not give a final value no matter what argument is submitted. Since Turing Machines can be enumerated, infinitely, there are $\aleph_0$ of them, and there are, therefore, only $\aleph_0$ computable functions. This implies that the vast majority of functions—if such terms of magnitude are meaningful in this realm—are not computable. Note also that every computable function has $\aleph_0$ distinct Turing Machines that compute it, which can be regarded as its intensional definitions.

2.4 Graph Theory

The abstract mathematical structure on which semantic nets in general rely is the *graph*, a set of discrete objects called *nodes* with connections called *arcs* between them, or, more specifically, the *directed graph*, a set of nodes with unidirectional arcs between them [Harary, 1972]. A directed graph G is formally defined as $V \cup E$, where V is a finite non-empty set of nodes and $E \subseteq V \times V$ is a set of ordered pairs of distinct nodes $\langle u, v\rangle$, which are the arcs of G. (E is usually a relation that is not a function.) The node u is called the *parent* (with respect to v) and the node v the *child* (with respect to u).

Two nodes u, v are *adjacent* if $\langle u, v\rangle \in E$ or $\langle v, u\rangle \in E$. Sometimes the predicate "$\langle u, v\rangle \in E$" will be written simply $u \rightarrow v$. A *path* is an alternating sequence of distinct nodes and distinct arcs, $u_1, \langle u_1, u_2\rangle, u_2, \langle u_2, u_3\rangle, \ldots, \langle u_n, u_{n+1}\rangle, u_{n+1}$, such that, for all $1 \leq i \leq n$, $\langle u_i, u_{i+1}\rangle \in E$. (We sometimes say that u_{n+1} can be *reached* from u_1.) The *length* of the path is the numbers of arcs n. A path may be given by the sequence of nodes alone, since that determines the necessary arcs. A *cycle* is a sequence of nodes $u_1, u_2, \ldots, u_n, u_{n+1}$ (where, for all $1 \leq i \leq n$, $\langle u_i, u_{i+1}\rangle \in E$) such that $u_{n+1} = u_1$ but all other nodes are distinct, so a cycle is like a path except that it ends at the node of origin. An *acyclic* graph is one that contains no cycles. We will also need a traversal of arcs that does not respect their direction; a *semipath* is a sequence of distinct adjacent nodes. Sometimes a node with no children—i.e., a node w such that $\forall v \in V, \langle w, v\rangle \notin E$—is called a *leaf*, but that term is usually reserved for *trees*, connected directed graphs with a distinguished node called the *root* and a unique path to every node. In a directed graph, the number of arcs extending out from a given

[1] This is the *extensional* view of functions, as opposed to the *intensional* view, under which functions are identified with the algorithms that compute them.

node is its *outdegree*, and the number coming in to the node is its *indegree*. A leaf, in other words, or any other node with no children, has outdegree zero, while the root of a tree has indegree zero.

A SNePS network is often described as "a directed acyclic graph" [Shapiro, 1991, page 137]. (See §2.5 for a presentation of SNePS terminology.) Since SNePS allows more than one arc $\langle n_1, n_2\rangle$ between the same two nodes n_1 and n_2 (as long as the arcs are labeled differently), the SNePS network is therefore, more precisely, a *multigraph* rather than a simple graph [Harary, 1972; Gibbons, 1985]. From these multigraphs, however, significant simple graphs are easily derived, by, for example, conflation of arcs (replacing multiple arcs from node u to node v with a single arc $\langle u, v\rangle$), or restriction to arcs with certain labels, yielding simple graphs to which the full machinery of graph theory may be applied. We formalize the first option in the following definition.

Definition 2.4.1 *Given a SNePS network S, its* ***unigraph*** *S' is the graph consisting of V, the set of nodes in S, and $E' \subseteq V \times V$, where $\langle x, y\rangle \in E'$ if and only if there is at least one arc in S from node x to node y.*

The unigraph S', then, depicts the raw directed connectivity or adjacency status of S.

A subgraph of G is a graph $G^- = V^- \cup E^-$ such that $V^- \subseteq V$, $E^- \subseteq E$, and $\forall\langle u, v\rangle \in E^-$, $u \in V$ & $v \in V$. A subset V^- of the nodes alone suffices to define a subgraph G^-, which then includes all arcs $a \in V^- \times V^-$ such that $a \in E$; we call this the *subgraph of G induced by V^-*. We will also have occasion to use subgraphs induced by a set of arcs; the *subgraph of G induced by E^-* is $G^- = V^- \cup E^-$, where a node v is in V^- if and only if there is some arc adjacent to it in the given E^- [Harary, 1972].

A SNePS network is not necessarily *(weakly) connected*, which would require that there be a semipath between every pair of points; though none of the examples used here show this, it is possible that a SNePS network could include several separate subnetworks, that is, disconnected components. The way in which each component is connected is the weakest, since all that we can claim is that there is a semipath between any two nodes (in a component). A component would be *strongly connected* if every two nodes are reachable on some path from each other, and *unilaterally connected* if for any two points, at least one is reachable from the other. Simple examination of Figure 2.1 ahead will show that (components of) SNePS networks are neither strongly nor unilaterally connected, since, for instance, there is no path in either direction between `m3` and `m4` (but there is a semipath).

Other graph-theoretic concepts also translate easily between the language of SNePS and the language of graph theory; for example, a base node has an *outdegree* of zero, and it is the rules for building networks (see [Shapiro and Rapaport, 1987]) that make them *acyclic*. An easy consequence of these properties is that there must exist, in every network (at least, in S' form), nodes with *indegree* zero [Harary, 1972, Theorem 16.2′]. In fact, there is a unique minimal set of such points that, between them, allow all other nodes to be reached along some path [Harary, 1972, Theorem 16.6], called the *point basis*. This may be a smaller set than the set of nodes with indegree zero.

2.5 The SNePS Environment

A SNePS network is a propositional semantic network, that is, one in which every proposition represented in the network is represented by a node (rather than an arc). Arcs are best regarded as punctuation, having no conceptual semantics. For this reason, it is forbidden to add an arc between two existing nodes. Certain arc labels come with SNePS; others necessary for a particular implementation are to be defined by the user.

Arcs are directed; the node at the origin is called the 'tail' node and the node at the arrowhead, the 'head' node. There would be no point in connecting two nodes with multiple instances of the same arc (arcs with the same label), but there may well be multiple arcs with the same label emanating from the same tail node but terminating at different head nodes, or multiple arcs with different labels connecting two nodes. Nodes with no arcs emanating from them (in the graph-theoretical terms above, with outdegree zero) are called *atomic*. They include (1) sensory nodes, which represent the real-world interface; (2) base nodes, which represent individual concepts; and (3) variable nodes, which represent arbitrary concepts (individuals or propositions). In this work, the domain of discourse is SNePS$_P$ only, which is SNePS without variable nodes. Nodes that do have arcs emanating from them, i.e., that dominate others, are called *molecular*. They include (1) structured individual nodes and (2) structured proposition nodes. The formal definitions that will be needed follow [Shapiro, 1991, page 145]:

> ***Definition 1*** A **wire** is an ordered pair $\langle r, n \rangle$, where r is a SNePS relation, and n is a SNePS node.
> ***Definition 2*** A **cable** is an ordered pair $\langle r, ns \rangle$, where r is a SNePS relation, and ns is a nonempty set of SNePS nodes.
> ***Definition 3*** A **cableset** is a nonempty set of cables, $\{\langle r_1, ns_1 \rangle, \ldots, \langle r_k, ns_k \rangle\}$, such that $r_i = r_j \iff i = j$.
> ***Definition 4*** Every cableset is a SNePS **node**. Every SNePS$_P$ node is either a base node or a cableset.
> ***Definition 5*** A **molecular node** is a cableset.
> ***Definition 6*** We will overload the **membership** relation "$\in$" so the $x \in s$ holds just under the following conditions:
>
> - If x is any object and s is a set of such objects, then $\in$ has its usual meaning. (Note that this situation obtains if x is a cable and s is a cableset.)
> - If x is a wire $\langle r_1, n \rangle$ and s is a cable $\langle r_2, ns \rangle$, then $x \in s \iff r_1 = r_2 \wedge n \in ns$.
> - If x is a wire and s is a cableset, then $x \in s \iff \exists(c)[c \in s \wedge x \in c]$.
> - If x is a wire or a cable and s is a base node, then $x \notin s$.
>
> ***Definition 7*** An **nrn-path** from the node n_1 to the node n_{k+1} is a sequence $n_1, r_1, \ldots, n_k, r_k, n_{k+1}$ where the n_i are nodes, the r_i are SNePS relations, and for each i, $\langle r_i, n_{i+1} \rangle$ is a wire in n_i. We say the the nrn-path $n_1, r_1, \ldots, n_k, r_k, n_{k+1}$ goes through $n_i, 1 \leq i \leq k$.
> ***Definition 8*** A node n_1 **dominates** a node n_2 just in case there is an nrn-path from n_1 to n_2.

When a SNePS network is used to model the mind of a cognitive agent, a node representing a proposition may be *asserted*, a special status that conveys the agent's belief in that proposition. A single cognitive agent may encompass several *belief spaces*, each of which contains its own assumptions and is that agent's view of another cognitive agent; assertions are relative to the belief space. These mechanisms allow for sophisticated "conscious" and "subconscious" reasoning, so that, for example, cognitive agent C, who does *not* believe that marzipan is delicious, has a crack at concocting a confection pleasing to cognitive agent D, who *does* believe that marzipan is delicious—either by, crudely put, actively figuring out D's taste, or by realizing something about D's taste. (The definitions sketched above can be found mainly in [Shapiro and Rapaport, 1987], [Shapiro, 1991], and [Shapiro and Rapaport, 1991].)

An example of a SNePS network is shown in Figure 2.1, the representation in the cognitive agent CASSIE of the sentence "John believes that the girl next door is sweet," from [Shapiro and Rapaport, 1991]. The subnetwork representing the property of "being the girl next door" is not shown, but will be taken to be rooted at a molecular node `m9`. Base nodes include `b1` and `b2`, which represent, respectively, John (that is, the object whose proper name is 'John') and the girl next door (that is, the object that, to John, has the property of being the girl next door). An example of a molecular node is `m6`, which represents the proposition that the girl next door is sweet. An example of an asserted molecular node is `m8`, which represents the proposition that $[\![$`b1`$]\!]$ (John) believes $[\![$`m6`$]\!]$ (that the girl next door is sweet). It is asserted because $[\![$`b1`$]\!]$ is John, and CASSIE holds this belief

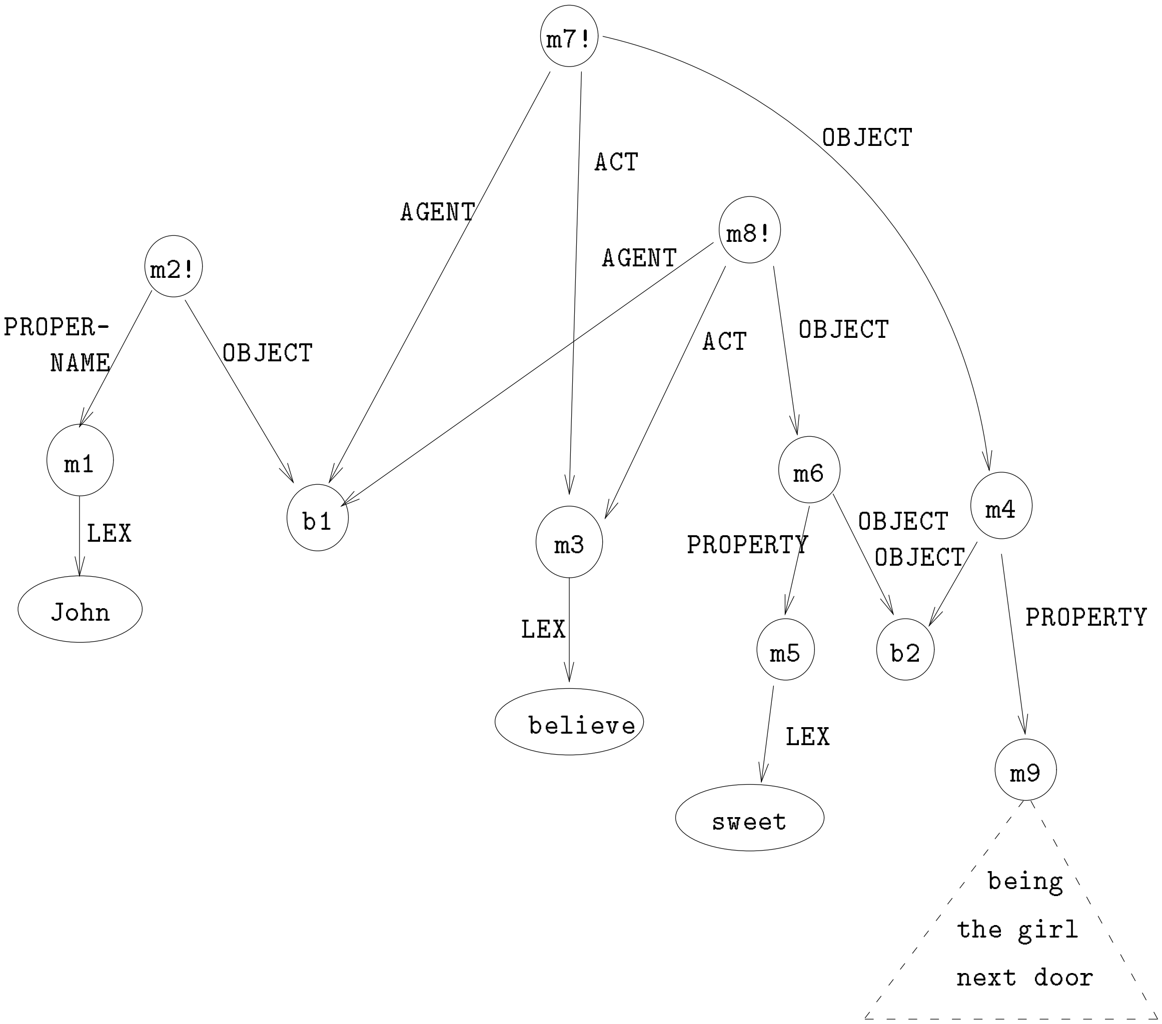


Figure 2.1: Example SNePS network representing "John believes that the girl next door is sweet."

about John's attitude toward $[\![\texttt{b2}]\!]$, the girl next door. The absence of assertion on the node **m6** shows that CASSIE herself has no such commitment to the sweetness of $[\![\texttt{b2}]\!]$.

2.6 The Semantics of SNePS So Far

The official version of SNePS is the latest release, SNePS-2.1, but by "SNePS" is meant the whole system and all of its parts, implemented and theoretical, comprising any form of study by the SNePS Research Group—programs, papers, conferences, meetings, and even usage conventions that have reached the status of implicit rule. We include especially, as a formal touchstone, the data structures and procedures that comprise the SNePS abstract data type constructed by Morgado [Morgado, 1986], as well as—as implied by the inclusion of "conventions"—SNePS as a dynamic social and intellectual phenomenon. That is, no feature will be discounted because it is held *only* by unwritten consensus, but such situations will be noted. Indeed, a major thrust of the study is to clarify and legitimize those features. This analysis is nominally directed at full SNePS, although we are primarily interested in its role as a cognitive agent, such as CASSIE in the research project SNePS/CASSIE [Shapiro and Rapaport, 1987]. The distinction will be drawn when necessary.

The mechanics of the current implementation of SNePS, SNePS-2.1, are described in [Shapiro and Group, 1989]. The major constructor is the BUILD command, which adds nodes, and thereby information, to networks. It also reinforces the Uniqueness Principle (see §2.7) by only constructing nodes that do not already exist.

2.6.1 Standard model-theoretic semantics

If SNePS is a formal propositional system, as stated in [Shapiro, 1991], then a truth-functional semantics is a standard choice, assigning to a proposition node a member of the domain {true, false}, to an individual node an extensional constant or atom, and to a rule node a function from closed sentences to {true, false} [Rapaport, 1992a; Rapaport, 1992b]. But the shortcomings of such a system, alluded to in §1.3.1, are obvious:

- Intensional individuals are not necessarily extensional (existing in the real world), as witnessed by Athena, above, by the paradoxical Russell Set, and by "the health benefits of smoking" (mentioned in advertisements of years past).
- The set {true, false} is inadequate as the range of interpretation of propositions, unless we accept that "The population of the United States in 1990 was 5,062" has exactly the same meaning as "Apollo was the mother of Pegasus". In other words (Frege's, in fact), intensionality requires the *sense* of a proposition as well as its *referent*.
- There is no clear semantics provided for *act* nodes.

Since the standard approach to propositional semantics does not respect intensionality, it is not suitable for SNePS.

2.6.2 Formal definitions

Some work has already been done to interpret the set of valid SNePS constructs, but all take slightly different points of view. According to [Shapiro and Rapaport, 1991, page 221],

> The entity represented by a node is determined in one of two ways: it is determined *assertionally* by the subnetwork connected to the node via arcs pointing into it; it is determined *structurally* by the subnetwork connected to the node via arcs pointing out of it. ...Once a node is created, it can get new arcs pointing into it, but the set of arcs

pointing out of it can never change, so the structurally determined nature of a node is more characteristic of it than its assertionally determined nature.

Shapiro also gives a dynamic belief semantics, through *path-based inference* [Shapiro, 1991]. The semantics of acts remains to be specified, and is the subject of current research [Kumar and Shapiro, 1991].

When it is necessary to refer to classes of structures formally, the following terms will be used:

Definition 2.6.1 ***SNodes:*** *the set of all well-formed SNePS/CASSIE proposition nodes, as specified by the generation rules SR.i [Shapiro and Rapaport, 1987], and the assumption that base nodes are given.*

Definition 2.6.2 ***SNets:*** *the set of all well-formed SNePS networks, as specified by the generation rules SR.i [Shapiro and Rapaport, 1987], and the assumption that base nodes are given.*

2.6.3 Intensionality is incorporated through *Aussersein*

Let $\mathcal{D}$ name the domain of entities represented by SNePS nodes. As discussed in §1.3.2, $\mathcal{D}$ can also be regarded as the Meinongian domain *Aussersein*—at least, for individual, proposition, and rule nodes. We would extend the definition of intensionality as first-class treatment of all objects by noting that, intensionally speaking, not only are all concepts first-class objects, but all first-class objects are (potential) concepts. The *necessary* condition for an entity's membership in $\mathcal{D}$ is that it be a first-class mental object; the *sufficient* condition is the same. That may be the closest we can get to circumscribing $\mathcal{D}$ in a more rigorous way than describing it in *Aussersein* terms as "everything we can talk and think about" [Rapaport, 1978].

2.7 Fundamental Principles of SNePS

2.7.1 Avowed principles

As described in the previous chapter, certain principles of SNePS stated in the literature are meant to distinguish SNePS from other cognitive models [Rapaport, 1985a; Shapiro and Rapaport, 1987].

- Intensionality: SNePS is (or allows) a fully intensional representation of belief; the nodes represent concepts ("objects of thought"), not limited to extensional or possible objects.
- The Uniqueness Principle: The relation between the cognitive agent's concepts and the nodes in the network is a bijection, a one-to-one correspondence.
- Permanence of Structural Status: No out-arcs may be added to a node. The only way to add new information about a concept is with nodes that dominate it. In other words, the *structural* status of a node cannot be changed; new information can only be added *assertionally*, a dichotomy used by Woods [Woods, 1975] and formalized for SNePS, as discussed in §2.5.

2.7.2 Articles of Faith

There are also unwritten principles held, or at least proposed, by those who have devoted significant time and discussion to the study and development of SNePS—and which are not fully addressed in the published literature:

Full embedding

The meaning of a node depends on its full embedding, that is, its location in the entire surrounding network. Stated conversely, the meaning of a node cannot be determined by an examination of some proper subset of its neighboring nodes and arcs. As put in [Ehrlich and Rapaport, 1992, page 2], describing work in progress:

> We take the meaning of a word (for a cognitive agent) to be the location of that word in a mental semantic network of words, propositions, and other concepts .... We thus adopt Quine's view that a cognitive agent's knowledge and beliefs form an interconnected web, a change or addition to which can affect other portions linked to it.

Distinctions of concepts

The only necessarily permanent attribute of a node is that it always exists distinct from all others.

The motivation is to reflect a human cognitive phenomenon, illustrated, for example, by the interchange given between C and D concerning Hugo's dog, which turned out to be about Hugo's cat. (See §1.2 above.) Some sort of context for the reference remains fixed, even though there is no way to tell *a priori* what attributes will persist. (No phenomenon corresponding to memory loss is currently provided.) So if everything about a node can change, what does it mean?

Circularity

The semantics of SNePS is circular, notwithstanding its acyclic graph representation. The meanings of certain directly-connected nodes influence each other.

There is support for the general idea of circular semantics in the AI community. In *The Situation in Logic*, Jon Barwise ([Barwise, 1989, pages 194–198] lists "some inherently circular situations" to justify his claim that "reality is not wellfounded". Here are simplified versions of two of them:

Example 7: A man forgets to bring a dish to a potluck dinner in honor of a friend, and admits, "This is a very embarassing situation." The remark itself, of course, is part of the situation mentioned.

Example 12: Players of five-card stud, a poker game in which each player gets one "down" card (known only to him/herself) and four "up" cards (visible to all), have an understanding of the situation that integrally involves each player's conception of the situation. There is no neat separation into independent levels of knowledge.

Bernhard Nebel studies the properties of *terminological cycles* in knowledge representation, constructions that "arise when a concept is defined by referring, directly or indirectly, to itself," such as dictionary entries [Nebel, 1991]. Most of the well-known knowledge/belief representation systems either ignore them or explicitly exclude them, but they are useful and common. Nebel says that these reasons are enough to analyze them, and the difficulty of doing so is not reason enough not to.

When Smith compares traditional logic and the CYC system to his own view of what true artificial intelligence would be, as discussed in Chapter 1, among his criticisms of the first two systems are their inadequate respect for usage as a determiner of meaning: "the meaning of 'water' is as much determined by the meaning of the discourse as the meaning of the discourse is determined by the meaning of 'water'." He then acknowledges that this view "challenges the traditional view that semantics can be 'compositionally' defined on top of a base set of atomic values" [Smith, 1991, pp. 265 ff.].

2.8 The Need for Further Analysis

2.8.1 Semantics of base nodes

While [Shapiro, 1991] addresses the semantics of molecular nodes, especially as regards their assertional (and therefore inferential) status, it does not provide a meaning for base nodes independent of their syntactic placement.

2.8.2 Semantics of molecular nodes

Somehow we want the meaning of a node to be its location in a network, encompassing both the structure that dominates the node and that is dominated by it. This notion is simply a wholesale adoption of the general intuition; it is meant to be less formal than the various semantic interpretations of a node that may be derived later. There are two facets to this characterization: (a) the entire surrounding network contributes to the meaning, in accordance with the emerging consensus on this issue mentioned above, and (b) the relative placement of a node among the others matters; otherwise, all nodes in a network would have the same meaning, namely, the entire network in aggregate, with no specific perspective. This suggests that the meaning of a node includes the entire history of the acquisition of the concept, that, in fact, connection is always made to concepts in the context of learning.

2.8.3 Bridling semantic computation

While the meaning of a node may be taken *theoretically* as the full network surrounding it, pragmatic considerations require that we explicitly acknowledge that processing of such a network to derive such a meaning must be limited. A SCOPE feature would provide a measurement of processing extent—i.e., to what depth was the search tree expanded?—and a measurement of difference in meaning between two nodes. The *difference* between $[\![n]\!]$ and $[\![m]\!]$ would be the *distance* (i.e., the number of arcs) that must be traversed outward (in any direction) from n and m until the subnetworks thus induced differ, i.e., until there is an arc or node in one subnetwork that is not in the other. If that distance is x, the nodes are *x-equivalent*. If the entire respective networks are covered, then we will say that the nodes are $*$-equivalent, and n and m have the same meaning: $[\![n]\!] = [\![m]\!]$. The distance employed in a given processing situation will be called the SCOPE, and formally defined, for a node n in a given SNePS network $S = V \cup L$ and a given distance d, as follows.

Definition 2.8.1

$$\begin{array}{rcl}\textbf{\textit{SCOPE}}(n,d) & = & \textit{the subgraph induced by } V_0 \cup V_1 \cup \ldots \cup V_d, \\ & & \textit{where } V_i \textit{ is the set of nodes on some semipath of length } i \textit{ from } n.\end{array}$$

Two examples of subnetworks defined from node `m6` of Figure 2.1 by different values of SCOPE are given in Figures 2.2 and 2.3.

For two nodes in the same cognitive agent, $*$-equivalence is ruled out by the Uniqueness Principle. What about two nodes in different cognitive agents? If the meaning is only the location in the network, then two corresponding nodes in C and D might end up at the same place (albeit at different times) and therefore be $*$-equivalent. But if the entire history of acquisition is included in the meaning, then $*$-equivalence is also effectively ruled out for two nodes in different cognitive agents, since they would have had to be acquired in absolutely identical contexts, under absolutely identical histories—virtually impossible for two distinct agents.

The incorporation of a scope feature into inference mechanisms of SNePS would alleviate the problem of "logical omniscience"—the problem that extant cognitive agents do not know or believe all the logical consequences of their beliefs—by limiting the range of inference that takes place at

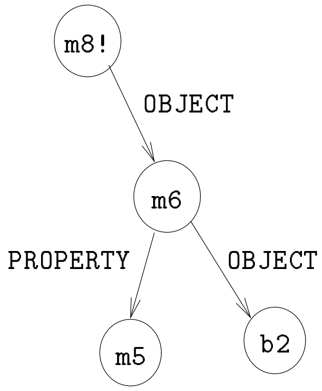


Figure 2.2: SCOPE(`m6`,1) from network of Figure 2.1

any given time. It would be complementary to the "subconscious" reasoning provided by path-based inference [Shapiro, 1991].

SCOPE may be seen as the institutionalization of the distinction between competence and performance. In natural language processing research, competence yields theoretical results such as grammars for languages of infinite cardinality, and performance explains why human speakers do not always act in accordance with those results. When the SCOPE is *, the results obtained are competence results; with any SCOPE short of that, they are performance results.

2.9 Summary

To provide a semantics for some group of structures is normally to provide meanings for them, but since 'meaning' is not a rigorously defined term, a semantics formally is nothing more than a mapping from one domain to another, the latter taken to be the set of meanings. Unless otherwise specified, the mapping is a function, so that a given structure only has one meaning.

Since the task in succeeding chapters is to provide a semantics for SNePS, its properties are of interest. It is a graphical model, and graph theory has been reviewed for relevant concepts and terms. (Axomatic set theory and the non-well-founded variant of it that is to be used extensively get a chapter of their own, the next one.) SNePS also follows characteristic principles not entailed by graph theory, such as the classification of nodes into different categories with different structural constraints and the one-to-one correspondence between concepts and nodes called the Uniqueness Principle; these have been enumerated.

As a knowledge/belief representation, SNePS incorporates intensionality by conceptualizing as a node any object of thought, supports the thesis that the full meaning of a node involves the entire network in which it is embedded, and embraces some degree of circularity in the semantic influence on each other of adjacent nodes. Given this background, and the SCOPE mechanism defined as a first attempt at limiting the context of semantic computation, we are ready to proceed toward a non-well-founded set-theoretic semantics.

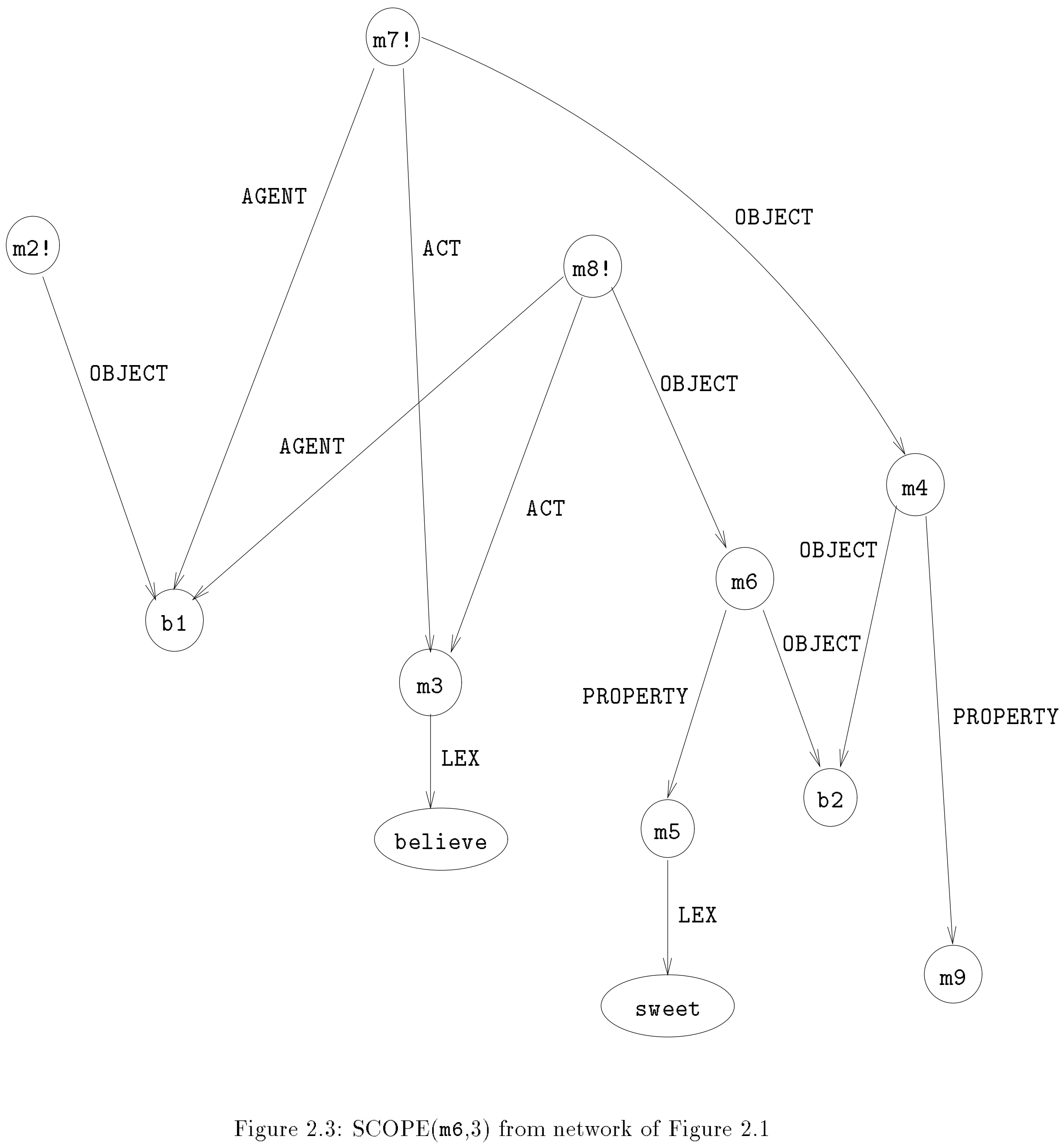


Figure 2.3: SCOPE(m6,3) from network of Figure 2.1

Chapter 3

Non-Well-Founded Set Theory

3.1 Introduction

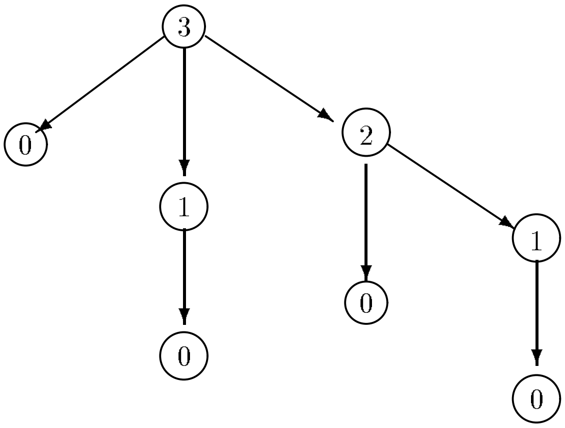
In this chapter, the non-well-founded set theory of Peter Aczel is sketched. Sets are construed as collections of objects that are sets themselves; in Aczel's theory a set can even contain itself. The graphical representation of a set is an important vehicle for the theory. Nodes are construed as sets, with arcs showing membership and cycles showing sets that somehow contain themselves, either directly or indirectly (through hereditary membership).

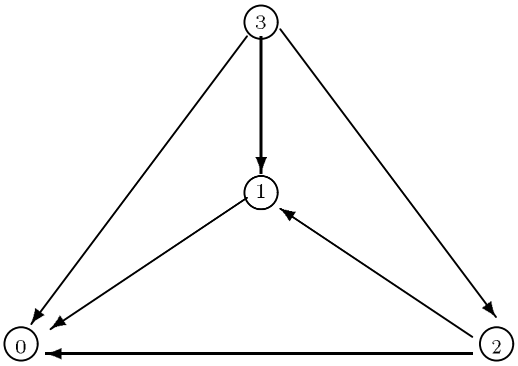
3.2 ZFC Set Theory

Set theory provides a rigorous environment in which to reason about objects, providing the foundation of much of mathematics and other formal studies. According to Cantor (in translation by Jourdain), where the more modern 'set' should be read for 'aggregate':

> By an "aggregate" we are to understand any collection into a whole M of definite and separate objects m of our intuition or our thought. [Cantor, 1952, page 85]

So as not to limit the scope of inquiry, set theorists commonly adopt a homogeneous typing under which all members of sets are regarded as sets themselves, and as *hereditary*, so that each set contains all members of its members. A member of a set S might also be a subset of S. When we need concrete objects that are not sets, we will call them 'atoms'. This simple mechanism permits the standard set-theoretic definition of the natural numbers $\mathcal{N}$ as $\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}, \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}, \ldots$. Each $n \in \mathcal{N}$ is represented by the set $\{m \in \mathcal{N} \mid m < n\}$, or, equivalently, $n + 1 = n \cup \{n\}$, where $\emptyset$ is identified with zero. For example, $\emptyset \subset \{\emptyset, \{\emptyset\}\}$ and $\emptyset \in \{\emptyset, \{\emptyset\}\}$; 0 is both a member and a subset of 2. The hereditary nature of these sets allows them to be meaningfully depicted as rooted directed graphs, with the arrows showing membership. Here are two pictures of the set 3, on the left with multiple occurrences of the member nodes, and on the right with unique occurrences:





We are used to forming sets by the satisfaction of predicates on variables. For example, the set of Buicks in the parking lot is given by $\{x \mid x \text{ is a Buick } \& \ x \text{ is in the parking lot}\}$. It is also convenient to recognize objects which may be "too big" to be sets, called *classes*. For ϕ a predicate on x, an object of the form $\{x \mid \phi(x)\}$, if the set cannot be formed (if the object *is* "too big"), is a *proper class*—for example, the paradoxical Russell Set $\{x \mid x \notin x\}$. As Kenneth Kunen puts it, they do not formally exist and are best thought of as abbreviations for other expressions [Kunen, 1980].

Various axiomatic systems of set theory have been developed; one standard is that called *Zermelo-Frankel Set Theory with the Axiom of Choice*, abbreviated *ZFC*. Its nine axioms follow, as given in [Aczel, 1988, page 117] (who has avoided the use of abbreviations; see also [Kunen, 1980] for a comprehensive discussion with helpful intermediate definitions, and [Partee *et al.*, 1990, page 218 ff.] for a slightly different exposition).

For all sets a and b:

Extensionality: A set is determined by its membership; a and b are equal if $z \in a$ implies $z \in b$ and $z \in b$ implies $z \in a$.

$$\forall z(z \in a \leftrightarrow z \in b) \rightarrow a = b$$

Pairing: Any two sets a and b form a set that is their pair z.

$$\exists z\,[a \in z \ \& \ b \in z]$$

Union: Sets can be combined into unions; there is a set z that contains all members y of all members x of a.

$$\exists z(\forall x \in a)(\forall y \in x)\,[y \in z]$$

Powerset: For any set a, there is a set z that contains all of a's subsets.

$$\exists z \forall x\,[(\forall u \in x)(u \in a) \rightarrow x \in z]$$

Infinity: There is an infinite set, a set z that contains the empty set and, for every $x \in z$, contains some y such that $x \in y$.

$$\exists z\,[(\exists x \in z)\forall y\,\neg(y \in x) \ \& \ (\forall x \in z)(\exists y \in z)(x \in y)]$$

Separation (or "Comprehension"): [1] Sets can be defined in terms of properties subject to the existence of a set from which the elements are drawn; given a predicate ϕ, there is a set z whose membership is drawn from a and depends on ϕ.

$$\exists z \forall x\,[x \in z \leftrightarrow x \in a \ \& \ \phi]$$

[1] These are axiom schemata, defining an axiom for each ϕ (any formula that does not contain a free occurrence of z).

Collection (or "Replacement"): [1] Elements x that correspond, under some property ϕ, to others y (antecedently given) can be formulated as a set z.

$$(\forall x \in a)\exists y\, \phi \rightarrow \exists z(\forall x \in a)(\exists y \in z)\, \phi$$

Choice: Every non-empty set of disjoint non-empty sets has some choice function, that is, a selection z of a unique element y from each member set x of a.

$$\begin{aligned}(\forall x \in a)\exists y(y \in x)\ \&\ (\forall x_1 \in a)(\forall x_2 \in a)\,[\exists y(y \in x_1\ \&\ y \in x_2) \rightarrow x_1 = x_2]\\ \rightarrow\ \exists z(\forall x \in a)(\exists y \in x)(\forall u \in x)\,[u \in z \leftrightarrow u = y]\end{aligned}$$

Foundation (or "Regularity"): Any non-empty set has a member that is disjoint from it; for every set $a \neq \emptyset$, there is a set $x \in a$ such that $x \cap a = \emptyset$, i.e., there is no $y \in x$ that is also $\in a$.

$$\exists x(x \in a) \rightarrow (\exists x \in a)(\forall y \in x)\,\neg(y \in a)$$

It is the last axiom, Foundation, that is of interest here. It disallows, for example, the sets below:

$$a = \{a\} \tag{3.1}$$
$$b = \{s, t\},\ \text{where } s = \{t\} \text{ and } t = \{s\} \tag{3.2}$$

For the set a defined in equation (3.1), the single member is the set itself, and a therefore contains no member disjoint from it. In equation (3.2), $b \cap s = \{t\}$, and $b \cap t = \{s\}$. And what about the set s itself, where $s = \{t\}$ and $t = \{s\}$? Although such an s does not seem to violate the Axiom of Foundation directly, it cannot be defined under the cumulative conception. If it could, and t could, then the set $\{s, t\}$ would exist, but that's just b, above; the non-well-foundedness of b suffices to show the non-well-foundedness of s, and also of t.

Kunen points out that the Axiom of Foundation has no function in ordinary mathematics. Nothing depends on it, in the sense that the mathematical sets of interest already have that property, so it is not necessary to establish it independently. The Axiom of Foundation is, however, necessary to the application of transfinite induction as a proof technique [Kunen, 1980].

3.3 The Anti-Foundation Axiom

In his development of non-well-founded sets, Peter Aczel replaces the Axiom of Foundation with a strong negation, which he calls the Anti-Foundation Axiom, abbreviated AFA [Aczel, 1988]. To state it, we must define intermediate terms.

Accessible Pointed Graph: An *accessible pointed graph*, or *apg*, is a directed graph with a distinguished node called the "point" from which every other node can be reached.

Decoration: A *decoration* of an accessible pointed graph or apg is an assignment of a set to each node of the graph in such a way that the elements of the set assigned to a parent node are the sets assigned to its children. A childless node is assigned the empty set.

Picture: A *picture* of a set is an accessible pointed graph with a decoration in which the set is assigned to the point (by convention, at the top of the picture).

An accessible pointed graph is well-founded if it contains no cycles. An accessible pointed graph that is not well-founded depicts a set that is not well-founded, because no such set can be constructed through the cumulative hierarchy. But consider the structures in Figure 3.1, and suppose that we want to regard them as valid descriptions of phenomena, circular phenomena.

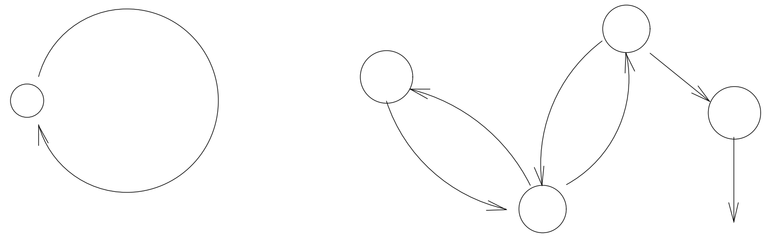


Figure 3.1: Accessible pointed graphs that are not well-founded

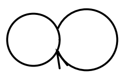
With appropriate decorations, these are pictures of *non-well-founded sets* or *hypersets*, legitimized by the replacement in ZFC of the Axiom of Foundation by the following:

The Anti-Foundation Axiom: Every graph has a unique decoration.

This statement is the conjunction of a claim of existence and a claim of uniqueness: every graph, even if it has cycles, has *at most one* and *at least one* hereditary assignment of sets to each node. (The weakest negation of the Axiom of Foundation would simply *allow* decorations for graphs with cycles.) Aczel calls the resulting axiom system $ZFC^{-} + AFA$, where ZFC^{-} is ZFC without the Axiom of Foundation.

By Mostowski's Collapsing Lemma, every *well-founded* graph has a unique decoration, with every childless node assigned the empty set and the same heredity property used to determine the sets that decorate parent nodes [Barwise and Etchemendy, 1987, page 41]. The introductory examples given using elements of the natural numbers $\mathcal{N}$ are illustrations.

Here are some examples of non-well-founded sets. The picture below represents the set x such that $x = \{x\}$, to which Aczel gives the special name Ω:



Note that writing down its contents in standard set notation is problematic since Ω also contains all members of its children. Because it is its own child, the solitary node would have to be decorated with something like this: $\{\{\{\ldots\}\}\}$. If only, as Aczel says, that expression "had an independently determined meaning!" [Aczel, 1988, page 7] Lacking that, its decoration is a set defined to be "a set equal to its own singleton" [Barwise and Etchemendy, 1987, page 37]—still awkward. Indeed, that is the advantage of the *picture* representation. It obviates the need for distracting and arbitrary names. Even the name Ω is unnecessary, of course, but useful, much like node labels in SNePS.

The set Ω is the completely circular hereditary set; it doesn't "bottom out" anywhere. In Figure 3.2 on the left, for example, the definition of *decoration* entails that the top node be assigned a set containing the bottom one, and the bottom node be assigned a set containing the top. As mentioned, no particular names for the sets are mandated, or even encouraged. But the top node *can* be assigned $\{\Omega\}$, as long as its sole child, beneath, is assigned its single member, i.e., Ω. But this is possible since $\Omega = \{\Omega\}$, and therefore that assignment is consistent with the requirement that the top node also be assigned Ω and $\{\Omega\}$ at the same time. The Anti-Foundation Axiom turns this possible decoration into a mandated decoration. Since the graph *can* be decorated that way, it *must be*. Both of the pictures in Figure 3.2 represent Ω. Any graph in which all nodes have at least one child is a picture of Ω [Aczel, 1988, page 7].

Following the development of Barwise and Etchemendy, we now bring in a set of atoms $\mathcal{A}$; a *decoration of a graph* then assigns to each childless node either an atom or the empty set. Further

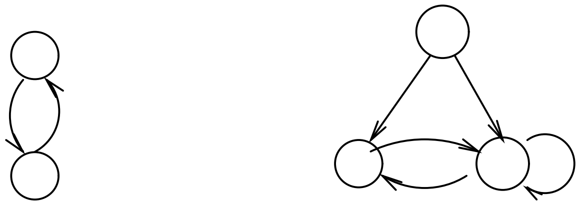


Figure 3.2: Other pictures of Ω

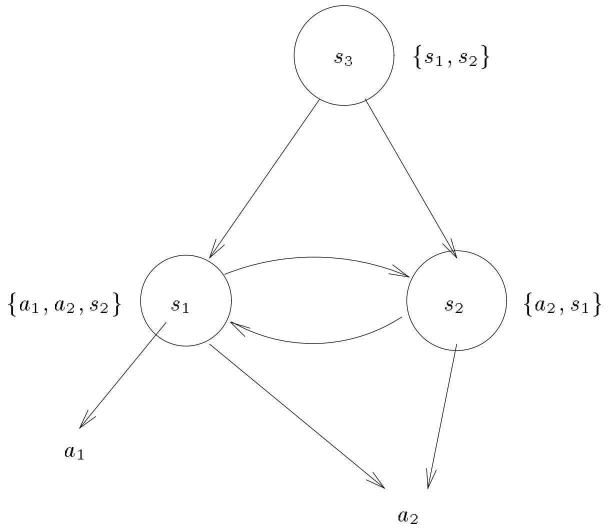


Figure 3.3: A picture of set s_3, showing abbreviated decoration

examples can now be constructed using elements from $\mathcal{A}$, which are not sets and hence have no picture representation with children, no out-arcs. They will be, for a graph G, the domain for a function that tags childless nodes with either the empty set or an atom,

$$tag \; : \; \{n \in G \mid outdegree(n) = 0\,\} \; \rightarrow \; \mathcal{A} \cup \{\emptyset\}$$

For a graph with no childless nodes, *tag* is the everywhere-undefined function $\lambda x.\uparrow$. The definition of *decoration* is extended so that assignments to other nodes are sets composed from the atoms of their descendants. For example, take $\mathcal{A} = \{a_1, a_2\}$ and sets $s_1 = \{a_1, a_2, s_2\}$, $s_2 = \{a_2, s_1\}$, $s_3 = \{s_1, s_2\}$. Figure 3.3 shows a tagged graph that serves as a picture of the non-well-founded set, or *hyperset*, s_3, with abbreviated decorations. Since it contains a cycle, full decoration would require infinite labels. The set s_3 is not well-founded because all of its pictures will contain cycles, due to the mutual membership of s_1 and s_2.

Since a hyperset may have several pictures, and they may be cyclic, the general problem of hyperset equivalence, when those hypersets are known only by apgs, is a difficult one. Fortunately, the hypersets to be used in conjunction with SNePS networks will be amenable to an extensional definition of equality, complicated but computable.

Aczel shows that the theory $ZFC^- + AFA$ is consistent, assuming that ZFC itself is, by embedding the universe of well-founded sets into a universe satisfying the new theory, thus producing a model of it. The status of classes, and of Russell's Paradox, is not affected. The Axiom of Comprehension rules out the troublesome $z = \{x \mid x \notin x\}$ as a set for both axiom systems, since existence, under that axiom, is contingent on membership of the elements in some given set a, unavailable here.[2] Of course, in traditional ZFC, we can still refer to a *proper class* of sets that do not contain themselves; this is what z is. In $ZFC^- + AFA$, both z and its benign twin $y = \{x \mid x \in x\}$ are proper classes [Barwise and Etchemendy, 1987].

3.4 Solution Lemma

Much of the appeal of Aczel's system $ZFC^- + AFA$ lies in a consequence of the Anti-Foundation Axiom, the Solution Lemma, which establishes that a list of set-defining equations that express some "unknowns" in terms of hypersets over those unknowns themselves and some "givens" has a unique solution in the hypersets of the givens. It is this technique that will be exploited to establish a semantic theory for circular phenomena such as exist in SNePS. Systems of equations and their solutions are built up as in algebra [Barwise and Etchemendy, 1987]. The notation is dense, largely because it allows for infinite cases that will not be necessary for the present purposes. An example follows the definitions.

$\mathcal{V}_\mathcal{A}$: Given a set $\mathcal{A}$ of atoms, $\mathcal{V}_\mathcal{A}$ is the universe of all sets, well-founded and non-well-founded, called a *hyperuniverse*, with atoms from $\mathcal{A}$.

$\mathcal{V}_{\mathcal{A}'}$: Given a superset $\mathcal{A}' \supseteq \mathcal{A}$ of atoms, the hyperuniverse $\mathcal{V}_{\mathcal{A}'}$ is all sets with atoms from $\mathcal{A}'$, so $\mathcal{V}_{\mathcal{A}'} \supseteq \mathcal{V}_\mathcal{A}$.

Indeterminates $\mathcal{X}$: The difference in the sets of atoms is $\mathcal{X} = \mathcal{A}' - \mathcal{A}$, and the elements $\mathbf{x} \in \mathcal{X}$ are the *indeterminates* over $\mathcal{V}_\mathcal{A}$, which can be regarded as unknowns ranging over that hyperuniverse in the sense that they will be mapped to hypersets. A set $a \in \mathcal{V}_{\mathcal{A}'}$ is a term built up from the indeterminates in its transitive closure under hereditary membership.

Equation in $\mathcal{X}$: An expression of the form $\mathbf{x} = a$ is an equation in $\mathcal{X}$, where $\mathbf{x} \in \mathcal{X}$ and $a \in \mathcal{V}_{\mathcal{A}'} - \mathcal{X}$. (The exclusion of $\mathcal{X}$ from the possible right-hand sides rules out equations such as $s = s$, but not, of course, more interesting ones such as $s = \{\ldots, s, \ldots\}$.)

System of Equations in $\mathcal{X}$: A system of equations in $\mathcal{X}$ is a family of equations $\{\mathbf{x} = a_\mathbf{x} \mid \mathbf{x} \in \mathcal{X}\}$, one for each indeterminate $\mathbf{x} \in \mathcal{X}$.

Assignment f for $\mathcal{X}$: An assignment for $\mathcal{X}$ in $\mathcal{V}_\mathcal{A}$ is a function $f : \mathcal{X} \longrightarrow \mathcal{V}_\mathcal{A}$ which assigns to each indeterminate $\mathbf{x} \in \mathcal{X}$ an element $f(\mathbf{x})$ from $\mathcal{V}_\mathcal{A}$. Any such f extends to an $\hat{f} : \mathcal{V}_{\mathcal{A}'} \longrightarrow \mathcal{V}_\mathcal{A}$ with the replacement of each $\mathbf{x} \in \mathcal{X}$ by its value $f(\mathbf{x})$.

[2] The Axiom of Comprehension is sometimes stated in the naive way, without this protection, as for instance, by Azriel Levy:

$$\exists y \forall x (x \in y \iff \phi)$$

He explains,

> According to one view, the axiom of comprehension is basically false, since it represent a mental act of "collecting" all sets which satisfy $\phi(x)$, and this cannot be done since we can "collect" only those sets which have been "obtained" at an "earlier" stage of the game...The other possible reaction to Russell's antinomy is to continue believing in the essential truth of the axiom schema of comprehension, viewing the Russell antinomy as a mere practical joke played on mankind by the goddess of wisdom. According to this point of view ...we should use [it] only in order to obtain new sets which are not too "large" compared to the sets whose existence is already assumed in the construction. ...In our framework of set theory both approaches lead to the same result ...[Levy, 1979, page 7]

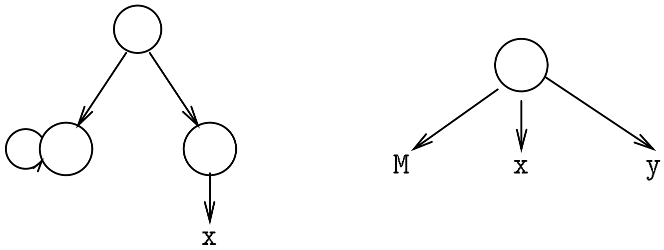


Figure 3.4: Pictures of sets defining indeterminates x and y

Solution of an Equation: An assignment f is a solution of an equation $\mathbf{x}_1 = a(\mathbf{x}_1, \mathbf{x}_2, \ldots)$ if $f(\mathbf{x}_1) = a(f(\mathbf{x}_1), f(\mathbf{x}_2), \ldots)$. For $\mathbf{x}_1$, the notation $a(\mathbf{x}_1, \mathbf{x}_2, \ldots)$ indicates any set composed of elements from $\mathcal{V}_{\mathcal{A}'}$, including, possibly, the indeterminates $\mathbf{x}_1, \mathbf{x}_2, \ldots$. The only restriction on the right-hand side is that it can't be simply $\mathbf{x}_1$ itself.

Solution of a System of Equations: An assignment f is a solution of a system of equations in $\mathcal{X}$ if it is a solution of each equation in the system.

The Solution Lemma itself states:

> **Every system of equations in a collection $\mathcal{X}$ of indeterminates over $\mathcal{V}_{\mathcal{A}}$ has a unique solution.**

The status of these indeterminates may be confusing. They are treated first as atoms, when the set of indeterminates $\mathcal{X}$ is given as $\mathcal{A}' - \mathcal{A}$, and later as sets, when they are assigned sets in the solution to the system. This is correct. They are atoms in the beginning because they have no known set structure. The solution f, however, maps them to hypersets; they become, for all intents and purposes, placeholders for sets, and questions of their ontological status are beside the point. The thrust of the Solution Lemma is that the elements of $\mathcal{X}$, which have no obvious relationship to the original hypersets $\mathcal{V}_{\mathcal{A}}$ (except possible inclusion of elements from $\mathcal{A}$), which are formulated antecedently to the introduction of those indeterminates $\mathcal{X}$, still have solutions in $\mathcal{V}_{\mathcal{A}}$.

For example (from [Barwise and Etchemendy, 1987]), consider the following system of equations for $\mathcal{X} = \{\mathbf{x}, \mathbf{y}\}$, where $\mathtt{M}$ is an arbitrary atom:

$$\begin{aligned} \mathbf{x} &= \{\Omega, \{\mathbf{x}\}\} \\ \mathbf{y} &= \{\mathtt{M}, \mathbf{x}, \mathbf{y}\} \end{aligned}$$

So $\mathcal{A}$, the given set of atoms, is simply $\{\mathtt{M}\}$, and $\mathcal{V}_{\mathcal{A}}$ is all hypersets over $\{\mathtt{M}\}$. The solution that we seek, by the definitions above, will be an assignment from these hypersets to the indeterminates $\mathbf{x}$ and $\mathbf{y}$, so far only defined by hypersets from the larger universe $\mathcal{V}_{\mathcal{A}'}$, that is, hypersets over $\mathcal{A}' = \mathcal{A} \cup \mathcal{X} = \{\mathbf{x}, \mathbf{y}, \mathtt{M}\}$. (This larger domain $\mathcal{A}'$ happens to be $\mathbf{y}$ itself.) The pictures in Figure 3.4 show the two sets on the right-hand sides of the equations.

To show the solutions, we replace the indeterminate $\mathbf{x}$ with the apg that is its picture (on the left in Figure 3.4), and the same for $\mathbf{y}$, as in Figure 3.5. Notice that what happens is simply that, in the first apg, the arc pointing to $\mathbf{x}$ moves up to the point—since this is the apg for $\mathbf{x}$—and in the second, the arcs to $\mathbf{x}$ and $\mathbf{y}$ are similarly "re-pointed", the one for $\mathbf{x}$ to a new subgraph.

The Anti-Foundation Axiom states that these graphs have unique decorations with the sets depicted by the top nodes as the solutions to the system of equations. Let $a = \{\mathtt{M}, \Omega, a\}$, depicted by the graph given in Figure 3.6.

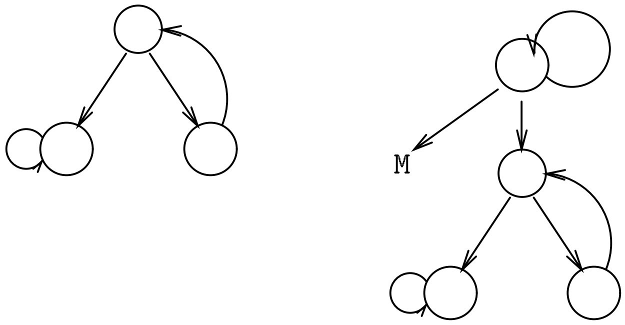


Figure 3.5: Construction of solution f

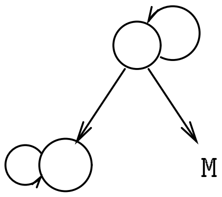


Figure 3.6: $a = \{\mathtt{M}, \Omega, a\}$

Then the solution is:

$$f(\mathbf{x}) = \Omega$$
$$f(\mathbf{y}) = a$$

We show this by substituting the assigned elements for the indeterminates in the equations, in accordance with the definition of **solution of a system of equations**.

$$\begin{array}{lcl}
f(\mathbf{x}) & \stackrel{?}{=} & \{\Omega, \{f(\mathbf{x})\}\} \\
 & \text{Yes, because} & f(\mathbf{x}) = \Omega = \{\Omega, \{\Omega\}\} = \{\Omega, \{f(\mathbf{x})\}\} \\
f(\mathbf{y}) & \stackrel{?}{=} & \{\mathtt{M}, f(\mathbf{x}), f(\mathbf{y})\} \\
 & \text{Yes, because} & f(\mathbf{y}) = a = \{\mathtt{M}, \Omega, a\} = \{\mathtt{M}, f(\mathbf{x}), f(\mathbf{y})\}
\end{array}$$

And how was a derived?—crudely, by conflating the circular subgraph into a single picture of Ω. The derivation of a solution (at least, one that allows for Ω) will not be pursued in depth yet because it is not strictly necessary to this discussion, which deals with the *existence* of a solution.

As required, the two objects assigned to the indeterminates by the solution f, Ω and the set shown called a, are members of $\mathcal{V}_{\mathcal{A}}$, the hyperuniverse over the set of atoms $\mathcal{A} = \{\mathtt{M}\}$. The Solution Lemma tells us that this is the only solution; any other solution to that system of equations will be equivalent to it.

The proof of the Solution Lemma in the axiomatization $ZFC^{-} + AFA$ is given by Aczel, who also shows that the Anti-Foundation Axiom AFA is provable from the Solution Lemma in ZFC^{-}. Since, then, the Solution Lemma is just a restatement of the Anti-Foundation Axiom, those who find its establishment of a unique solution too facile may take comfort in viewing the Solution Lemma as the axiom of interest, rather than as a result.

3.5 Analogy to a Ring of Polynomials

In [Aczel, 1988, pages 11–12], Aczel mentions that the technique of constructing sets by adjoining atoms to a given universe has an analogy in ring theory. The reader with mathematical experience may find the development of the Solution Lemma easier to grasp through its similarity to the construction of a ring $R[X]$ of polynomials from a ring R. The reader without such experience may be assured of the respectability of the technique.

Let $(R, +, \cdot, 0, 1)$ be a commutative ring. That means, for all a, b, and c in the set R:

1. $a + b \in R$ and $a \cdot b \in R$
2. $(a + b) + c = a + (b + c)$
3. $a + b = b + a$
4. $a + 0 = a = 0 + a$
5. $\forall a \exists\, a' \, [a + a' = 0 = a' + a]$ (Read a' as $-a$.)
6. $(a \cdot b) \cdot c = a \cdot (b \cdot c)$
7. $a \cdot 1 = a = 1 \cdot a$
8. $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$ and $(a + b) \cdot c = (a \cdot c) + (b \cdot c)$

The properties (2) through (5) can be stated, "R is an abelian group," and (6) through (7), "R is a monoid."

For $R[x]$, the polynomials over R in one unknown x, we take the members to be the infinite sequences $(a_0, a_1, a_2, \ldots)$ for which $\exists\, i\, [a_j = 0, j \geq i]$. These a_i represent the coefficients of the increasing powers of x. Since all coefficients are zero beyond some i, the "degree," these are finite polynomials. One of the sequences, for example, is $(4, 17, 0, 9, 803, 0, 0, 0, \ldots)$, corresponding to the polynomial $4 + 17x + 9x^3 + 803x^4$. Identity holds between two polynomials if and only if all of their coefficients are equal.

The "+" and "·" operators, and the "0" and "1" elements are defined to provide the other components of a ring.

1. $(a_0, a_1, \ldots) + (b_0, b_1, \ldots) = (a_0 + b_0, a_1 + b_1, \ldots)$; this composition is a member of $R[x]$.
2. $(a_0, a_1, \ldots) \cdot (b_0, b_1, \ldots) = (p_0, p_1, \ldots)$, where $p_i = \sum_{j+k=i} a_j b_k$; this composition is also in $R[x]$.
3. The zero element $0 = (0, 0, \ldots)$.
4. The one element $1 = (1, 0, 0, \ldots)$.

These elements and operations can be shown to satisfy the requirements above, so $(R[x], +, \cdot, 0, 1)$ is a ring (with R a subring, represented by those polynomials of degree zero) [Jacobson, 1974].

There is a member of $R[x]$ which can be interpreted as the indeterminate x itself, $x = (0, 1, 0, 0, \ldots)$. The powers of x are also in there, x^n as $(0, 0, \ldots, 1, 0, 0, \ldots)$, with the "1" in the a_n place. Any $a \in R$ is in $R[x]$ as $(a, 0, 0, \ldots)$, and an individual term ax^n as $(0, 0, \ldots, a, 0, 0, \ldots)$, with the a in the a_n place.

Successive adjunction (i.e., addition) of $x_1, x_2, \ldots, x_n$ forms the ring $R[x_1][x_2] \ldots [x_n]$, written $R[x_1, x_2, \ldots, x_n]$, and consisting of all sums $\sum a_{\alpha_1 \ldots \alpha_n} x_1^{\alpha_1} \ldots x_n^{\alpha_n}$.

If $f(x) = \sum_v a_v x^v$ is a polynomial over R and $\alpha \in R$ (or in an overring of R) and commutes with all elements of R, then replacement of x by α gives $f(\alpha) = \sum_v a_v \alpha^v$. Given another polynomial in one indeterminate $g(x)$, consider the sum $s(x) = f(x) + g(x)$ and the product $p(x) = f(x) \cdot g(x)$ as defined above. We want it to be the case that $f(\alpha)+g(\alpha) = s(\alpha)$, and that $f(\alpha)\cdot g(\alpha) = p(\alpha)$ when the value $x = \alpha$. The property for the sum is easily verified directly from the definition of "+" for $R[x]$, and for the product:

$$p(\alpha) = \sum_{v}\sum_{j+k=v} a_j b_k \alpha^v = \sum_{j}\sum_{k} a_k b_k \alpha^{j+k} = (\sum_{j} a_j \alpha^j)\cdot(\sum_{k} b_k \alpha^k) = f(\alpha)\cdot g(\alpha)$$

All relations, then, between $f(x)$ and $g(x)$ derived from the ring operations are preserved if x is replaced by a member of the ring $R[x]$ which commutes with all elements of R. Furthermore, the same relations hold when several indeterminates are adjoined [van der Waerden, 1991]. In other words, the indeterminates $x_1, x_2, ..., x_n$, the adjunction of which created a new ring, have solutions in the original ring R, much as the indeterminates $\mathcal{X}$ can be assigned, by the Solution Lemma, hypersets over the original atoms $\mathcal{A}$ that satisfy the equations defining the indeterminates $\mathcal{X}$.

3.6 Application to Terminological Cycles

Robert Dionne, Eric Mays, and Frank J. Oles apply the disciplined circularity of non-well-founded sets to the semantics of a term subsumption language, K-REP [Dionne *et al.*, 1992]. The stated motivation is the provision of intensionality in concept descriptions.

Terminological cycles, as noted by Nebel (cf. § 2.7.2), occur naturally in such a language, intended to capture word definitions. In K-REP, the concepts—the formal terms of the language—are defined by a set of equations, possibly involving mutual recursion reflecting interdependent reference. Thus, concepts are represented by accessible pointed graphs. A "universal concept algebra" $\mathcal{C}$ is derived, with Aczel's Solution Lemma deployed to guarantee the existence and uniqueness of a solution, in $\mathcal{C}$, to the system of concept equations constituting the knowledge base. The elements of $\mathcal{C}$ are intensions rather than extensions because they are built from descriptions of the roles of the concept. The analysis done by Dionne *et alia* is mathematically very detailed and couched in the formal terms of a comprehensive representational theory (K-REP), hence is not an obvious analog to the development undertaken here, but it is encouraging to see the same notion applied to solve a parallel problem.

3.7 Summary and Intended Application

Sets that have as their members other sets, possibly themselves, can be used to describe circular phenomena. As developed by Peter Aczel under the rubric "non-well-founded set theory", these sets are denoted by directed graphs, with the directed arcs signifying membership. His theory "lets us bring to bear all of the familiar set-theoretic techniques to the problem of modeling circular phenomena." [Barwise and Etchemendy, 1987, page 58]

We have seen that the theory of non-well-founded sets is an extension of classical set theory as it is commonly used in mathematics (that field having little need for the Axiom of Foundation). The Anti-Foundation Axiom did not originate with Aczel, although he was apparently "the first to see that AFA could be obtained from a coherent, intuitive conception of set, rather than just being a formally consistent axiom, and to demonstrate that it is an important mathematical tool for the modeling of various kinds of real-world circularity, not just a mathematical curiosity". [Barwise and Etchemendy, 1987, page 58]

The purpose here is modelling of the circular semantics of SNePS, and we now turn to that.

Chapter 4

Circular Semantics in Network Models

4.1 Introduction

In this chapter, the hyperset mechanism described in the previous chapter is applied to the problem of finding a semantics for SNePS that accommodates its goals and development. Alternatives are considered, and one, in which base nodes are assigned sets of semantic elements that include their parent molecular nodes, is selected and developed in depth. Finally, the results derived are discussed in terms of the principles of SNePS.

4.2 Circularity of Meaning

Let us explore the possibility of circularity in a semantic network representation such as SNePS, the graphical structure of which might nicely accommodate Aczel's theory. To facilitate the focus on the graphical structure, we will work with the "unigraph" version S' of a given SNePS network $S \in$ SNets (cf. §2.4), in which there is at most one arc between nodes, and arcs are not labeled with relations. Relations will be incorporated into the semantics in §7.2. Node labels will be retained to enable reference to them. Three questions must be answered:

1. How can the nodes, the meaning-bearing objects of SNePS, be defined as sets or hypersets?
2. Where is the circularity?
3. Which nodes are atomic?

Alternative sets of answers are considered below. All of them treat SNePS arcs as defining hereditary membership. A node n_1 is a member of the set assigned to the node n_2 if there is an arc from n_2 to n_1. In other words, the definition from [Shapiro, 1991] of the membership of a wire $\langle r, n\rangle$ in a cableset (molecular node) m will be a sufficient condition for n's membership in m's hyperset—but not a necessary condition, as we shall see in the development of answers to question 2. KL-ONE also allows a set-theoretic interpretation of arcs:

> Given two KL-ONE descriptions, an important question to consider is whether one *subsumes* the other—that is, whether an instance of one is always an instance of the other. In semantic nets, this question usually comes down to looking from one node up the hierarchy to see if another happens to lie on a superset path. In KL-ONE, the subsumption question can also be answered by looking up a hierarchy, with one crucial difference

> ...the descriptions must be in their proper places in the network before any conclusions can be drawn. [Brachman and Schmolze, 1985, page 178]

So there is nothing revolutionary in this treatment of arcs as set membership in a graph S', and that will be the answer to question 1. The remaining specifications—the attribution of circular and atomic qualities—will require more discussion. We first consider rendering all nodes circular, and then limiting circularity to particular classes of nodes.

4.3 All Nodes Are Circular

4.3.1 Nodes Are Self-Circular

One uniform way to incorporate circularity is to add a self-loop, an arc from n to n, to each node n. Each node then consists of a set as defined above, but which also contains itself. This does not seem well-motivated cognitively. That's just as well, since the theoretical results are uninteresting—the network is a picture of Ω. There are no atomic nodes, and all nodes are Ω themselves, so the network itself (if connected; otherwise, each connected subnetwork in it) has no decoration other than Ω.

4.3.2 All Arcs Are Bi-Directional

Another way to make nodes circular, perhaps more intuitive, is to define the set-inclusion relationship to go in both directions along an arc; that is, as a molecular node includes its child nodes as members, so does each child include its parent as a member. This construal, however, leads to the same result as that above. No nodes are atomic, and since there would be a path from each node to every other, every node in a cycle (in each connected component of S'), the entire network (if connected; otherwise, each connected subnetwork in it) is a picture of Ω.

4.4 Circularity Is Granted to Nodes by Type

The problems with the proposals above is that no nodes are left atomic, to form the ground elements for the semantics of others. There are two obvious choices for atoms, sensory nodes and base nodes, whichever is left after the other category is determined to be the better choice for the circularity.

4.4.1 Sensory Nodes Are Circular

Sensory nodes form a qualitatively distinct subset of the nodes of a network, providing the interface to the external world. They have outdegree zero (as do base nodes). Could sensory nodes influence their parents? The idea has some appeal, especially in terms of the word 'sensory', taken to be input. But other sensory nodes are actions, output. It is not clear that an act should influence the meaning of the cognitive structures that caused it, at least not in that cognitive agent itself. (The situation might be different for *observing* cognitive agents.) Do we want even strictly-input sensory nodes, such as the words at the heads of `LEX` arcs, to *influence* the meanings of the nodes above? Should they not rather *contain* those meanings? For cognitive modeling, the latter; the naked string of letters 'cigar' is a receptacle of meaning, not a generator. Certainly the concept of cigars affects other closely-related concepts, but that effect is made through the concepts, not the words—unless all (English) words are onomatopoeic. Act nodes—the other sensory nodes—are certainly best viewed as receptacles of meaning, also. In short, sensory nodes should not be involved in circularity.

4.4.2 Base Nodes are Circular

SNePS already has a locus for concepts, the base node. Base nodes need semantics, in fact, semantics that influence nodes above them. Suppose `n43` is the node in some cognitive agent representing "the New York State Legislature". It is not just a conceptual sink, but a source—with impact on the meanings of the molecular nodes that use it. The meaning of something includes, at least, all that we know *about* it. For these reasons, we will take base nodes to be the circular structures for the application of non-well-founded set theory to formalize the semantics of a SNePS network S. The semantics can be shown—in both a superficial and profound sense, as we hope to demonstrate—by a new graphical structure called S^* that is derived from $S' = (V, E')$.

Definition 4.4.1 *Given a unigraph S', the stargraph S^* is the graph consisting of the set of nodes V from S and the set of edges E^*, where $\langle x, y\rangle \in E^*$ if and only if (1) $\langle x, y\rangle \in E'$, or (2) x is a base node and $\langle y, x\rangle \in E$ (or, equivalently, $\in E'$).*

The three necessary specifications, answers to the introductory questions, are as follows. (To review SNePS terms, see §2.5.)

1. A (molecular) node in S' will be construed as a set in the standard hereditary sense, so that its "members" are its immediately subordinate nodes, following the directed arcs formed under the SR.i rules of [Shapiro and Rapaport, 1987]. Support for this simple view comes from [Shapiro, 1991, page 146]: "... a node is determined by the arcs emanating from it, not by the arcs pointing into it." For the intital development, the arc label will be ignored, but since it contributes to the meanings of the two nodes connected, eventually (§7.2.4), it will be incorporated by virtue of the definition of a molecular node as a set of *wires* (ordered pairs of arc-label relations and nodes, as described in §2.5).

2. Base nodes will be regarded as circular, both being influenced and influencing their parent molecular nodes. In other words, the set at a base node will include its parents as members; base nodes will be assigned non-empty sets in the hereditary construal. Graphically, this will be pictured by a new graphical structure derived from S' called S^*, in which (unlabeled) arcs are added from base nodes to their parent nodes.

3. Sensory nodes will be regarded as strictly atomic, with no set-theoretic structure. Since a sensory node is always at the end of some arc in S' and therefore in S^*, it will be a member of the set at its parent molecular node, contributing simply the lexeme or other sensory datum associated with it, but its outdgree is zero; it contains nothing.

   In analysis of subnetworks, other "childless" nodes will appear, because the full SNePS network will be circumscribed by some boundary that cuts off the rest of the connections. We will generalize, then, to the policy that, given some finite network without regard to its place in a larger one, *any* node of outdegree zero is treated as an atom. This means that molecular nodes at the edge of a defined subnetwork will be treated as atomic in the same way that sensory nodes are. Note that a base node retains its cyclic connection to some parent, even if, in a subnetwork, some of its dominating molecular nodes are cut off, so the outdegree of a base node in S^* is always greater than zero.

Molecular nodes could certainly be made circular, also. We elect not to do this, on the grounds that they already have outgoing arcs that allow a hereditary set-theoretic interpretation, and that this first plunge into circularity should be predicated on minimal changes to SNePS networks, but that approach is also explored later (§7.7).

Examples of a SNePS network S and its derived S' and S^* follow.

4.5 SNePS Structures for Non-Well-Founded Sets

Using the approach developed above that locates circularity in base nodes, consider the network implementing the sentence "Nancy asked Tom whether an inanimate object, such as a table, can exert a force" [Rapaport, 1988a], as modeled in some unspecified observer (possibly, but not necessarily, Nancy or Tom), reproduced in Figure 4.1. This is a SNePS network $S \in$ SNets. Figure 4.2 shows the S' version. Since the orginal does not have multiple arcs between nodes, the only obvious difference is the lack of arc labels. The non-well-founded set framework version S^* is shown in Figure 4.3, with "backward" arcs from base nodes. We will now use Aczel's theory to computer hyperset semantics for some of the nodes in a restricted portion of the network S.

4.6 Using the Solution Lemma to Compute Semantics

To simplify the discussion, we focus on only a portion of the network S'—the upper-left subgraph, where Nancy is asking Tom something. Call that subnetwork W', as shown in Figure 4.4.[1] (In other such examples, the truncated subgraph will not be explicitly shown.) This gives the derived graph W^* shown in Figure 4.5. W^* is an accessible pointed graph, of which the point is `m25`, and other nodes forms the points of their own apgs. There are three atoms, shown in elliptical nodes—**`Nancy`**, **`ask-whether`**, and **`Tom`**. The labels on the other nodes are for convenience only, just as they are in general in SNePS. Decorations for each node are computable according to the hereditary membership principle. The node labeled **`m4`**, for example, has as its decoration the set $\{\texttt{Tom}\}$; node **`m5`** has the set $\{\texttt{b2}, \texttt{m4}\} = \{\texttt{b2}, \{\texttt{Tom}\}\}$, etc. The more difficult question is what exactly is the set at `b2`, which includes only **`m5`** and its other parent **`m24`**, both of which themselves include `b2`? Indeed, what is the full specification of any of those sets, respecting the circularity involved? To work up to the answer, we take a simpler example first, developing it in some detail, and then tackle the question for `b1` and `b2`.

4.6.1 Simple example

Consider the very small subnetwork Z^* shown in Figure 4.6.[2]

Let us solve for the hypersets at nodes `m2` and `b1`. In other words, let those nodes be the indeterminates, which means that what we are doing is finding the hypersets that express `m2` and `b1` in terms of the givens, the atoms—in this case, just `m1`. We have truncated the graph, rendering `m1` atomic. So $\mathcal{A} = \{\texttt{m1}\}$ and $\mathcal{X} = \{\texttt{m2}, \texttt{b1}\}$. Recall that $\mathcal{A}'$ is the set of atoms extended by the indeterminates, that is, $\mathcal{A}' = \mathcal{A} \cup \mathcal{X}$. The solutions will be found in the hyperuniverse $\mathcal{V}_{\mathcal{A}}$. The system of equations from which to start will express each indeterminate in terms of a hyperset over $\mathcal{V}_{\mathcal{A}'} - \mathcal{X}$. [3]

$$\begin{aligned} \texttt{m2} &= \{\texttt{m1}, \texttt{b1}\} \qquad (4.1) \\ \texttt{b1} &= \{\texttt{m2}\} \qquad (4.2) \end{aligned}$$

Each right-hand side is a hyperset over the atoms $\mathcal{A}$ and indeterminates $\mathcal{X}$, as required. The solution will be an assignment to each indeterminate of a hyperset from $\mathcal{V}_{\mathcal{A}}$ *such that the defining equations still hold when the corresponding values are substituted for the indeterminates.* In other words, under the solution $f : \{\texttt{m2}, \texttt{b1}\} \longrightarrow \mathcal{V}_{\{\texttt{m1}\}}$, these statements (derived from Equations 4.1 and

[1] We are not employing the SCOPE mechanism, just marking off a relatively self-contained piece.

[2] Which could be defined from W^* as SCOPE(`m2`, 1)

[3] Recall also that the subtraction of the set of indeterminates $\mathcal{X}$ from the possible right-hand sides is meant only to exclude vacuous equations like `m2` = `m2` from the system; this restriction will be assumed hereafter.

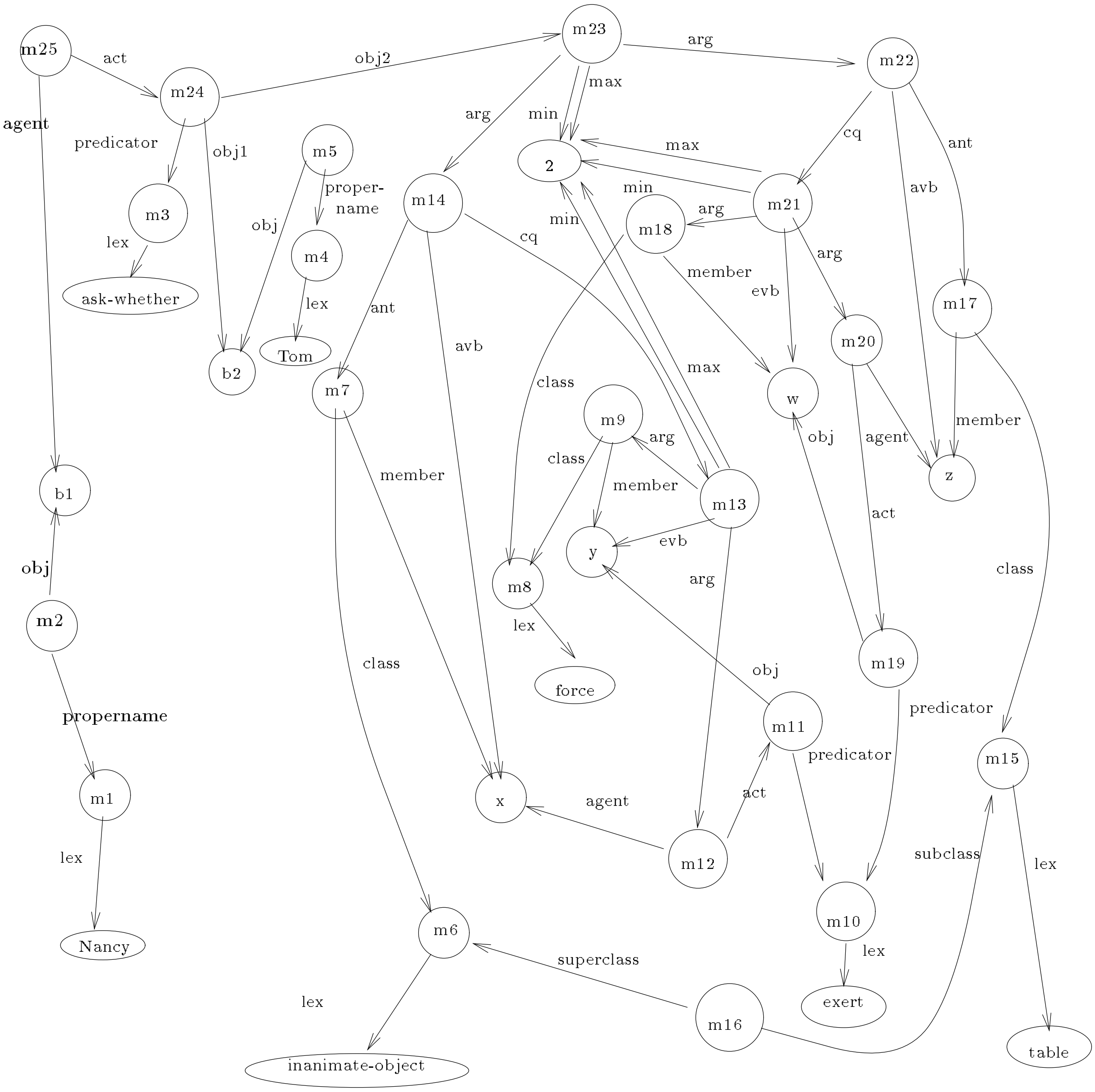


Figure 4.1: TABLE sentence as a SNePS network S

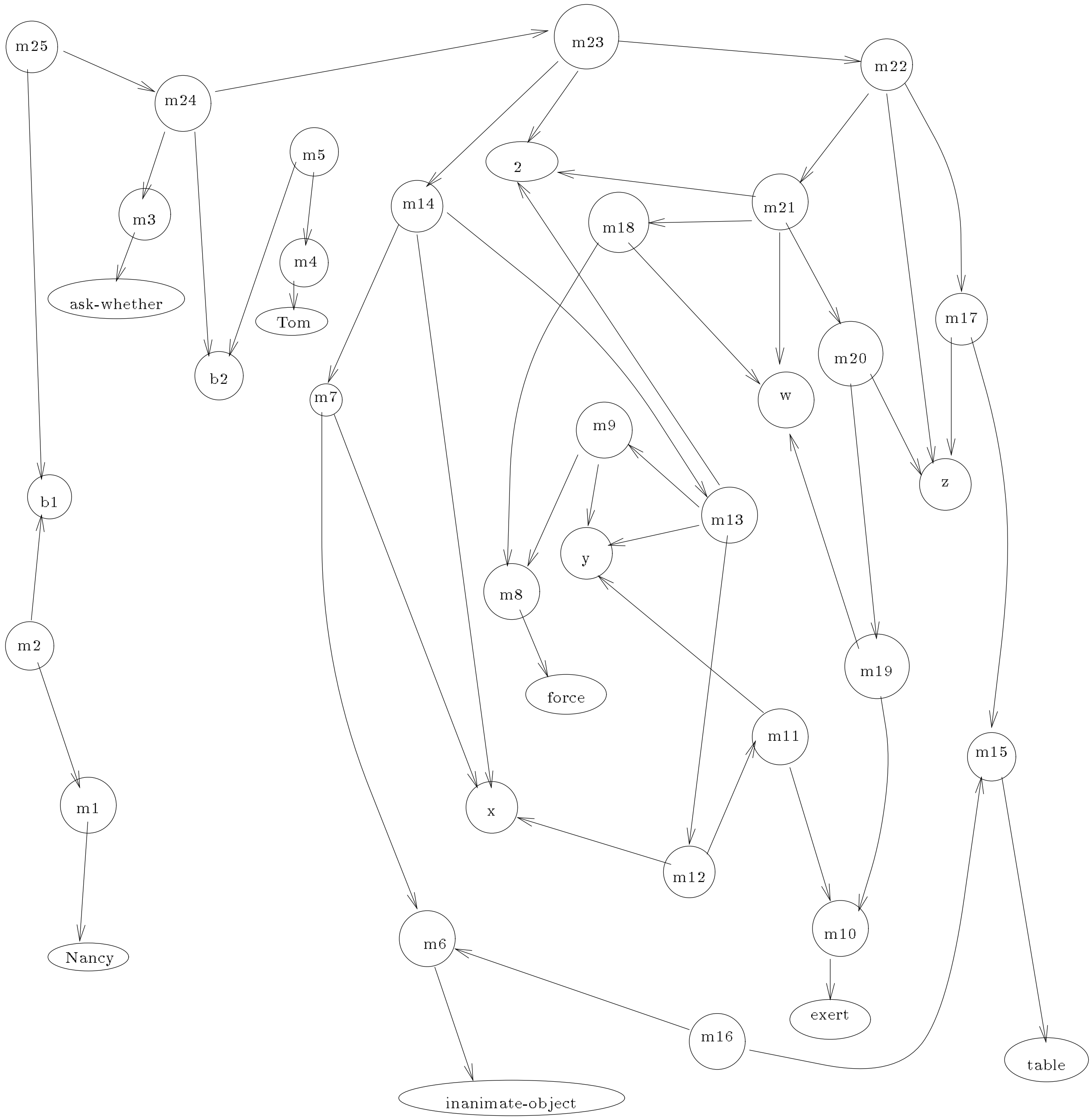


Figure 4.2: TABLE sentence as a SNePS unigraph S'

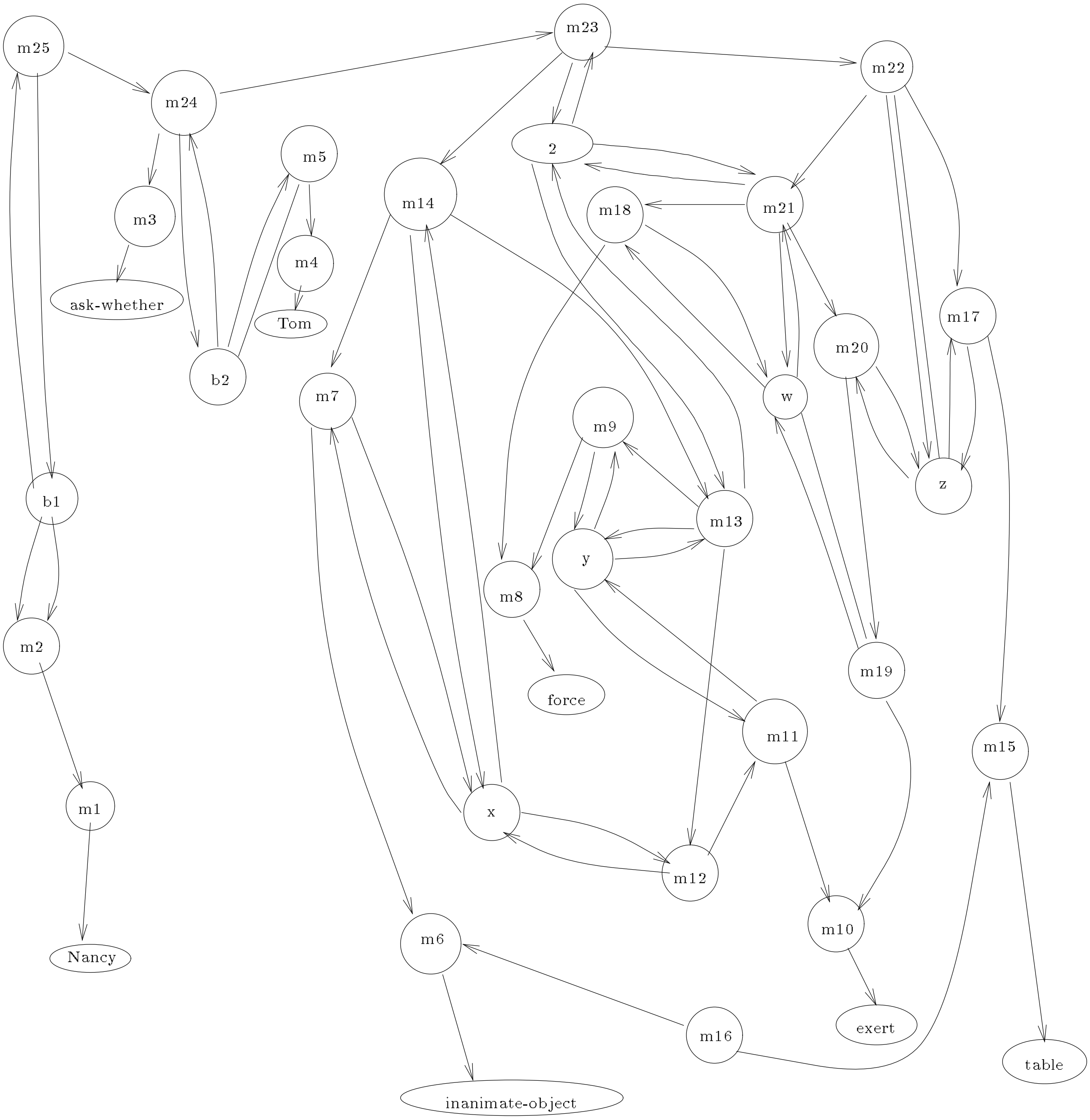


Figure 4.3: TABLE sentence as a SNePS non-well-founded set S^*

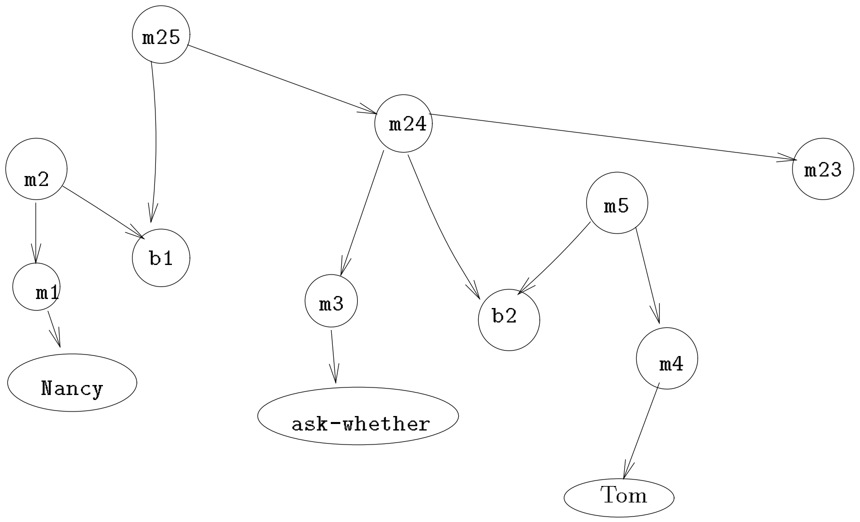


Figure 4.4: Subnetwork W' of TABLE example

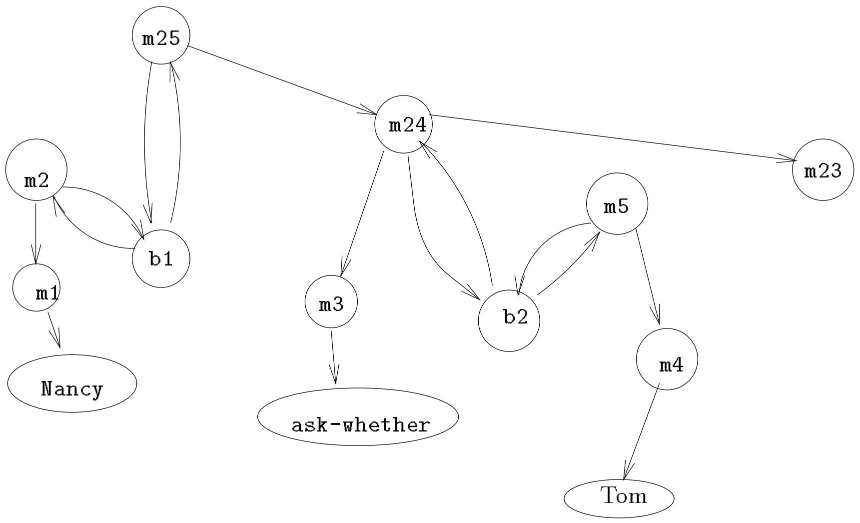


Figure 4.5: W^* from TABLE example S^*

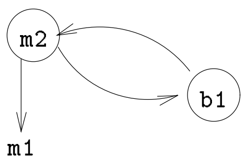


Figure 4.6: The small derived subnetwork Z^*

Figure 4.7: Assignments under proposed solution f to the indeterminates in Z^*

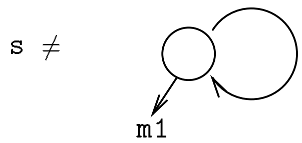
4.2) will be true, also:

$$f(\mathtt{m2}) = \{\mathtt{m1}, f(\mathtt{b1})\} \tag{4.3}$$
$$f(\mathtt{b1}) = \{f(\mathtt{m2})\} \tag{4.4}$$

Let s be the set such that $s = \{\mathtt{m1}, \{s\}\}$. The proposed solution (obtained, as usual, through simple inspection) is:

$$f(\mathtt{m2}) = s \tag{4.5}$$
$$f(\mathtt{b1}) = \{s\} \tag{4.6}$$

These hyperset assignments under f are given graphically in Figure 4.7. Note that there are superficially similar sets and pictures that are not s. For examples, $s \neq \{s\}$ (if so, it would be Ω), and



In other words, there is a genuine and substantive structural claim being made.

The proposed solution is tested by evaluating the system of equations under substitution of the mapped value for the indeterminate:

$$\begin{array}{lcll} f(\mathtt{m2}) & = & s & \text{(proposed solution)} \\ & = & \{\mathtt{m1}, \{s\}\} & \text{(definition of } s\text{)} \\ & = & \{\mathtt{m1}, f(\mathtt{b1})\} & \text{(definition of } f \text{ for } \mathtt{b1}\text{)} \\ & & QED & \text{Equation 4.3 is verified.} \end{array}$$

$$\begin{array}{lcll} f(\mathtt{b1}) & = & \{s\} & \text{(proposed solution)} \\ & = & \{f(\mathtt{m2})\} & \text{(definition of } f \text{ for } \mathtt{m2}\text{)} \\ & & QED & \text{Equation 4.4 is verified.} \end{array}$$

The solution pictures look very much like the orginal network. There are other pictures of the solution assignments $f(\mathtt{m2})$ and $f(\mathtt{b1})$, of course,[4] but any will do; it is the *set depicted* that is the solution, and its (unique) existence is guaranteed by the Solution Lemma. We now have hypersets over $\mathcal{A} = \{\mathtt{m1}\}$ that can serve as meanings of `b1` and `m2`.

[4]For example, $f(\mathtt{b1})$ is $\{s\}$, which could be depicted by the same picture as $f(\mathtt{m2})$, but with the *other* node as the point, as below.

4.6.2 Larger example

Now, using Aczel's framework, let us solve the larger problem in W^*, for the circular nodes `b1` and `b2` in terms of the others. Since they comprise the indeterminates,

$$\mathcal{X} = \{\texttt{b1}, \texttt{b2}\},$$

and

$$\mathcal{A} = \{\texttt{Nancy}, \texttt{ask-whether}, \texttt{m23}, \texttt{Tom}\}.$$

The setting is restricted to the subnetwork shown in Figure 4.5, entailing that `m23` be taken as an atom, rather than as a set with its own subordinates/members. The universe $\mathcal{V}_{\mathcal{A}}$, then, is all hypersets over those elements $\mathcal{A}$. The solution sought will be an assignment f of sets from $\mathcal{V}_{\mathcal{A}}$ to `b1` and `b2`. The hyperuniverse $\mathcal{V}_{\mathcal{A}'}$ is all hypersets over $\mathcal{A} \cup \mathcal{X}$. We need a system of equations defining `b1` and `b2`, where each set on the right-hand side is in $\mathcal{V}_{\mathcal{A}'}$:

$$\begin{aligned}
\texttt{b1} &= \{\texttt{m2}, \texttt{m25}\} = \{\{\{\texttt{Nancy}\}, \texttt{b1}\}, \{\texttt{b1}, \{\{\texttt{ask-whether}\}, \texttt{b2}, \texttt{m23}\}\}\} \\
\texttt{b2} &= \{\texttt{m24}, \texttt{m5}\} = \{\{\{\texttt{ask-whether}\}, \texttt{m23}, \texttt{b2}\}, \{\texttt{b2}, \{\texttt{Tom}\}\}\}
\end{aligned}$$

As before, what we seek is a set for `b1` and a set for `b2` such that:

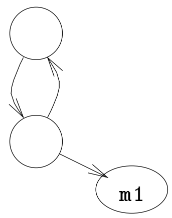
$$f(\texttt{b1}) = \{\{\{\texttt{Nancy}\}, f(\texttt{b1})\}, \{f(\texttt{b1}), \{\{\texttt{ask-whether}\}, f(\texttt{b2}), \texttt{m23}\}\}\} \tag{4.7}$$
$$f(\texttt{b2}) = \{\{\{\texttt{ask-whether}\}, \texttt{m23}, f(\texttt{b2})\}, \{f(\texttt{b2}), \{\texttt{Tom}\}\}\} \tag{4.8}$$

Learning from experience, we suspect that the solution is pretty much just the relevant subgraphs. The full network contains the solution, as we would expect. The unlabeled subnetwork rooted at `b1` is the assignment under f to the unknown `b1`, and the subnetwork rooted at `b2`, the assignment under f to the unknown `b2`, as shown in Figure 4.8. Verification is obvious:

$$\begin{aligned}
f(\texttt{b1}) &= \{\{\{\texttt{Nancy}\}, f(\texttt{b1})\}, \{f(\texttt{b1}), \{\{\texttt{ask-whether}\}, f(\texttt{b2}), \texttt{m23}\}\}\} && \text{(proposed solution)} \\
& \quad QED && \text{Equation 4.7 is verified.} \\
f(\texttt{b2}) &= \{\{\{\texttt{ask-whether}\}, \texttt{m23}, f(\texttt{b2})\}, \{f(\texttt{b2}), \{\texttt{Tom}\}\}\} && \text{(proposed solution)} \\
& \quad QED && \text{Equation 4.8 is verified.}
\end{aligned}$$

Compare the hyperset assignment to `b1` given here, in Figure 4.8, to the assignment given in Figure 4.7 under the solution worked out in the earlier examples. They differ because the context W^* is wider in this second example; the set of atoms $\mathcal{A}$ is larger, and the solutions, of course, are always hypersets from those atoms. The meaning becomes more refined as the context widens, as we would hope.

Derivation and verification of the actual solution is tedious and anticlimactic, for reasons to be explored. The point of the Solution Lemma is that a solution in terms of the original hyperuniverse $\mathcal{V}_{\mathcal{A}}$ exists. (As discussed in Chapter 3, the existence of the solution is not dependent on non-well-foundedness; what matters here is that it is not thwarted by non-well-foundedness.)



The reader should verify that all members of one set are in the other, and vice versa, at every level.

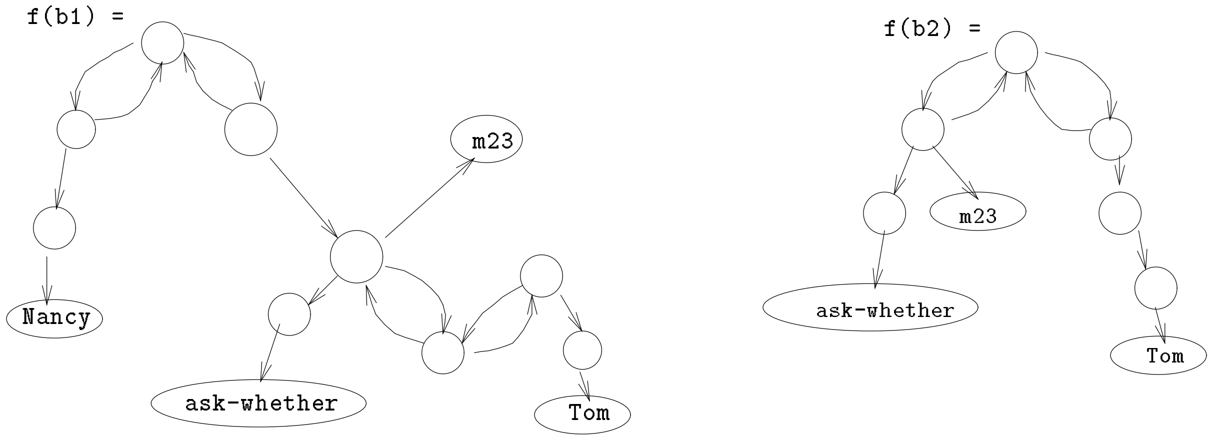


Figure 4.8: Assignments to indeterminates `b1` and `b2` in W^*

4.6.3 Cognitive significance

Informal interpretations or "glosses" of these apgs, in terms of everyday mental life, are not immediately obvious. The best that can be done is to say that this agent's current concept of Nancy, embodied in the node `b1`, includes her name and the activity of asking whether something-or-other, of somebody else, the somebody-else represented by the node `b2`, whose name is "Tom". But that's beyond the strict set-representational semantics; all it can do is demonstrate membership in `b2` of a thing known as "asking-whether", and, at an even more indirect level, a thing known as "Tom", etc.

4.7 A Semantic Function μ

The techniques described above enable the formulation of a semantic function μ for SNePS networks (as modeled in their S^* form), in which, broadly, the domain of meanings $\mathcal{Z}$ is sets with pictures—the same as the syntactic domain $\mathcal{Y}$. The meaning of a molecular node will be the union of the meanings of all its children, and the meaning of a base node will be the solution to the equation expressing it as a hyperset over the union of the meanings of its dominating nodes (in which the base node itself participates). Both sensory nodes and molecular nodes at the edge of a circumscribed network will be treated as atomic, not themselves in the domain of μ, but contributing their values to the meanings of other nodes. The definition of the function μ that accomplishes this will require some axioms, using the definitions of **SNodes** and **SNets** from §2.6.2 along with some new ones.

Definition 4.7.1 ***SNets***$'$: *the set of unigraphs resulting from the transformation of each* $S \in$ *SNets into* S', *as described in §2.4.*

Definition 4.7.2 ***SNets****: *the set of graphs resulting from the transformation of each* $S \in$ *SNets into* S^*.

Note that the members of $SNets^*$ are not necessarily apgs, since they may not be connected graphs, and since they may not have unique points. The example S^* in Figure 4.3 is not an apg for this latter reason; there is no unique node (the point) from which every other is reachable.

Definition 4.7.3 ***Full Network***: *a SNePS network* $S \in$ *SNets that embodies the complete structure of a cognitive agent's mind (not a subnetwork).*

Axiom 1 $\forall\, n \in$ *SNodes, outdegree*$(n) > 0$ *or indegree*$(n) > 0$.

In other words, every node in SNodes has at least one arc attached to it, either incoming or outgoing; there are no isolated nodes. For a statement of this principle, see [Shapiro, 1991].

Axiom 2 *Every SNePS network $S \in$ SNets is finite, with a finite set of nodes V and a finite set of arcs E.*

Axiom 3 *Every full network $S \in$ SNets contains sensory nodes. If S contains more than one component, then every connected component of S contains sensory nodes.*

This is never explicitly stated, but seems reasonable. Science fiction and philosophy aside, a cognitive agent is not cognitive unless there is some interface to the external world.

The domain of discourse will generally be limited to SNets$'$. Given a SNePS unigraph $S' \in$ SNets$'$,

Definition 4.7.4 ***SENSORY***(S')**:** $\{$ *sensory nodes of S* $\}$

Definition 4.7.5 ***MOLATOM***(S')**:** $\{$ *molecular nodes of S' to be treated as atomic* $\}$

Definition 4.7.6 ***MOLFULL***(S')**:** $\{$ *other molecular nodes of S', which therefor have at least one child and any finite number of parents (including zero)* $\}$

Definition 4.7.7 ***BASE***(S')**:** $\{$ *base nodes of S', which therefore have no children and at least one parent* $\}$

These classes partition the set of nodes in S'. Note that BASE(S) = BASE(S') = BASE(S^*). If a full SNePS network C is under consideration, then MOLATOM(C') will be empty and MOLFULL(C') will contain all molecular nodes. When the context is clear, the unigraph argument will be omitted; the previous consequent could be written "MOLATOM will be empty and MOLFULL will contain all molecular nodes".

The first step in the formal application of the Solution Lemma to SNePS is a *tag* function for nodes (in some given S') with outdegree zero. Each sensory node $s \in$ SENSORY should be assigned its lexeme or other sensory datum, and each "atomic" molecular node $a \in$ MOLATOM its label. In other words, we assume that a given SNePS network unigraph $S' \in$ SNets$'$ comes with its own set of sensory data $\mathcal{SA}$, and, unless S is a full network, its own set of molecular node labels $\mathcal{LA}$, that together will constitute the atoms.

Definition 4.7.8

$$\begin{aligned} \textit{sensetag} &: \textit{SENSORY}(S') \longrightarrow \mathcal{SA} \\ \textit{labeltag} &: \textit{MOLATOM}(S') \longrightarrow \mathcal{LA} \\ \textit{tag} &= \textit{sensetag} \cup \textit{labeltag} \end{aligned}$$

Next we need to decorate S' by supplying a set for each node. Closely following the development given in [Barwise and Etchemendy, 1987, pages 39–40], we do so respecting its given graph-theoretic structure, *except* that we decorate base nodes with their parent nodes' sets. We are decorating S' as if it were S^*, of course, but maintaining as much rigor as possible. The decoration is the function $\mathcal{D}$, defined for arbitrary nodes $s \in$ SENSORY, $a \in$ MOLATOM, $m \in$ MOLFULL, and $b \in$ BASE.

Definition 4.7.9

$$\begin{aligned} \mathcal{D}(s) &= tag(s) \\ \mathcal{D}(a) &= tag(a) \\ \mathcal{D}(m) &= \{\mathcal{D}(c) \,|\, S' \textit{ has an arc from } m \textit{ to } c\} \\ \mathcal{D}(b) &= \{\mathcal{D}(m) \,|\, S' \textit{ has an arc from } m \textit{ to } b\} \end{aligned}$$

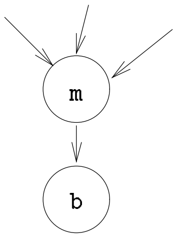


Figure 4.9: Invalid SNePS structure by Axiom 4

The result is that every node in S' forms the point of an accessible pointed graph, which has, by the Anti-Foundation Axiom, a *unique* decoration. A semantic function μ based on the decoration $\mathcal{D}$ will then allow derivation of the semantics of new nodes, "indeterminates", as hypersets over the atoms $\mathcal{SA} \cup \mathcal{LA}$ already in S'. The definition of μ has a compositional flavor, as usual for a semantic function. No semantics is assigned to nodes from SENSORY or MOLATOM, on the principle that they are best regarded only as sources of input (either actual or potential), not as meaningful in their own right.

Definition 4.7.10

$$\begin{aligned} \mu(m) &= \mathcal{D}(m) \\ \mu(b) &= f(b), \textit{where } f \textit{ is the solution to this system of equations:} \\ & \qquad\qquad b = \{\ \mathcal{D}(m) \mid b \in \mathcal{D}(m)\} \\ & \qquad\qquad m_i = \{\ \mathcal{D}(m_i)\},\ \forall m_i \ \ \textit{such that } b \in \mathcal{D}(m_i) \end{aligned}$$

Under this definition, the semantic value of a base node b is influenced by its parent nodes, as well as vice versa, as desired. The result is a semantic function μ, given a SNePS network $S \in$ SNets, and its subsequent transformation into the unigraph S'.

$$\mu : \text{MOLFULL}(S') \cup \text{BASE}(S') \longrightarrow \mathcal{V}_{\mathcal{SA} \cup \mathcal{LA}}$$

There is no claim that μ is surjective. The class of hypersets $\mathcal{V}_{\mathcal{SA} \cup \mathcal{LA}}$ may (and, in fact, will) be a proper superset of the class of hypersets assigned to a node by μ.

Before this μ can be explored, in the succeeding sections, one more axiom is necessary, more of an assumption about SNePS than the others in that it is supported by examples of SNePS networks, but not by any formal or informal statement.

Axiom 4 *Given a network $S \in$ SNets, every molecular node $m \in$ MOLFULL(S) heads a semipath $\langle m, n_1\rangle, \langle n_1, n_2\rangle, \ldots \langle n_k, t\rangle$, where $t \in$ SENSORY(S) and for every arc $\langle n_i, n_{i+1}\rangle$, either (1) there is an arc from n_i to n_{i+1} in S, or (2) $n_i \in$ BASE(S) and there is an arc from n_{i+1} to n_i in S.*

We already know that every molecular node is the parent of *something*, but it would be nice to know that, construed as a hereditary set, it contains atoms. What this axiom rules out is the sort of structure shown in Figure 4.9, where the molecular node m has no other children than base nodes that have no other parents than m. This axiom entails the previous Axiom 3, but is worthy of separate statement for discussion purposes. Its significance is that SNePS contains no conflatable concepts; there are no base nodes that are nothing more than "restatements" of dominating nodes.

As final preparation for results concerning μ, here is an immediate consequence of Axiom 4:

Lemma 1 *No node $n \in$ MOLFULL(S') $\cup$ BASE(S') $\cup$ MOLATOM(S') $\cup$ SENSORY(S') can be decorated with Ω; $\forall n \in S,\ \mu(n) \neq \Omega$.*

Proof: Ω can only decorate points of accessible pointed graphs in which every node has a child [Aczel, 1988, page 7]. Since, by Axiom 4, every $m \in$ MOLFULL(S) heads a path to sensory nodes (which have no children) that entails set inclusion under the decoration $\mathcal{D}$, then $m \in$ MOLFULL(S') cannot be decorated with Ω, and consequently $\mu(m) \neq \Omega$. Since every $b \in$ BASE(S') contains its parent molecular nodes in the decoration—as shown explicitly in S^*—and those parent nodes contain sensory nodes which have no children, then neither can $\mu(b)$ be Ω. Nodes in MOLATOM(S') and SENSORY(S') have no children by definition, so they cannot be decorated with Ω, and therefore will not be assigned Ω under μ. □

4.8 Meaning is Determined by the SNePS Network

4.8.1 SNePS networks as accessible pointed graphs

Recall from §2.5 that Shapiro develops a definition of *domination* of one node by another that is analogous to hereditary set definition. The notion of domination abstracts away multiple arcs between a parent node and a child node, as does the notion of a unigraph, a correspondence formalized by the first theorem.

Theorem 2 (Node-Picture Principle) *For all molecular nodes $m \in$ SNodes,*

$$\mu(m) = \{\mu(c) \mid m \textit{ dominates } c\}$$

Proof: By the definition of μ and $\mathcal{D}$, $\mu(m) = \{\mathcal{D}(c) \mid S' \text{ has an arc from } m \text{ to } c\}$. By the definition of "domination" in SNePS (see §2.5), this set is the same as $\{\mu(c) \mid m \text{ dominates } c\}$. □

The point of Theorem 2 is to show that the meaning of a molecular node is captured by the subnetwork rooted at that node. The semantics of a network is reflected in its construction; what you see is what you get—as long as what you see is S^*. The apg rooted at a node $n \in S^*$ is $\mu(n)$.

4.8.2 Unique μ-semantics

A qualified analog to the Uniqueness Principle, as discussed in §2.7, holds for the semantic function μ. Nodes with different meanings under μ are different nodes and therefore represent different concepts, with one important exception. It must be realized that the notion of "different nodes" in SNePS relies entirely on the FIND/BUILD mechanism that creates networks, which, when some new concept is presented to it, has one of two effects:

1. If the concept already exists (a node with exactly the right connections is already in the network), then the new information is added to it.

2. If such a node does not exist, it is created and assigned an unused unique identifier.

It is these identifiers that appear as labels in all network examples and as elements of μ-values through the subsets MOLATOM and SENSORY.

Theorem 3 (Uniqueness Principle under μ) *In any $\mathcal{C}' \in$ SNets$'$ derived from a full network $\mathcal{C}$, unless nodes n and m dominated exactly the same subordinate nodes (in which case the arc labels differed),*

$$n = m \textit{ if and only if } \mu(n) = \mu(m)$$

Expressed as an identity claim, this statement is also a distinction claim, and may be more illuminating in the contrapositive form, $n \neq m$ iff $\mu(n) \neq \mu(m)$. "Even in cases where there is a simple correspondence of objects, as when the numeral 3 stands for the number three, it is really the object's *being that and not some other numeral* that corresponds to the number's *being that and not some other number*." [Smith, 1987, page 11]

Proof:
(1) Assume $n = m$ in SNodes. Then, by the recursive rules for well-formed SNePS nodes SR.i, they are required to have the same identifier. This is impossible for distinct nodes under SNePS mechanisms—i.e., BUILD does not allow it—and therefore there is only one node under consideration, and only one value of μ.
(2) Assume $\mu(n) = \mu(m)$. By the Solution Lemma, this value of μ is the only one that satisfies the systems of equations defining $\mu(n)$ and $\mu(m)$, respectively. Those equations depend on the decorations of m and n, which depend on subordinate graphs that are always unique. (Again, BUILD does not allow a node to be added to a network if it is redundant in its adjacency matrix and arc labels to any other node.) The Uniqueness Principle for SNePS amounts to the requirement that the decoration $\mathcal{D}$ of the original network S' be injective; no two nodes can be decorated with the same set (unless they differed only in arc labels). □

The exceptional case in the theorem above is not a trivial one, and may easily occur in a single cognitive agent. See the statement of the problem in §7.2. Furthermore, the dropping of arc labels in the formation of $\mathcal{C}'$ has had the consequence that two networks with the same structure *except* for arc labels, i.e., the same adjacency matrix and tags for atoms, will yield the same semantics. This is unlikely to occur if there are any molecular nodes treated as atoms—it is not clear how two cognitive agents would end up with the same molecular node labels, and certainly two subnetworks of a single cognitive agent would not—but still, a crucial aspect of the meaning of the network is being neglected. The semantic function μ as developed here, therefore, can reasonably be called somewhat sterile in its regard only for connections between nodes, and an alternative, richer, semantics based on non-well-founded set principles is sketched in §7.2.

4.9 Contributions to SNePS Research

Even though the semantics of nodes and networks defined by μ via hypersets are "on the surface" (virtually given by the network itself), μ makes some contributions toward elucidating the principles of SNePS.

1. **Semantics for Base Nodes**
   A base node must *mean* something. The theory of SNePS has heretofore been able to provide semantics for molecular nodes, as in the SI.i rules of [Shapiro and Rapaport, 1987], but not for base nodes. The semantic function μ provides a complete semantics, assigning meanings, sets of relationships rooted in the sensory interface, to all SNePS nodes.

2. **Support for the Uniqueness Principle**
   Since every node in the network represents a unique concept, AFA ensures that every node in the network—because it has a unique decoration— has a unique meaning, as shown by the Uniquness Principle under μ. The universe of possible concepts is hypersets over whatever sensory atoms we consider.

3. **Static Semantics**
   As seen in the example above, we can define the meaning of a node in terms of the meaning of others. Any base or molecular node is subject to such treatment. Given a full network $C \in$ SNets fixed at a certain state in the development of the cognitive agent C, μ can be computed for any node in terms of SENSORY(C), or in terms of SENSORY(C) $\cup$ MOLATOM(C), with whatever selection of molecular nodes to be taken as atoms that suits the purpose.

4. **Dynamic Semantics**
   If a full network is under study as the cognitive agent C learns and develops, μ can express the meaning of newly-acquired nodes in terms of the new sensory atoms, if any, and the atomic

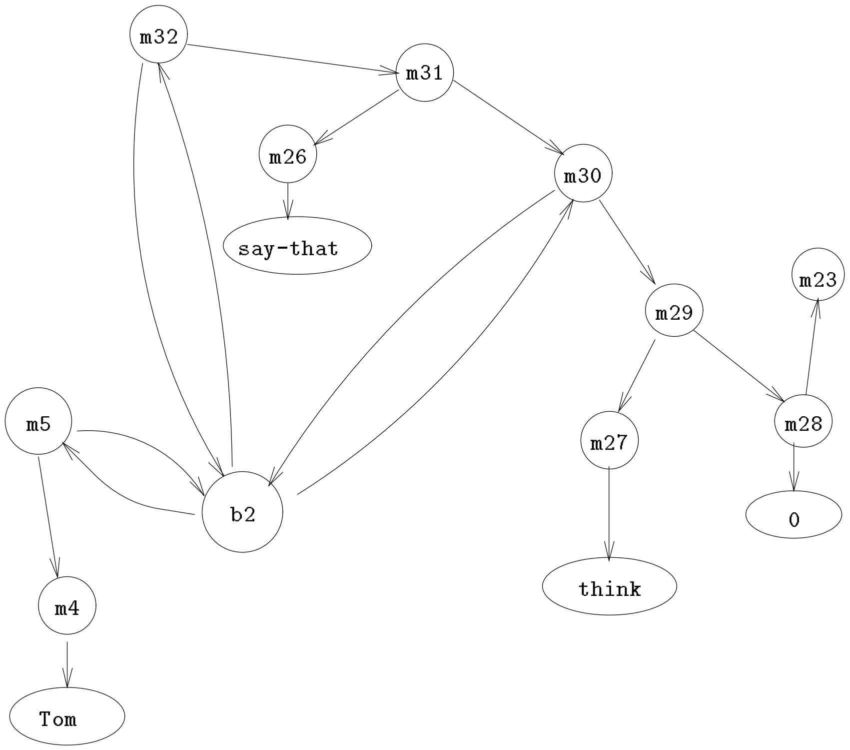
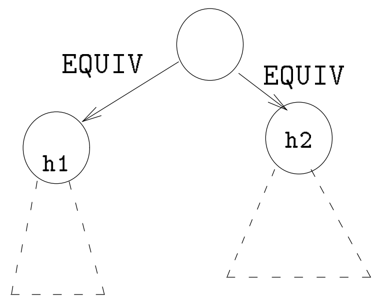


Figure 4.10: DIDN'T-THINK-SO sentence as an accessible pointed graph

nodes of the original network. In the TABLE example, as Nancy and Tom interact, new nodes are added to the network of the observer. The second sentence incorporated into the model is "Tom said he didn't think so," building the additional structure in Figure 4.10 [Rapaport, 1988a]. If one were to need, on the fly, the meaning of node `m30`, for example, in terms of the atoms SENSORY(C) $\cup$ {`think`}, the answer is computable as μ(`m30`), which would be μ(`b2`) $\cup$ μ(`m29`), the latter component μ(`m29`) a brand-new extension to the function μ and the former, μ(`b2`), changed by the incorporation of μ(`m30`). This example is revisited in §6.5.2.

5. **Intensionality**
Insofar as *intensionality* is a conceptual distinction between extensionally identical concepts, it is maintained under μ even when two nodes are established to be "the same thing" through the imposition of the **`EQUIV/EQUIV`** case frame. For example, consider cognitive agent A, who learns suddenly that the English King Henry who wrote those lovely madrigals (node `h1`) is the same historical personage as the English King Henry who callously disposed of his wives and enemies (node `h2`). The situation is depicted thus (although it is likely that `h1` and `h2` would be base nodes and therefore dominate no subgraphs):



Since the two nodes are distinct, $\mu(\texttt{h1}) \neq \mu(\texttt{h2})$, although each will exercise a high degree of influence on the other through components of their meanings that they already shared, such as the concepts of *olden-times*, *England*, and *what-fabric-was-used-to-make-a-ruff?*. In some sense, this makes them *physically* close. It may be a comfort to A to know that these two concepts do not have the same meaning—but their semantic contribution under μ to far-flung nodes will be virtually indistinguishable. In the actual computation of semantics for nodes in some given derived subnetwork S' that encloses two nodes, the smaller the given S', the more different the two nodes, in some relative sense. The more restricted the context, the less they have in common (simply because there are fewer nodes around to be built in to their meanings). This seems intuitively correct: the more focused on stars and planets is the current cognitive activity, the sharper the distinction between the Morning Star and Evening Star.

For every node n and distance d, there is a $\mu(n)$, by the Solution Lemma. All nodes have μ-semantics no matter how restricted the context, in other words, even if no sensory nodes are included. Support is thereby lent to the principle of intensionality that concepts are meaningful without regard to extensionality or other relationships to the external world.

6. **Internal Semantics**
   The semantics is as internal as it could get: The meaning of a node *is other nodes*, rather than some property of the concept itself—although the properties (as perceived by the agent) should eventually be incorporated through sensory nodes. In fact, the meaning of a node in a network is nothing more nor less than the subnetwork around it—which would be the naive interpretation anyway.

7. **Meaning is Location**
   The meaning of a node is highly dependent on its location within the surrounding network. Every arc, every connection, matters, since it adds an element to the decoration and thence a component to the semantics. Yet the meanings of two adjacent nodes (connected by an arc) are not exactly the same and cannot be the same, since the network definitions build in the structural distinction required. The next chapter investigates computation of μ under different measured scopes and the resulting manifestations of distance between nodes.

8. **Explanation of Degree of Compositionality**
   Definition of SNePS semantics in terms of non-well-founded sets allows discussion of compositionality in formal terms (see Chapter 6).

9. **Semantics of a Cognitive Agent** The semantics of a whole network—that is, the entire contents of a cognitive agent's "mind"—can be defined as the union of the semantics of the point basis (see §2.4), since those nodes dominate all others.

$$\mu(\mathcal{C}') = \cup \{ \mu(m) \ : \ m \in \text{ point basis of } \mathcal{C}' \}$$

For example, if the network shown in Figure 2.1 were the entire cognitive contents of some (impoverished) agent $\mathcal{C}$, then

$$\mu(\mathcal{C}') = \{\mu(\texttt{m2}), \mu(\texttt{m7}), \mu(\texttt{m8})\}$$

In fact, $\mathcal{C}^*$ would be made up of the three apgs whose points are `m2`, `m7`, and `m8`; all other nodes would be points of apgs embedded in one or more of those. So cognitive agent $\mathcal{C}$ would be fairly described as one who has no knowledge or beliefs beyond a a few extremely limited concepts concerning John's attitude toward the girl next door.

10. **Distinction between SNePS and other intensional semantic nets**
The system KL-ONE, while sharing with SNePS much of its philosophy of intensional representation, does not allow mutual influence between nodes, or any other hint of circularity. In a discussion of example Concepts, in which *MAMMAL* is a superConcept of *HUMAN*, Brachman and Schmolze state:

> Finally, it is important to reiterate that a Concept like *MAMMAL* does not derive any of its meaning from the Concept *HUMAN*. A Concept's meaning is strictly determined by its subsuming Concepts plus the information associated specifically with the Concept. [Brachman and Schmolze, 1985, page 181]

To the developers of KL-ONE, of course, and to the conventional wisdom of knowledge representation, strict one-way semantic influence is a virtue. The foregoing chapters are meant to question that, to demonstrate that not only is semantic circularity *desirable*, but *achievable*, and in a disciplined manner.

Many of the points made above are further discussed and refined in the next chapter, which proposes a "measured" μ.

Chapter 5

Limited Circular Semantics

5.1 Introduction

In this chapter, a qualified version of the general semantic function μ is introduced, which will provide for recursive computation of the semantics of SNePS nodes within given contexts, and allow algorithmic analysis.

5.2 Examples of Restricted Semantics

Recall the example SNePS network in which John believes that the girl next door is sweet, as shown in Figure 2.1. Asked to provide meanings for, say, node `m6` to increasing levels of elaboration, one might come up with the straightforward approach given in Figure 5.1, indicating the "distance" or "degree of elaboration" with a superscript δ attached to μ. These are not subnetworks of C *per se*, but the actual hypersets to be assigned to `m6` under μ^δ; node labels are included for convenience.

And what should be the semantic values of the base node `b2`, as the SCOPE broadens? These are shown in Figure 5.2.

We have already seen a SCOPE operator used to define subnetworks, of course, in §2.8.3, defined by counting off semipaths of a certain length from a given node n. Given a full SNePS network C and some node n, $\mu^\delta(n)$ will be close to, but not quite the same as, first marking off the subnetwork of C (actually, its unigraph form C'; see §2.4) from n with SCOPE(n, δ), and then applying μ to that node n in the resulting subnetwork. Note that if we were to compute $\mu^1(\mathtt{b2})$ as $\mu(\mathtt{b2})$ in SCOPE $(\mathtt{b2}, 1)$, it would be Ω, since the definition of μ would, via the decoration, include both `m6` and `m4` in $\mu(\mathtt{b2})$, as shown in Figure 5.3. The defined subnetwork SCOPE(n, d) also shows nodes whose meanings are *influenced by* n, not just those whose meanings *influence* n. The restricted μ^δ, on the other hand, will be defined to provide the semantics shown in Figures 5.1 and 5.2.

5.3 A Qualified Semantic Function μ^δ

Again, we assume that we are given a full SNePS unigraph C', with nodes partitioned into the classes SENSORY, MOLATOM, MOLFULL, and BASE, and *sensetag* and *labeltag* functions for nodes in SENSORY and MOLATOM. We need identifiers for *all* nodes in C', however, since the scope δ may leave any node unelaborated, rendering it atomic, so we simply assume a given set of node labels, $\mathcal{NA}$, and a function

$$tag \; : \; \text{NODES}(C') \longrightarrow \mathcal{NA}$$

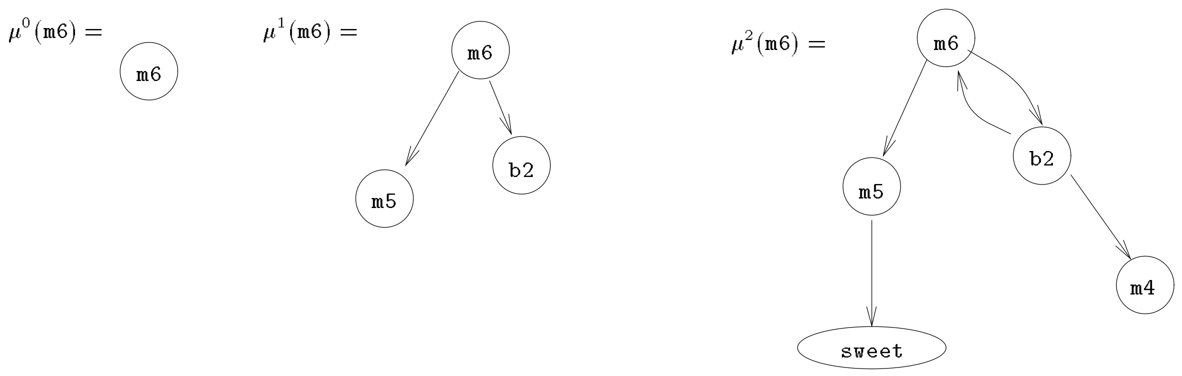


Figure 5.1: Suggestion for qualified μ^δ applied to `m6`, for $\delta = 0, 1,$ and 2

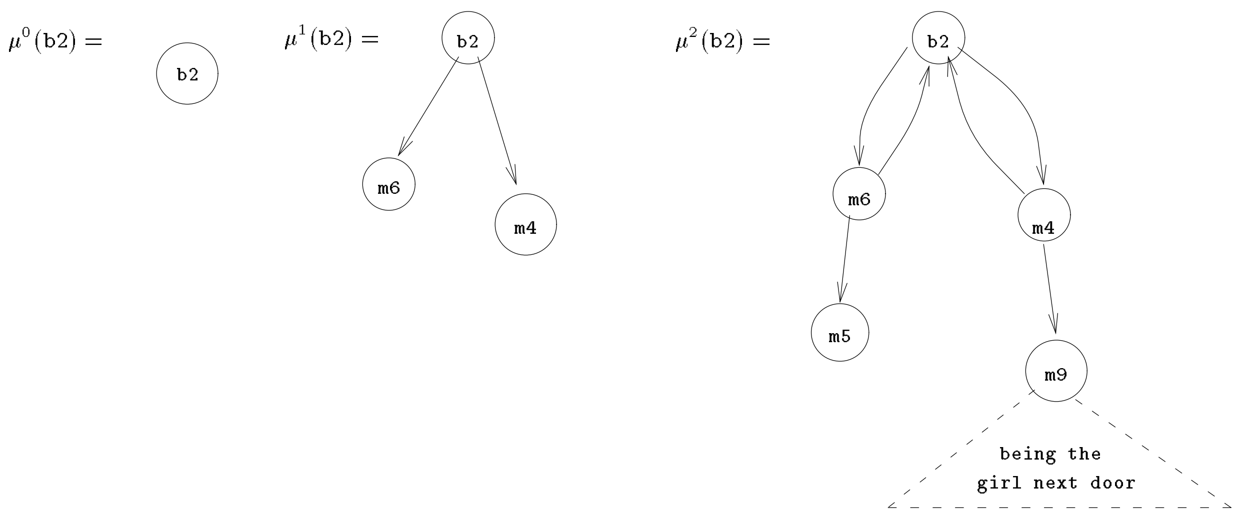


Figure 5.2: Suggestion for qualified μ^δ applied to `b2`, for $\delta = 0, 1,$ and 2

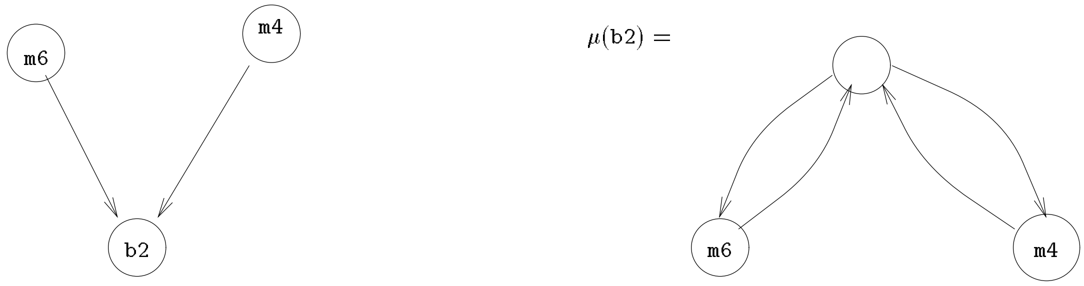


Figure 5.3: SCOPE($\texttt{b2}, 1$) and the resulting $\mu(\texttt{b2})$

Our aim is the definition of $\mu^\delta(n)$ for an arbitrary node n that reflects all and only those semantic influences within a certain "distance" δ of n, as illustrated in Figures 5.1 and 5.2,

The appropriate definition will again rely heavily on the decoration, now qualified by the superscript δ and defined inductively, attached to a node n.

Definition 5.3.1

$$\text{Base Case : } \mathcal{D}^0(n) \quad = \quad tag(n)$$

$$\text{For } \delta \geq 1 : \mathcal{D}^\delta(n) \quad = \quad \begin{cases} \{\mathcal{D}^{\delta-1}(c) \mid C' \textit{ has an arc from } n \textit{ to } c\}, & \textit{if } n \in \text{MOLFULL}(C') \\ \{\mathcal{D}^{\delta-1}(m) \mid C' \textit{ has an arc from } m \textit{ to } n\}, & \textit{if } n \in \text{BASE}(C') \\ tag(n), & \textit{otherwise.} \end{cases}$$

As an example, we decorate `m20` from Figure 4.2 to $\delta = 3$ according to the definition, paying particular attention to the treatment of the semantic cycle between `m20` and the base node `z`. The label or tag is sometimes used to denote the node itself, as in the use of expressions like $\mathcal{D}^0(\texttt{exert})$.

$$\begin{aligned} \mathcal{D}^3(\texttt{m20}) &= \{\mathcal{D}^2(\texttt{m19}), \mathcal{D}^2(\texttt{z})\} \\ &= \{\{\mathcal{D}^1(\texttt{m10}), \mathcal{D}^1(\texttt{w})\}, \{\mathcal{D}^1(\texttt{m20}), \mathcal{D}^1(\texttt{m22}), \mathcal{D}^1(\texttt{m17})\}\} \\ &= \{\{\{\mathcal{D}^0(\texttt{exert})\}, \{\mathcal{D}^0(\texttt{m19}), \mathcal{D}^0(\texttt{m18}), \mathcal{D}^0(\texttt{m21})\}\}, \\ &\qquad \{\{\mathcal{D}^0(\texttt{m19}), \mathcal{D}^0(\texttt{z})\}, \{\mathcal{D}^0(\texttt{z}), \mathcal{D}^0(\texttt{m17})\}, \{\mathcal{D}^0(\texttt{z}), \mathcal{D}^0(\texttt{m15})\}\}\} \\ &= \{\{\{\texttt{exert}\}, \{\texttt{m19}, \texttt{m18}, \texttt{m21}\}\}, \{\{\texttt{m19}, \texttt{z}\}, \{\texttt{z}, \texttt{m17}\}, \{\texttt{z}, \texttt{m15}\}\}\} \end{aligned}$$

The decoration can be viewed as the "unfolding" (Aczel's term [Aczel, 1988, page 5]) of the set at the point, `m20`, into a *tree* apg of height δ, as shown in Figure 5.4. Various degrees of conflation of indentical nodes yield other pictures (including, with unique nodes for every set, that is, an injective decoration, the subnetwork of Figure 4.2 that is the source itself). One other picture that will be of interest shortly is the apg shown in Figure 5.5, where each *atom* has a unique node, that is, the *tag* function is injective. Cycles appear as repeated sequences of nodes along paths. This unfolding can be seen as a recursive algorithm for computing the decoration of a node to some depth δ.

Since all SNePS nets are finite by Axiom 2, the unqualified decoration $\mathcal{D}(n)$ of a node $n \in$ SNodes (the topic of Chapter 4) can be defined as $\mathcal{D}^x(n)$, where x is large enough to reach all relevant nodes from n.

The definition of the qualified μ is different only in its use of the qualified decoration.

Definition 5.3.2

$$\mu^\delta(m) \quad = \quad \mathcal{D}^\delta(m)$$

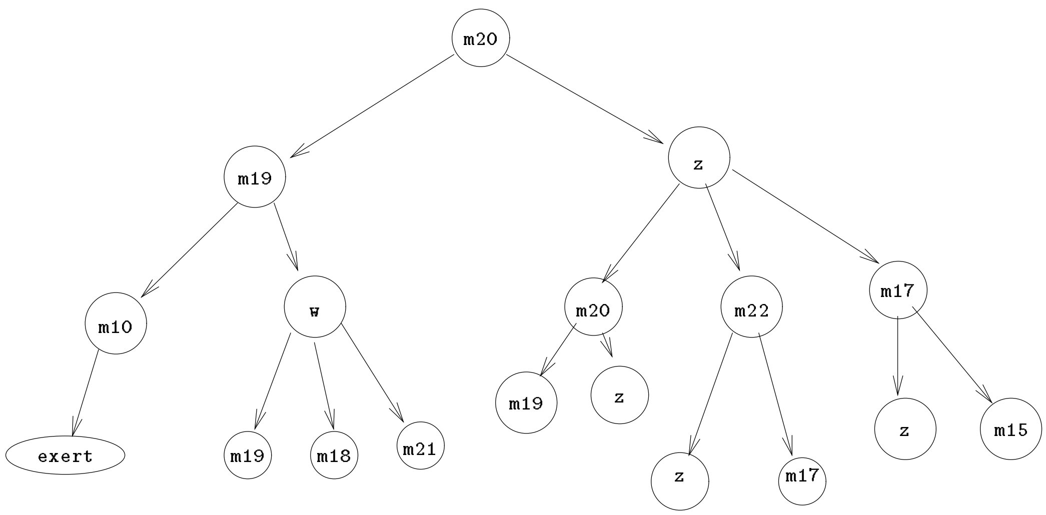


Figure 5.4: A tree picture of `m20`, to a depth of 3

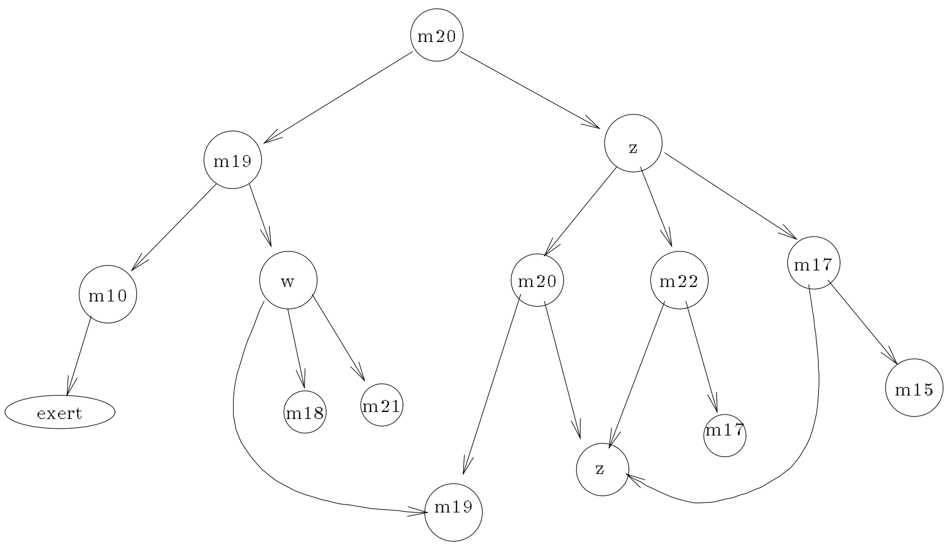


Figure 5.5: A graph picture of `m20` (with injective *tag* function), to a depth of 3

$$\mu^\delta(b) \quad = \quad f(b), \textit{ where } f \textit{ is the solution to this system of equations:}$$
$$b = \mathcal{D}^\delta(b)$$
$$m_i = \{\, \mathcal{D}^{\delta-1}(m_i)\}, \qquad \forall m_i \;\; \textit{such that } b \in \mathcal{D}^{\delta-1}(m_i)$$

As an example that can be compared to the result for the unqualified μ, let us compute $\mu^3(\texttt{b2})$, as was done in §4.6.2. We will take the full network to be W' again, with $\texttt{m23} \in$ MOLATOM, (Figure 4.4), and should get the same hyperset assigned to b2 as derived there and shown in Figure 4.8, since $\delta = 3$ is sufficiently large to involve all of the structure used there.

Our set of atoms is therefore

$$\mathcal{A} = \mathcal{SA} \cup \mathcal{LA} = \{\texttt{m23}, \texttt{ask-whether}, \texttt{Tom}\}$$

First, the decoration:

$$\begin{aligned}
\mathcal{D}^3(\texttt{b2}) &= \{\mathcal{D}^2(\texttt{m24}), \mathcal{D}^2(\texttt{m5})\} \\
&= \{\{\mathcal{D}^1(\texttt{m23}), \mathcal{D}^1(\texttt{m3}), \mathcal{D}^1(\texttt{b2})\}, \{\mathcal{D}^1(\texttt{b2}), \mathcal{D}^1(\texttt{m4})\}\} \\
&= \{\{tag(\texttt{m23}), \{\mathcal{D}^0(\texttt{ask-whether}), \}, \{\mathcal{D}^0(\texttt{m24}), \mathcal{D}^0(\texttt{m5})\}\}, \{\{\mathcal{D}^0(\texttt{m24}), \mathcal{D}^0(\texttt{m5})\}, \{\mathcal{D}^0(\texttt{Tom})\}\}\} \\
&= \{\{\texttt{m23}, \{tag(\texttt{ask-whether}), \}, \{tag(\texttt{m24}), tag(\texttt{m5})\}\}, \{\{tag(\texttt{m24}), tag(\texttt{m5})\}, \{tag(\texttt{Tom})\}\}\} \\
&= \{\{\texttt{m23}, \{\texttt{ask-whether}\}, \{\texttt{m24}, \texttt{m5}\}\}, \{\{\texttt{m24}, \texttt{m5}\}, \{\texttt{Tom}\}\}\}
\end{aligned}$$

By the definition, $\mu^3(\texttt{b2})$ is derived thus:

$$\mu^3(\texttt{b2}) = f(b), \text{where } f \text{ is the solution to this system of equations:}$$
$$\begin{aligned}
\texttt{b2} &= \mathcal{D}^3(\texttt{b2}) &&= \{\{\texttt{m23}, \{\texttt{ask-whether}\}, \{\texttt{m24}, \texttt{m5}\}\}, \{\{\texttt{m24}, \texttt{m5}\}, \{\texttt{Tom}\}\}\} \\
\texttt{m24} &= \{\, \mathcal{D}^2(\texttt{m24})\} &&= \{\texttt{m23}, \{\texttt{ask-whether}\}, \{\texttt{m24}, \texttt{m5}\}\} \\
\texttt{m5} &= \{\, \mathcal{D}^2(\texttt{m5})\} &&= \{\{\texttt{m24}, \texttt{m5}\}, \{\texttt{Tom}\}\}
\end{aligned}$$

So the set of unknowns is $\mathcal{X} = \{\texttt{b2}, \texttt{m24}, \texttt{m5}\}$. The steps given so far should be familiar, as they parallel the development in Chapter 4. Next is the development of the solution f and its verification.

As in [Barwise and Etchemendy, 1987, pages 51–52], we will methodically construct the solution from the apgs given directly by the equations, as shown in Figures 5.6, 5.7, and 5.8, and then combine them.

Before continuing, note that, under non-well-founded set theory, any way of coming up with a solution is as good as any other. As long as verfication can be done, the Solution Lemma (or, alternatively, the Anti-Foundation Axiom) ensures that we have the "right" hyperset assignments. Any hypersets that work are the right ones. Those that work, of course, are those that maintain the relationships among the unknowns given by the system of equations *after* the assignments to the unknowns have been substituted for the unknowns. In the current case, the goal is a function, any function, $f : \mathcal{X} \longrightarrow \mathcal{V}_\mathcal{A}$ such that:

$$f(\texttt{b2}) = \{\{\texttt{m23}, \{\texttt{ask-whether}\}, \{f(\texttt{m24}), f(\texttt{m5})\}\}, \{\{f(\texttt{m24}), f(\texttt{m5})\}, \{\texttt{Tom}\}\}\} \tag{5.1}$$
$$f(\texttt{m24}) = \{\texttt{m23}, \{\texttt{ask-whether}\}, \{f(\texttt{m24}), f(\texttt{m5})\}\} \tag{5.2}$$
$$f(\texttt{m5}) = \{\{f(\texttt{m24}), f(\texttt{m5})\}, \{\texttt{Tom}\}\} \tag{5.3}$$

In this chapter, rather than settling for "any solution f that works", we are attempting to make the process algorithmic, through careful development of a particular constructible solution. It consists of putting together the apgs for the hypersets over $\mathcal{A} \cup \mathcal{X}$ in such a way that an appropriate assignment to each unknown results, and doing so by following the method employed (or at least implied) by [Barwise and Etchemendy, 1987]. After deriving the apg for a hyperset $G_\mathbf{x}$ on the right-hand side of the equations defining an unknown $\mathbf{x}$, "we simply alter these graphs by replacing all edges terminating in a node tagged with $\mathbf{x}$ by an edge terminating in the top node of $G_\mathbf{x}$ ..."

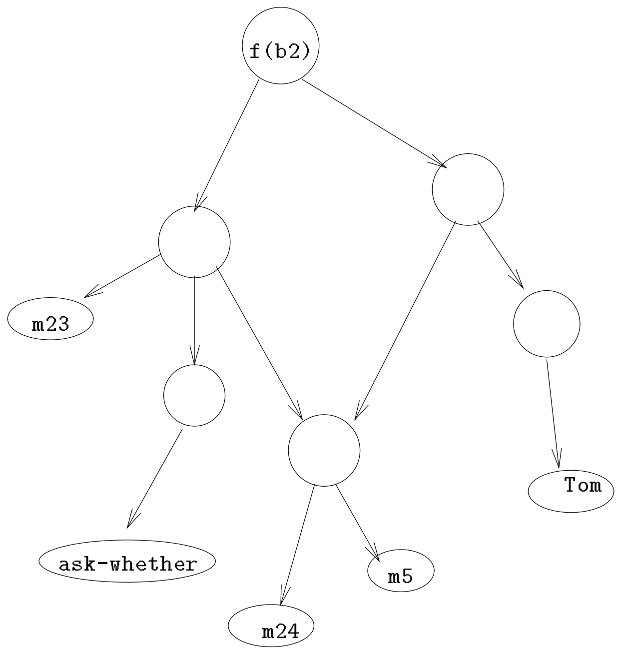


Figure 5.6: Proposed solution; assignment to `b2`

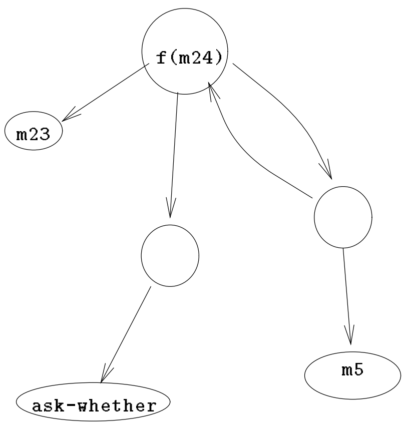


Figure 5.7: f(`m24`)

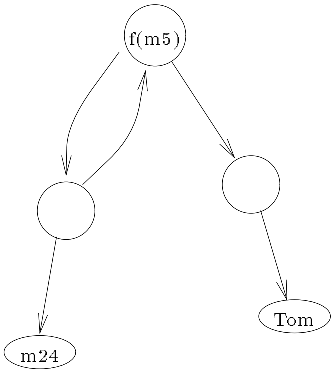


Figure 5.8: f(`m5`)

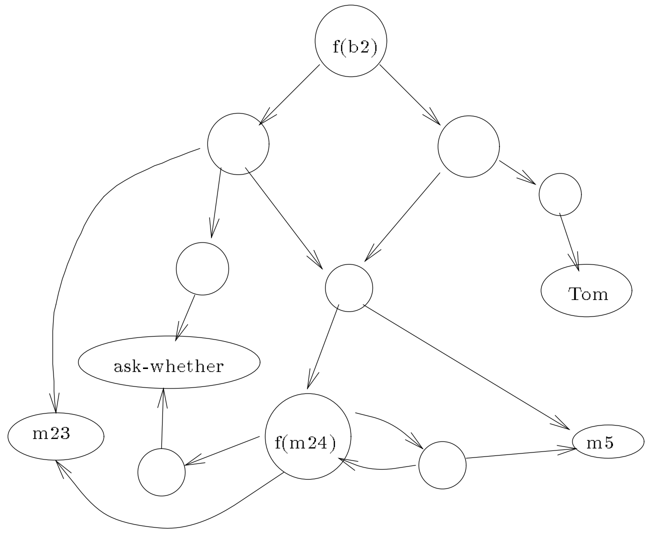


Figure 5.9: Solution under construction; m24 has been replaced by f(m24)

[Barwise and Etchemendy, 1987, page 51]. There is mutual membership in the three apgs derived as solutions—$f(\texttt{m24})$ includes m5, and $f(\texttt{m5})$ includes m24, and $f(\texttt{b2})$ includes both—but we use the apg $f(\texttt{b2})$ as the "bed" for the solution because its semantics is the object of this exercise. We replace each b2 in Equations (or hypersets) 5.2 and 5.3 with the hyperset given by Equation 5.1, and we replace each m24 in Equations 5.1 and 5.3 with the right-hand side of Equation 5.2, and similarly for m5. There are no occurrences of b2 to replace (although with a different δ, there would have been), but let us put the hypserset for m24 in place of the node m24 in Figure 5.6 in the most simple-minded way possible, making the out-arcs to atoms connect to the nodes already there, as shown in Figure 5.9, and then build in the hyperset for m5 in the same way, as shown in Figure 5.10. (Critical node labels have been retained for convenience, but the only officially labeled nodes are atoms in $\mathcal{A}$.) The result is one diagram that contains the assignments to all three unknowns, each obtained by making a different labeled node the point of an apg, and each, as required, a hyperset over the atoms $\mathcal{A} = \{\texttt{m23}, \texttt{ask-whether}, \texttt{Tom}\}$.

These are not the simplest pictures of the hypersets, but they were simple to derive algorithmically. Let's see if verification can be done. The assignments that result follow in written form, taken from the apg in Figure 5.10. (We are forgoing the use of intermediate names x, etc., for the hyperset assignments, as was practiced in Chapter 4, referring to them simply as $f(\texttt{b2})$, etc.)

$$
\begin{aligned}
f(\texttt{b2}) &= \{\{\texttt{m23}, \{\texttt{ask-whether}\}, \{f(\texttt{m24}), f(\texttt{m5})\}\}, \{\{f(\texttt{m24}), f(\texttt{m5})\}, \{\texttt{Tom}\}\}\} \\
f(\texttt{m24}) &= \{\texttt{m23}, \{\texttt{ask-whether}\}, \{f(\texttt{m24}), f(\texttt{m5})\}\} \\
f(\texttt{m5}) &= \{\{f(\texttt{m24}), f(\texttt{m5})\}, \{\texttt{Tom}\}\}
\end{aligned}
$$

The equations above provide immediate verification of Equations 5.1, 5.2, and 5.3, so the assignments depicted in Figure 5.10 work. The construction made it so, and in fact, guarantees that the derived solution apgs will be pictures of the correct hypersets, as discussed in the next section.

Figure 5.10 is not a subgraph of any S^*. It could not be a SNePS structure, and, fact, it violates Theorm 3. The decoration is not injective, and so the BUILD mechanism could not have created it. For example, there are easily seen to be two nodes that would be decorated with $\{\texttt{Tom}\}$.

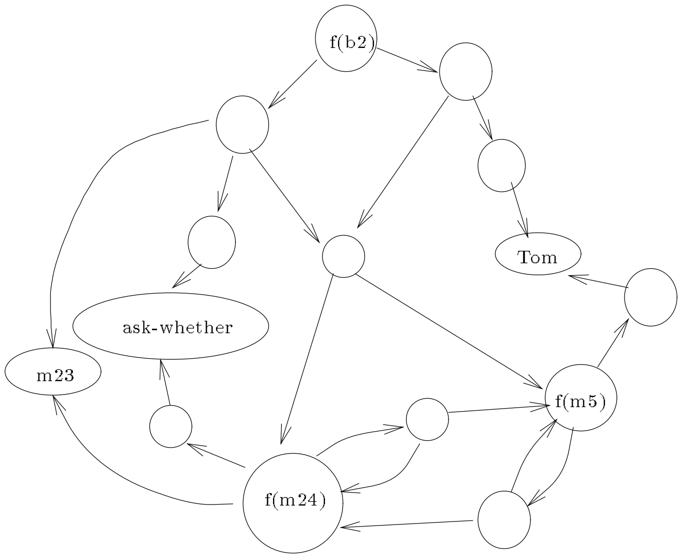


Figure 5.10: Completed solution G^3_{b2}; $\mathtt{m5}$ has been replaced by f($\mathtt{m5}$)

5.4 Hyperset Equivalence

Simpler pictures of the solution depicted in Figure 5.10 also work. Consider the picture of Figure 4.8, replicated in Figure 5.11, suggesting the solution below.

$$f(\mathtt{b2}) = \{f(\mathtt{m24}), f(\mathtt{m5})\} \tag{5.4}$$
$$f(\mathtt{m24}) = \{\mathtt{m23}, \{\mathtt{ask-whether}\}, f(\mathtt{b2})\} \tag{5.5}$$
$$f(\mathtt{m5}) = \{f(\mathtt{b2}), \{\mathtt{Tom}\}\} \tag{5.6}$$

Can verification be done?

$$\begin{array}{rcll}
f(\mathtt{b2}) & = & \{f(\mathtt{m24}), f(\mathtt{m5})\} & \text{by 5.4} \\
 & = & \{\{\mathtt{m23}, \{\mathtt{ask-whether}\}, f(\mathtt{b2})\}, \{f(\mathtt{b2}), \{\mathtt{Tom}\}\}\} & \text{by 5.5, 5.6} \\
 & = & \{\{\mathtt{m23}, \{\mathtt{ask-whether}\}, \{f(\mathtt{m24}), f(\mathtt{m5})\}\}, \{\{f(\mathtt{m24}), f(\mathtt{m5})\}, \{\mathtt{Tom}\}\}\} & \text{by 5.4} \\
 & QED & \text{Equation 5.1 is verified.} & \\
f(\mathtt{m24}) & = & \{\mathtt{m23}, \{\mathtt{ask-whether}\}, f(\mathtt{b2})\} & \text{by 5.5} \\
 & = & \{\mathtt{m23}, \{\mathtt{ask-whether}\}, \{f(\mathtt{m24}), f(\mathtt{m5})\}\} & \text{by 5.4} \\
 & QED & \text{Equation 5.2 is verified.} & \\
f(\mathtt{m5}) & = & \{f(\mathtt{b2}), \{\mathtt{Tom}\}\} & \text{by 5.6} \\
 & = & \{\{f(\mathtt{m24}), f(\mathtt{m5})\}, \{\mathtt{Tom}\}\} & \text{by 5.4} \\
 & QED & \text{Equation 5.3 is verified.} &
\end{array}$$

What this shows is that Figures 5.10 and 5.11 (the apg on the right-hand side of Figure 4.8, which was derived earlier "by inspection") are pictures of the same hyperset. Not surprisingly, there are many apgs for the same solution, the same hyperset assignments.

Let us briefly explore the equivalance problem in general, and then return to its manifestations in the computation of μ^{δ}. Like any other sets, hypersets are equivalent if they contain exactly the same members. Yet there are pitfalls in this simple extensionality criterion when circularity

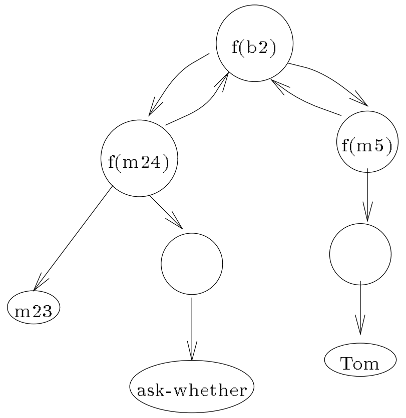


Figure 5.11: A simpler solution

is admitted. It is due to our dependence on the apg as the finite representation of the hyperset. For example, consider the four apgs G_1, G_2, G_3, G_4 in Figure 5.12. The pair G_1 and G_2 represent distinct hypersets, but the pair G_3 and G_4 are two pictures of one and the same hyperset (Ω), notwithstanding the structural similarity between G_1 and G_3 and between G_2 and G_4. Recall also the variation in pictures of Ω noted in Chapter 3, and illustrated in Figure 3.2, showing that apgs with great structural *difference* can picture the same hyperset. The correlation between graphical and representational properties of apgs is not at all obvious in general. Extracting some version of extensionality from the potentially cyclic apgs will require some care.

In terms of difficulty, two extreme cases of detection of hyperset equivalence through apg examination can be identified: (1) for apgs with no cycles and (2) for apgs with what we might call "unrestricted" branching and circularity. For (1), two tagged apgs with no cycles, equivalence of the depicted hypersets is easy to evaluate recursively (as long as the *tag* function is injective; otherwise we could have the two different pictures of the set we know as the number 3 shown at the beginning of Chapter 3). Recall that acyclic apgs are called "well-founded" and depict sets that are well-founded, for which the extensionality criterion says all there is to say about equality, and "by a transfinite induction on the membership relation the equality relation between well-founded sets is uniquely determined." [Aczel, 1988, page 19].

For the difficult case (2), including infinite apgs, those that contain Ω, and others that have no restrictions on the circularity of the decorations, Aczel's relation '$\equiv_{\mathcal{V}}$' captures the wide variation possible in apgs for the same hyperset (over some universe $\mathcal{V}$). To illustrate the problem, consider the apgs G_5 and G_6 in Figure 5.13. Both are legitimate apgs, accessible pointed graphs, since the general definition of a graph allows for infinite sets of nodes V and arcs E and since every node is accessible from the point. Both are pictures of the same hyperset, Ω, like apgs G_3 and G_4 in Figure 5.12. But there is no clean way to apply the extensionality criterion to demonstrate that that is so, no induction from a base case. To handle these problems, the definition of the relation $\equiv_{\mathcal{V}}$, in very crude terms, is as *permissive* as possible; two hypersets are considered equal *unless* proven unequal by production of a member of one that is not in the other. It is clear that in all four apgs, G_3, G_4, G_5, and G_6, production of such a member is impossible; never mind that it is also impossible to produce a member of one that is clearly *in* any of the others.

Aczel's definition of $\equiv_{\mathcal{V}}$ requires the notion of "bisimulation". Its definition is not inductive; it has no "basis". [Maurer and Ralston, 1991, page 185]. (Recall that the notation "$\forall x | a \rightarrow x$" means "all children x of a in the given apg of which a is the point".) A binary relation R is a *bisimulation*

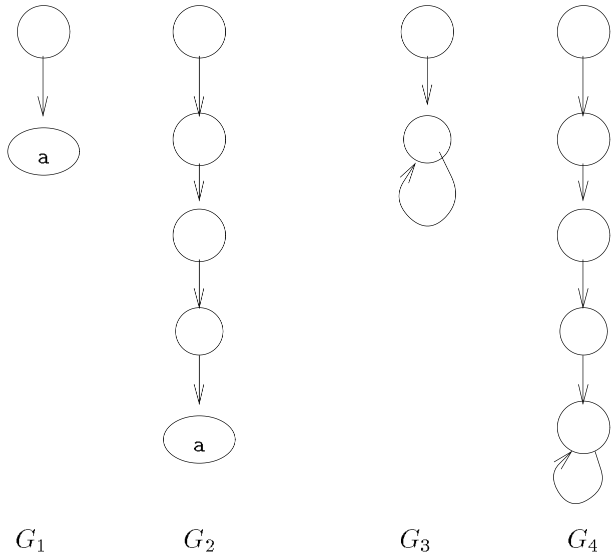


Figure 5.12: Four pairwise similar apgs, representing three distinct hypersets.

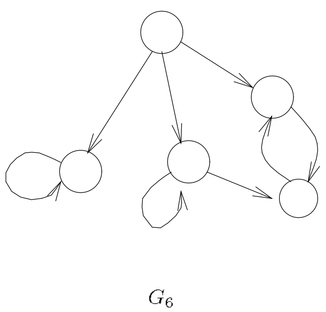


Figure 5.13: Yet more pictures of the hyperset Ω

if and only if $aRb \Longrightarrow aR^+b$, where [Aczel, 1988, page 20]:

$$aR^+b \iff (\forall x|a \to x)(\exists y|b \to y)\ xRy\ \&\ (\forall y|b \to y)(\exists x|a \to x)\ xRy$$

Given any universe $\mathcal{V}$ of hypersets, the relation $\equiv_{\mathcal{V}}$ on its members is the *unique maximum bisimulation.* That is, by Aczel's Theorem 2.4 [ibid.], there is one relation $\equiv_{\mathcal{V}}$ on $\mathcal{V}$ such that:

1. $\equiv_{\mathcal{V}}$ is a bisimulation.
2. If R is a bisimulation on $\mathcal{V}$, then $\forall a, b \in \mathcal{V}, aRb \Longrightarrow a \equiv_{\mathcal{V}} b$.

In fact, $\equiv_{\mathcal{V}}$ is the greatest fixed point of the operator $()^+$ that creates, from a relation R, the relation aR^+b. It is a fixed point because $(\equiv_{\mathcal{V}})^+$ is just $\equiv_{\mathcal{V}}$, since $\langle a,b\rangle \in (\equiv_{\mathcal{V}})^+ \Longrightarrow \langle a,b\rangle \in\ \equiv_{\mathcal{V}}$ and $\langle a,b\rangle \in\ \equiv_{\mathcal{V}} \Longrightarrow \langle a,b\rangle \in (\equiv_{\mathcal{V}})^+$. It is the greatest fixed point because for any other relation Z that is a fixed point (i.e., $Z^+ = Z$), $\langle a,b\rangle \in Z \Longrightarrow$ Z is a bisimulation $\Longrightarrow \langle a,b\rangle \in\ \equiv_{\mathcal{V}}$. Aczel then shows that $a \equiv_{\mathcal{V}} b$ if and only if there is an apg that is a picture of both a and b [Aczel, 1988, page 21, Proposition 2.5]. This gives another way of stating the "permissiveness" of hyperset equivalence through apgs: Two hypersets are equivalent if they have *any* picture in common; never mind demonstrating that *all* pictures share some property. These considerations should reveal how difficult it can be to detect hyperset equivalence from apgs in the broadest case. Fortunately, the case of apgs drawn from SNets* will prove less difficult than this broadest case, though not as easy as for well-founded apgs.

5.5 Correctness of the μ^δ Computation

Verification with the system of equations showed that the graph constructed as G^3_{b2} in Figure 5.10 is a picture of the same hyperset as that depicted by the graph in Figure 5.11, earlier shown to be $\mu(\texttt{b2})$ (in the subnetwork W', Figure 4.4). But the general problem of proving the correctness of the algorithm used amounts to proving that, for any case, the resulting graph will be a picture of the right hyperset, that is, identical to the hyperset given by the subnetwork rooted at that node in the full network, that is, identical to the hyperset depicted in the source SNePS network by the graph S^δ_n, defined below.

Definition 5.5.1 *The graph S^δ_n = the subgraph of S^* induced by the set of edges $E^\delta_n \subseteq E^*$, where $e \in E^\delta_n \iff e$ is in a path originating at n of length $\leq \delta$ in S^*.*

Note that S^δ_n is both "pointed" and "accessible" and is therefore and apg. The problem, then, is to show that the hyperset pictured by the graph G^δ_n really is the hyperset pictured by S^δ_n.

To begin, here is the explicit algorithm for drawing the graph G^δ_n, given δ and $n \in$ SNodes, embedded in a full network from SNets*:

Delta-Graph Algorithm

Step1: Using the network containing node n, form the system of equations required by the definition of μ^δ, using the decoration $\mathcal{D}^\delta(n)$ as defined.

Step2: For each equation in the system for an unknown $\mathbf{x}$, draw an apg for the set on the right-hand side, using unique nodes for atoms, and call it $f(\mathbf{x})$. One of these apgs will be $f(n)$, of course, and all of them will have nodes of outdegree zero from $\mathcal{A} \cup \mathcal{X}$ labeled with an injective *tag* function.

Step3: Let the graph $f(n)$ be g, the goal graph in progress. Select another graph $f(\mathbf{x})$ from the set and, if there is a node labeled $\mathbf{x}$ in g, replace it with the point of $f(\mathbf{x})$. Find any arcs in $f(\mathbf{x})$ to nodes *tag*ged t such that the node labeled t is already in g and redirect those arcs to node t instead of including a new node labeled t; i.e., keep the *tag* function for g injective. When all graphs $f(\mathbf{x})$ have been integrated, let g be G^δ_n.

Note that G_n^δ will be an apg, and that the only nodes with outdegree zero will be tagged by $\mathcal{A}$, those tagged by $\mathcal{X}$ having been replaced.

The Delta-Graph Algorithm is easier to perform than it is to explain. The replacement of the node $\mathbf{x}$ by the apg $f(\mathbf{x})$ is unambiguous because, if there is a node labeled $\mathbf{x}$, there will only be one, by the construction invariant that the *tag* function is injective, and it will have outdegree zero, since only such nodes get tagged by $\mathcal{D}^\delta$. There may also be other nodes that depict the same hyperset as some $f(\mathbf{x})$ being inserted in place of $\mathbf{x}$, but the algorithm does not attempt to detect it. (See §5.8.1 for discussion of that enhancement.) The result is ungainly constructions like Figure 5.10, and that's why we have the problem of proving that the hyperset it depicts is the same as that depicted by Figure 5.11.

Ultimately, the judgment of equivalence will come down to comparison of sensory node tags, since every graph built as above from an S^* is finite by Axiom 2 and, for every node $n \in$ SNodes and every value of δ, $\mu^\delta(n)$ will include some atom by Axiom 4. The Delta-Graph Algorithm must be shown to entail some relation, based on the equivalence of atoms, that, in turn, entails equivalence of the appropriate hypersets. More precisely, to define equivalence of two hypersets given by apgs G_1, G_2 from some $S^* \in$ SNets*, with points p_1 and p_2, we need a relation Q based only the graphical properties of G_1 and G_2 that suffices to determine equivalence of the hypersets they picture. Here it is; the relation Q that follows means "the hyperset H_1 depicted by G_1 is identical to the hyperset H_2 depicted by G_2".

Definition 5.5.2 $G_1 \; Q \; G_2 \Longleftrightarrow$

$$\begin{cases} tag(p_1) = tag(p_2), \; \textit{if outdegree}\ (p_1) = \ \textit{outdegree}\ (p_2) = 0 \\ \exists\ \textit{bijection}\ b\ :\ (c_1|p_1 \rightarrow c_1) \longrightarrow (c_2|p_2 \rightarrow c_2) \\ \qquad\qquad\qquad\qquad \textit{such that}\ [c_1 \; Q \; b(c_2)],\ \textit{otherwise} \end{cases}$$

The relation between the children of the points p_1 and p_2 can be required to be a bijection because, by Theorem 3 and Theorem 2 and the derivation of μ through the decoration $\mathcal{D}$, it is impossible to have two children of the same node represent the same hyperset. Q is an inductive version of $\equiv_{\mathcal{V}}$, and it is appropriate for comparion of G_n^δ and S_n^δ because they are finite and "ground out" at atoms.

See, for an example application of Q, the two apgs in Figure 5.14, where both $A1$ and $A2$ are meant to be atoms, with the same tag, say, the lexeme 'woodchuck', and consider whether they are pictures of the same hyperset.

The computation of the predicate $G_7 \; Q \; G_8$ starts with the test whether there exists the necessary bijection b between the children of their points, p_1 (labeled $B1$) and p_2 (labeled $D2$), and proceeds recursively as shown in Figure 5.15, with a new bijection sought between new nodes p_1 and p_2 at every level; they are shown by dotted lines. The tree can terminate there, graph theory telling us that since $B1$ and $D2$ have already been matched and expanded, the graph contains a cycle and therefore no new information can be extracted. In other words, all necessary bijections have been constructed.

The important thing about Q is that it is a relation on *graphs* that provides the answer to the main question of this section, "How do we know that the Delta-Graph Algorithm for constructing G_n^δ depicts the right hyperset?" The right hyperset, of course, is the one pictured by S_n^δ.

Theorem 4 $\quad G_n^\delta \; Q \; S_n^\delta$

Proof: By induction on δ:

1. For $\delta = 0$, G_n^δ is the trivial graph (with one node and no edges), decorated with its label, and so is S_n^δ.
2. Assume $G_n^{\delta-1} \; Q \; S_n^{\delta-1}$. Then there is a bijection between descendants of n at every level.

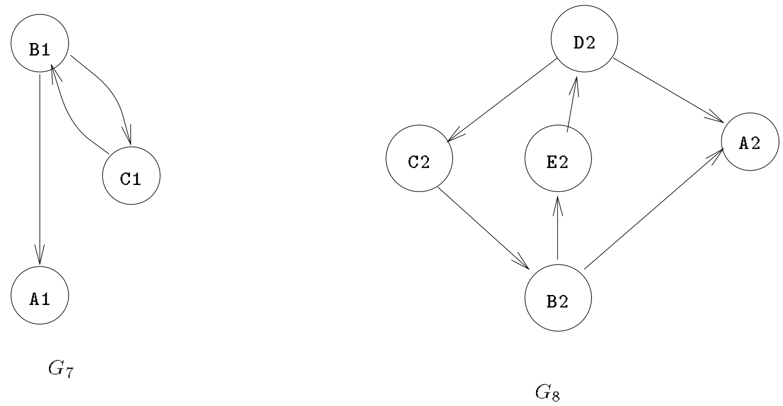


Figure 5.14: Are these pictures of the same hyperset?

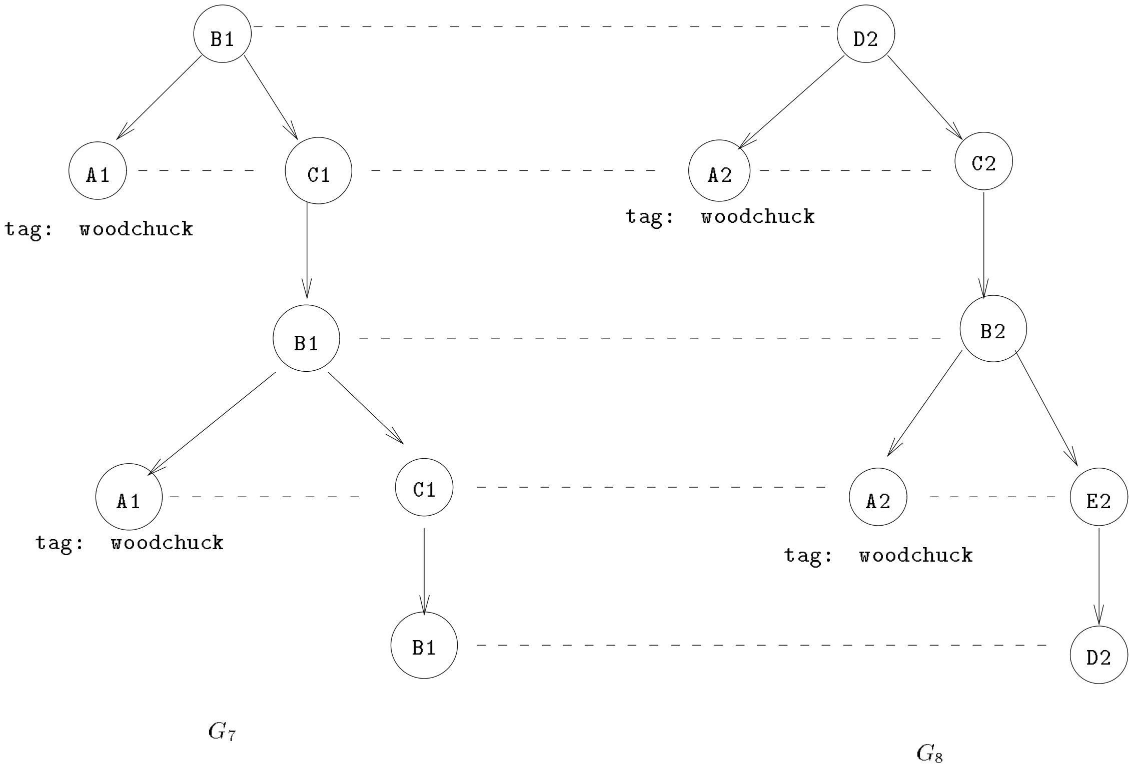


Figure 5.15: Computation of $G_7 \, Q \, G_8$

(a) Let c_G be an arbitrary node of outdegree zero in G_n^δ. It must be the child of some node p_G that had outdegree zero in $G_n^{\delta-1}$. The new node c_G came from either:

   i. The decoration (Steps 1, 2 of the Delta-Graph Algorithm), in which case the node p_S in $S_n^{\delta-1}$ that corresponded to p_G by the assumption has a child c_S in the SNePS network; or,

   ii. Replacement of p_G by $f(p_G)$ containing the child node c_G (Step 3 of the algorithm), which is only possible if the equation defining p_G shows that it contains c_G, a fact that must be reflected in S_n^δ as $p_S \rightarrow c_S$.

(b) For an arbitrary node of outdegree zero c_S in S_n^δ, which has parent p_S (among others, possibly), there must be a corresponding c_G because the fact that $p_S \rightarrow c_S$ must be reflected in the equations that are, by the Solution Lemma, all satisfied by G_n^δ.

□

We can now show that the limited idea of equivalence Q that is adequate for apgs derived from SNePS networks entails Aczel's more general criteria for hyperset identity. Let the hypersets pictured by arbitrary Delta-Graph Algorithm output apgs G_1 and G_2 be, respectively, H_1 and H_2, unique by the Anti-Foundation Axiom.

Theorem 5 $G_1 \; Q \; G_2 \implies H_1 \equiv_{\mathcal{V}} H_2$

Proof: If $H_1 \; Q \; H_2$, then either both are identical atoms or there is a bijection between the children of n_1 and n_2 for which Q holds; in either case, the conditions for $H_1 \equiv_{\mathcal{V}} H_2$ are satisfied. □

Finally, Aczel provides a chain of implications to show that $H_1 \equiv_{\mathcal{V}} H_2$ implies that $H_1 = H_2$ [Aczel, 1988, pages 20, 21], and we have the desired proof of identity of the hypersets, and assurance that the Delta-Graph Algorithm is correct.

5.6 Analysis of μ^δ-Values

What types of hypersets are returned by the limited μ^δ; how is the restricted circularity manifested? The decoration procedure in $\mathcal{D}^\delta$ reaches out to child nodes one level of arcs at a time, rather than effectively defining subgraphs induced by a set of nodes, which would include *all* arcs between parent and child. This property, in conjunction with Axiom 4 (all base and molecular nodes lead to sensory nodes) prevents the application of μ^δ to any subnetworks in which all nodes have children. The difference between μ^1(`b2`) in Figure 5.2 and μ(`b2`) under SCOPE of 1 in Figure 5.3 illustrates the problem avoided. The property is worthy of formalization as a lemma concerning the qualified μ^δ, similar to Lemma 1.

Lemma 6 *$\forall n \in SNodes$ and $\forall \delta \geq 0$, $\mu^\delta(n) \neq \Omega$*

Proof: Given n, $\mu^\delta(n)$ can only be Ω for some δ if every node in the hyperset assigned through μ, $f(n)$, has a child. This would be the case if all nodes at some level of decoration $\mathcal{D}^k$ were identical to nodes already in the tree at an earlier level. Then replacement of those nodes by their apgs in the Delta-Graph Algorithm would create cycles at each of them, making each, and therefore the entire solution apg, a picture of Ω. But this cannot be the case, because subsequent levels of decoration would not be able to introduce any new elements, and there would therefore be molecular nodes that did not dominate (in S^*) sensory nodes, in contradiction to Axiom 4. □

Quite a strong claim can, in fact, be made about the power of μ^δ to discriminate meanings of nodes, to the effect that Theorem 3 holds for every value of δ in a full cognitive unigraph C' (with the same necessary exception as in Theorem 3).

Theorem 7 *$\forall n, m \in C'$ where $n \neq m$ and $\forall \delta \geq 0$, unless n and m dominated exactly the same subordinate nodes (through arcs labeled with different relations),*

$$\mu^\delta(n) \neq \mu^\delta(m)$$

Proof: Assume not; assume that $\mu^\delta(n) = \mu^\delta(m)$, for some δ and some distinct nodes n and m in C. Again, by the Solution Lemma, there is only one way to decorate n and only one way to decorate m. Those decorations depend on the network structure subordinate to the nodes; if the decorations are the same, then n and m share all conceptual linkage, impossible for distinct nodes in SNePS, by its Uniqueness Principle (or by the Theorem 3, the Uniqueness Principle for μ). $\Box$

In other words, it is not necessary to go to any length to determine that the meaning of n_1 differs from the meaning of n_2; it is only necessary to compare their sets of immediate descendants.

The relation Q (and its acceptance as the definition of equivalence) allows the derivation of an interesting aspect of SNets*—that the apgs contained in it are minimum pictures as values of μ. The *canonical picture* of a hyperset a contains only the nodes and arcs necessary to show the hereditary membership relations among the elements of a [Aczel, 1988, page 5]. By the Uniqueness Principle for SNePS, the decoration of any node $n \in$ SNodes will be injective, and therefore $S^\delta(n)$ will be a canonical picture. When there *is* a discrepancy in size—'size' being $|V|$ and $|E|$—between the solution as expressed in S^δ and that derived through the computation of μ^δ, the former will be the smaller (as seen in Figures 5.10 and 5.11).

5.7 Distance Between Nodes

The qualified $\mathcal{D}^\delta$ and μ^δ provide a measure of distance between nodes, and therefore between their semantic values, and between their respective semantic influences as well, in the sense that every arc traversed is a set formed; $n \neq \{n\} \neq \{\{n\}\}$, etc., making an actual count of "levels of influence," perhaps, or "indirect reference," between concepts. Recall the discussion in §2.8.3 of *x-equivalence* between node meanings $[\![n]\!]$ and $[\![m]\!]$, where x is length of the semipath from n and m that must be traversed until the induced subnetworks differ. How do the formal results of preceding sections refine this idea?

First of all, only for distinct cognitive agents does it make sense to ask "when" two node meanings become different. As established by Theorem 7, any two nodes in the same cognitive agent are different *right away*, to any distance greater than (or equal to) zero. If the question is how much meaning they share, some other metric must be developed.

One version of such a metric might measure the relative semantic influence of two nodes n and m on a third. Consider node `m22` in Figure 4.1, which represents the proposition that a particular table exerts something called a 'force'. To use an example the motivation for which is left to the generosity of the reader, we might ask about the relative influence on μ(`m22`) of two concepts—(1) the particular table in question, node `z`, and (2) the concept of exertion, node `m10` (in the verb form '`exert`'). The direct approach is counting the arcs on the set-membership paths used for μ (that is, the paths that appear in the S^* version of this network, Figure 4.3) from `m22` to `z` and `m10`. The lengths of the (shortest) paths, respectively, are one and four, for a difference in length of three. Computation of the decoration of `m22` yields the same answer, in the form of the difference in the values of δ at which $\mathcal{D}^\delta$(`m22`) includes `z` and `m10`:

$$\begin{aligned}
\mathcal{D}^0(\texttt{m22}) &= \texttt{m22} \\
\mathcal{D}^1(\texttt{m22}) &= \{\texttt{m21}, \texttt{z}, \texttt{m17}\} \\
&\quad \text{The element } \texttt{z} \text{ has appeared at level 1.} \\
\mathcal{D}^2(\texttt{m22}) &= \{\{2, \texttt{m18}, \texttt{w}\}, \{\texttt{m20}, \texttt{m22}, \texttt{m17}\}, \{\texttt{m15}\}\} \\
\mathcal{D}^3(\texttt{m22}) &= \{\{2, \{\texttt{m8}, \texttt{w}\}, \{\texttt{m18}, \texttt{m21}, \texttt{m19}\}\}, \\
&\quad \{\{\texttt{m19}, \texttt{z}\}, \{\texttt{m21}, \texttt{z}, \texttt{m17}\}, \{\texttt{z}, \texttt{m15}\}\}, \{\{\texttt{Table}\}\}\} \\
\mathcal{D}^4(\texttt{m22}) &= \{\{2, \{\{\texttt{force}\}, \{\texttt{m18}, \texttt{m21}, \texttt{m19}\}\}, \{\{\texttt{m8}, \texttt{w}\}, \{2, \texttt{m18}, \texttt{w}, \texttt{m19}\}, \{\texttt{w}, \texttt{m10}\}\}\}, \\
&\quad \{\{\{\texttt{w}, \texttt{m10}\}, \{\texttt{m20}, \texttt{m22}, \texttt{m17}\}\}, \{\{2, \texttt{m18}, \texttt{w}, \texttt{m20}\}, \{\texttt{m20}, \texttt{m22}, \texttt{m17}\}, \{\texttt{m15}\}\}, \\
&\quad \{\{\texttt{m20}, \texttt{m22}, \texttt{m17}\}, \{\texttt{Table}\}\}\}, \{\{\texttt{Table}\}\}\}
\end{aligned}$$

The element `m10` has appeared at level 4.

So `z` has a "closer" or "stronger" influence on `m22` than does `m10`.

We can also count the raw distance between nodes in the same agent, leading to another inquiry about this example (with perhaps even shakier cognitive motivation): What is the distance between `z` and `m10`? The answer is three, obtained once again easily through examination of the network S^*. It is not clear that there is an interesting semantic use for this number. The fact that the concept of the exerted force itself, as represented by node `y`, does not participate in μ(`m22`) may be of interest, however. (Other molecular nodes in the network embody the concepts that tie *that* force to *this* table.) To complete the distance mechanism, we might define the distance between `m22` and `y` to be infinity.

Although inappropriate for a single cognitive agent, it still makes sense to ask "when" two nodes differ in meaning in *distinct* cognitive agents. If Figure 4.1 shows CASSIE's mind while observing the interchange between Nancy and Tom, consider the same network in the mind of another cognitive agent OSCAR, except that he notices in addition that the table needs polishing. That concept would be represented as a molecular node dominating `z`, presumably with other complex structure attached to it. CASSIE's `m22` would differ from OSCAR's `m22` at level two, as soon as the table node `z` were expanded into its elements in each $\mathcal{D}^2$(`m22`).

5.8 Algorithmic Analysis

SNePS networks in their derived forms SNets* (and sometimes the acyclic unigraphs SNets$'$) are graphs consisting of given sets of nodes V, with cardinality $|V|$, and arcs E, with cardinality $|E|$. They could be given as LISP expressions, or matrices, or structured strings. We note that there are some characteristics of SNets* that distinguish them from graphs with homogeneous (untyped) nodes and unbounded cyclicity.

1. All cycles are of length two.
2. No sensory node is in a cycle.
3. All base nodes are in cycles.

5.8.1 Complexity of computation of μ^δ

For construction of a graph G_n^δ with the Delta-Graph Algorithm, the problem instance consists of the construction of several apgs and their combination in such a way that all nodes labeled with an indeterminate are replaced by the apg representing the hypergraph assigned to that indeterminate by the solution. Each step of the algorithm will be examined for time requirements.

Step1: All the real work is done in the decoration, for which a recursive algorithm is implicit in the definition of $\mathcal{D}^\delta$. A bottom-up approach, such as dynamic programming, might appear to be an efficient choice [Brassard and Bratley, 1988, Chapter 5], but since computation of $\mathcal{D}^i(n)$ is not necessarily independent of $\mathcal{D}^{i+1}(n)$, the recursion must go all the way down to $\mathcal{D}^0(n)$ in order for $\mathcal{D}^i(n)$ to be computed. Indeed, the computation of the decoration at one level of a node n embedded in a unigraph C' of $|V|$ nodes involves $O(|V|)$ nodes, and there are δ levels. The number of recursive steps required in decoration is therefore $O(|V|^\delta)$. Since δ is bounded by the number of nodes $|V|$ in the sense that $\mathcal{D}^{|V|+1}$ cannot add anything to the decoration not already there in $\mathcal{D}^{|V|}$, the complexity measure independent of δ is $O(|V|^{|V|})$. Explicit decoration as defined by the algorithm is an exponential task, not polynomial.

Step2: For each equation defining an unknown $\mathbf{x}$, the apg can be drawn as the preorder construction of a tree (starting a subtree with each '{', finishing one with each '}'), except at the bottom level, where the tree property is lost because there may be nodes with indegree > 1. There are $O(|V|)$ equations in the system, and the last level requires a search through $O(|V|)$ nodes, making this step $O(|V|^2)$.

Step3: The apgs $f(\mathbf{x})$ are inserted, in some order, into the apg $f(n)$. With $O(|V|)$ equations and searching at the bottom level, again, the time requirement is $O(|V|^2)$.

What would be the cost of improving the Delta-Graph Algorithm so that it yields smaller graphs G^δ, perhaps the graphs already in the network? The difference between G^3_{b2} in Figure 5.10 and the apg $\mu(\texttt{b2})$ of Figure 5.11 is due to the fact that no "minimization" algorithm was employed in the incorporation of the hypersets in Figures 5.7 and 5.8 into that of Figure 5.6. A search for an appropriate subgraph that already existed would have revealed that $f(\texttt{m24})$ and $f(\texttt{m5})$, with the addition of backward arcs, would fit into the left and right child nodes, respectively, of $f(\texttt{b2})$. Consider an algorithm for the search problem consisting of:

- *Instances:* Pairs of apgs $\langle f(\mathbf{x}), f(\mathbf{y}\rangle$, where $\mathbf{x}, \mathbf{y} \in \mathcal{X}$, the unknowns
- *Solutions:* $\{p \mid p$ is a node in $f(\mathbf{x})$ and p is the point of an apg α such that $f(\mathbf{y}) = \alpha\}$

Note that the solution set for each instance will be either empty or a singleton. Enhancing the Delta-Graph Algorithm with such a mechanism would maintain the invariant that the decoration is injective—every hyperset involved would be represented by exactly one node—and would therefore result in the construction of a smaller G^δ_n. Its resource demand, however, would not be trivial. We have a target apg $T_1 = V_1 \cup E_1$ and an apg to insert, $T_2 = V_2 \cup E_2$, and we seek a subset of the nodes and edges of T_1, respectively, V and E, such that there is a bijection $h : V \longrightarrow V_2$ satisfying the requirement that $\langle h(u), h(v)\rangle \in E_2 \iff \langle u, v\rangle \in E$. This problem is the detection of subgraph isomorphism, which is NP-complete [Garey and Johnson, 1979, page 202].

And that's not even the end of the story. The better solution in the example above would not be detected by incremental incorporation of apgs, since *both* $f(\texttt{m24})$ and $f(\texttt{m5})$ have to be examined before it is apparent that they picture the same hypersets as the apgs at the right and left child nodes of $f(\texttt{b2})$. It is impossible to tell from examination of only $f(\texttt{b2})$ and $f(\texttt{m24})$ that the latter would be depicted by the apg rooted at the left child of $f(\texttt{b2})$ with the addition of one backward arc from that node to the point. It is clear from the $f(\texttt{m24})$ apg alone that it includes $f(\texttt{m5})$, but it is not clear that that entails inclusion of `Tom`. Determination of all such interdependent relationships would make the problem intractable were it not already. Note that the problem would be obviated by the retention of labels on the nodes, but they are not part of the hyperset or its pictures.

Strictly speaking, however, the "best" placement of hyperset assignments into the overall solution is unnecessary. The mechanically constructed G^3_{b2} is a picture of the same hyperset as the more elegant Figure 5.11. For these reasons, the most important result concerning the complexity of the computation of μ^δ is that *it is already done by the SNePS network in the S^* form.* We don't actually need to compute $\mathcal{D}^\delta(n)$. As long as we can count arcs, Theorem 2 assures us that the hyperset that would be assigned to n in $\mu(n)$ is the subgraph S^δ_n rooted at n, to some path length δ. The Delta-Graph Algorithm, in other words, is unnecessary to determination of $\mu(n)$ in a given (full) SNePS network. There may be a need for it, however, in unanticipated situations involving independently determined hypersets from $\mathcal{V}_{\mathcal{A}\cup\mathcal{X}}$.

5.8.2 Complexity of measuring distance

Two distance questions discussed above are open to solution by more straightforward and efficient algorithms. Given two nodes n, m in the same S, both the question whether $m \in \mu(n)$ and the measurement of distance between n and m are amenable to standard polynomial search algorithms.

- Is $m \in \mu(n)$? Note that this problem exists because SNePS networks are only *weakly* connected, rather than *strongly* or *unilaterally* connected. If every $S \in$ SNets were strongly connected, the answer would be trivially "yes". That could be the case also if the decoration as shown in the derived S^* had included further circularity. But under this development, it remains a real question. Above, it was pointed out that $y \notin \mu(\texttt{m22})$ in Figure 4.1. For another example using the same figure, check whether `'force'` is in $\mu(\texttt{m19})$. Yes. But the concept of Tom, node `b2`, is not. For this decision problem, using breadth-first search (or depth-first search), the entire S^* may have to be traversed for a negative answer.

- What is the distance from n to m (or vice versa)? Again, breadth-first search will give the answer, the answer being ∞ if there is no path from n to m in S^*.

Both questions can be answered, therefore, in $O(|V|^2)$ steps [Maurer and Ralston, 1991, page 249 ff.].

5.8.3 Complexity of computing the unqualified μ

Lastly, note that the discussion of the decoration algorithm above gives a resource bound for the computation of the unqualified μ of Chapter 4—even though, again, that computation is redundant to the derivation of S^*, which contains everything there is to know about μ. Since the number of nodes $|V|$ bounds the possible semantic influences on a node n, $\mathcal{D}(n)$ is the same as $\mathcal{D}^{|V|}(n)$ and $\mu(n)$ is the same as $\mu^{|V|}(n)$.

5.9 Summary

Whereas the semantics $\mu(n)$ will probably be of only theoretical interest in a "life-size" SNePS network, a semantics $\mu^{\delta}(n)$ restricted to interpreting a node in terms of only its neighboring nodes may be pragmatically applicable (and, of course, would provide $\mu(n)$ itself for sufficiently large neighborhoods). Such a μ^{δ} was defined, an algorithm sketched for computing it, and the algorithm proved to yield the correct hyperset.

Chapter 6

Compositionality

6.1 Introduction

The foregoing analysis was inspired in large part by the issue whether SNePS is semantically *compositional*. In this chapter, that question is pursued, starting with a look at compositionality in general, and ending with consideration of the effects of the non-well-founded set semantics μ on the compositionality of SNePS.

6.2 Informal Definitions of Compositionality

The Principle of Compositionality with regard to language is stated by a group of linguists in [Gamut, 1991, page 11] as "two principles of Frege":

1. The reference of a composite expression is a function of the references of its component parts.
2. The sense of a composite expression is a function of the senses of its component parts.

Some views explicitly require a mapping between the syntactic combining operations. From Smith, discussing it in more general terms as a "correspondence" relation between abstract domains:

> [Domains] can be defined *compositionally*, in the sense that what corresponds to (or is corresponded to by) a whole is systematically constituted out of what corresponds to (or, again, is corresponded to by) its parts. If the part/whole relation is itself absorbed, a very strong version of compositional correspondence obtains, where parts of a source correspond to parts of the target. [Smith, 1987, page 11]

Partee *et alia* offer the following definition of compositionality (calling it "Frege's principle") to open their comprehensive look at the subject as a linguistic property, *followed* in the case of formal languages, and *sought* in the case of natural languages:

> **The Principle of Compositionality**. The meaning of a complex expression is a function of the meanings of its parts and of the syntactic rules by which they are combined. [Partee *et al.*, 1990, page 318]

with the remarks that follow on its limitations:

> ...the exact import of the compositionality principle depends on how one makes precise the notions of *meaning*, of *part*, and of *syntactic rule*, as well as on the class of functions permitted to instantiate the "is a function of" requirement.

Fodor and LePore put it this way, also as a property of languages:

> A language is *compositional* iff (idioms aside) the meaning of its syntactically complex expressions is a function of their syntatic structures together with the meanings of their syntactic constituents. [Fodor and LePore, 1991, page 332]

They provide an enumeration of the manifestations of compositionality in natural language: *productivity*, "roughly, the fact that every natural language can express an open ended set of propositions"; *systematicity*, "roughly, the fact that any natural language that can express the proposition P will also be able to express many propositions that are semantically close to P"; and the property that the structure of sentences is *isomorphic* to the structure of the propositions they express, in the sense that "if a sentence S expresses the proposition that P, then syntactic constituents of S express the constituents of P." [ibid.]

In essence, compositionality requires that there be a *grammar* of meanings, that there exist some set of primitive meanings and that there be rules for the composition of others from these (*and* that these rules correspond closely to the rules of the syntactic grammar). Note that this makes the set of meanings, i.e., the semantic range $\mathcal{Z}$, recursively enumerable. (Formalization follows.)

6.3 Compositionality in Natural Language

The question whether any natural language (taking English as our paradigm without apparent loss of generality) can be said to be compositional has been the subject of debate for years among linguists. Barbara Partee surveyed, explained, and critiqued the various points of view in [Partee, 1984]. Recently (late 1991), a question about the nature of *non-compositional* semantics generated discussion on the LINGUIST LIST, an electronic bulletin board, with particular contributions mentioned below [Linguist List, 1991].

6.3.1 The Troublesome Constructions

For English, the syntactic domain $\mathcal{Y}$ is words or other constituents (the atoms $\mathcal{A}$) and sentences (the compound objects $\mathcal{B}$), and the semantic domain $\mathcal{Z}$ is the meanings of words, phrases, other constituents, and sentences. If English has a compositional semantics, then the meaning of a sentence is a function of the meanings of its words. Indeed, this is clearly the case most of the time. One understands the meaning of the sentence "Violet drove home" by virtue of understanding the respective meanings of the words 'Violet', 'drove', and 'home', and the semantic effects of their grammatical combination in this order.

But the following sentences are not so easy to interpret this way:

S1: Susan dated an occasional sailor. [Linguist List, 1991, Hutchinson (Vol. 2, No. 507)]

S2: Fifty is what forty used to be. (G. Steinem, cited by [Wilks, 1984])

These examples are meaningful to English speakers, but not in a way that easily reflects a combination of the meanings of their words. In sentence S1, it is difficult to formulate a meaning for 'occasional' that will allow it to play the role of modifier to the word 'sailor' without generating a noun phrase constituent meaning something like 'a guy who sails every now and then'—a semantic item that plays no role in the meaning of the resulting sentence. In sentence S2, although 'fifty' and 'forty' can be pre-defined to include the idea that they are ages, how can the verbs 'is' and 'used to be' come bearing the necessary shades of meaning that allow the strong distinction between the numeric values to be overridden in this usage but not others? Note that **pre**-definition is necessary. A compositional semantics requires that the semantic contribution be unidirectional: All necessary meaning must be incorporated into the constituents before they are combined into a compound object; otherwise, of course, the compound is not a function of its components. (See "bottom-up" versus "top-down" interpretation, [Partee *et al.*, 1990, pages 286-287].)

Yorick Wilks argues that the principle of compositionality in natural language is either trivial or false—*false* for the reasons given in the examples above, or *trivial* in that nothing is ever taken as refuting it. To switch languages for a moment, the Spanish sentence "Esto no significa nada" (glossed as "That means nothing") contains two constituents that contribute negation, so in a compositional system, one of them must be taken as having no semantics (requiring an arbitrary choice) [Wilks, 1984].

Consider pluralization in general. In the word 'dogs', the constituent that marks that the semantics is plural is the '-s' suffix. What then is the marker for singular in 'dog'? One hypothesis is a zero (empty) morpheme [Linguist List, 1991, Fintel (Vol. 2, No. 523)], a solution with some respectability. But extrapolation would require a zero morpheme for all unmarked features of words. In other words, it is not enough to claim that default cases hold unless otherwise specified. Truly compositional semantic rules, if they are to handle the plural because of some marking, must recognize the singular by some marking; they must be written to anticipate the distinction. The marking may indeed be the lack of something, but it must be the lack of something in a pre-defined place. This is a transformation of the question of compositionality into the problem of making an exhaustive list of features. If these are the measures that must be taken to preserve compositionality, then it is not being driven by the data, but vice versa. If examples S1 and S2 lead to backward definition of the relevant words, with the justification that they must have had more to their meanings to start with than we realized, then Wilks's point is made: The principle of compositionality stands simply because no contradiction of it is accepted.

6.3.2 What non-compositionality is not

Some suggested alternatives to compositionality turn out, on formal analysis, to be only flourishes on the basic notion. Such analysis will help elucidate the critical elements of compositionality and the nature of a compositional semantic function ν for English. The question is what can be said about the possibility of the existence of a *computable* $\nu : \{$ English words and sentences $\} \longrightarrow \{$ Meanings $\}$. The formal tool is recursive function theory, sketched in §2.3.

Compositionality is not violated by exceptions. Any finite set of exceptions to the compositional semantic function can be built into a more discriminatory function. Idioms, such as "kick the bucket", present no qualitative challenge to a semantics that otherwise employs standard meanings for the three constituent words, as long as that semantics recognizes the three-word string and computes its meaning independently of the individual words in it [Linguist List, 1991, Manaster-Ramer, Coates; (Vol. 2, No. 514)]. Note that this requires the set of base, or atomic, elements to include both 'kick' and 'kick the bucket'.

Compositional functions can return sets. A function whose range includes sets is still a function. Every finite set is encodable into a natural number and can therefore be the value of a function for some argument. Set values can be distinguished from single values; in fact, single values would simply be singleton sets. There is nothing formally suspect, that is, about a semantic function that returns the set $\{z_1, z_2\}$ as the value of $\nu(y)$ for some syntactic object y. To want to distinguish between z_1 and z_2 is to want something besides ν.

Compositionality is consistent with a choice of functions selected by some other parameter. To say that the meaning of a constituent is determined by one of several functions, with the selection of function being driven by some other parameter of the environment, such as the context of the sentence, is simply to defer the semantics to some "larger" function that takes that other parameter as an additional argument [Grandy, 1990], [Partee *et al.*, 1990, page 290], [Linguist List, 1991, Nerbonne (Vol. 2, No. 507), Fleck (Vol. 2, No. 523)].

In fact, this is *why* both 'kick' and 'kick the bucket' can be accommodated. Given $\nu_1($ 'fifty' $) = 49 + 1$ (or some other expression that clearly denotes a numeric value) and $\nu_2($ 'fifty' $) =$ *the human state of being 50 years old*, and some other factor, such as *category*, which switches to either ν_1 (if category is 'adjective') or ν_2 (if category is 'noun'), the semantic function that accomodates both of these, correctly, is:

$$\nu(\text{ `fifty' },\ \text{category}) = \begin{cases} \nu_1(\text{ `fifty' }), & \text{if category is `adjective';} \\ \nu_2(\text{ `fifty' }), & \text{if category is `noun'.} \end{cases}$$

Compositionality need not ignore pragmatics. The pragmatics of a sentence—its larger context—can be part of the input to the compositional semantics. This is the previous case, where the extra parameter is taken to be *context of discourse*. The problems involved in determining the scope and values of the relevant inputs is enormous, but they are not different in kind from those of isolating, into discrete units, the semantic contributions of the constituent or sentence itself [Partee *et al.*, 1990, pages 293–294], [Linguist List, 1991, Nerbonne (Vol. 2, No. 507)].

Compositionality cannot extend only part of the way. To say that some linguistic constructions can be interpreted compositionally but others cannot is to beg the question. Compositionality is intended to apply to any sentence with the antecedent guarantee that *this is a way to figure out the meaning*. If that is only true for some sentences, the problem then becomes how to (antecedently) tell which are the exceptions and how to interpret them. Should these problems be solved, then the resulting method *is* compositional. Partial compositionality is no help (in the theoretical sense) in determining an algorithm for semantics.

6.3.3 What compositionality is—severe

Compositionality in the full sense, that is, without the qualifications considered and rejected above, is no joke. But compositionality in that sense is all there is. Fodor and Lepore argue that the claim of compositionality is a strong one in that it precludes the concurrent upholding of two other popular linguistic theses—the rejection of the anayltic/synthetic distinction and the identification of meaning with inferential role. The argument, using as an example the sentence "Brown cows are dangerous," is that the meaning of this sentence has nothing to do with the inferential power of the words and phrases as such. If the sentence is true, it is true because it is a fact about the world and not derivable from the meanings of the constituents (so compositionality is threatened), unless, of course, the meaning is taken to be distinct from the strictly *analytic* inferential power of the constituents (so the analytic/synthetic distinction returns). [Fodor and LePore, 1991]

In natural language, compositionality is not a choice between well-formed alternatives. That is, non-compositionality is not a theory [Barbara Partee, personal communication]. It is not clear what the implications are, therefore, of a rejection of compositionality. If we understood the universe of possible semantic systems, then it is the properties of the *complement* of the compositional set that we wish to investigate; lacking such an understanding, that formulation gets us no further.

6.4 Formal Definitions of Compositionality

6.4.1 As in denotational semantics

Some versions of the principle of compositionality are stated in terms of allowance for substitution. For example, Gamut elaborates the previous definition thus:

These two principles can also be presented as replacement principles.

- If two expressions have the same reference, then substitution of one for the other in a third expression does not change its reference.
- If two expressions have the same sense, then substitution of one for the other in a third expression does not change its sense. [Gamut, 1991, page 11]

In denotational semantics, compositionality is critical, allowing the straightforward incremental interpretation of computational expressions. Consider the **Substitution Lemma** (with ξ for evaluation and "$[\![\ .\ ]\!]$" for meaning) for the λ-calculus, and its paraphrase:

$$\xi\ [\![\ [\ E_1/I\ ]\ E_0]\!]\ \rho = \xi\ [\![E_0]\!]\ (\rho\ [\ \xi\ [\![E_1]\!]\ \rho/I\ ]\ )$$

> This states that the result of evaluating a substituted expression in some environment is the same as evaluating the original expression in a modified environment. [Stoy, 1977, page 161]

If we regard substitution of one expression for another (E_1 for I in E_0) as syntactic manipulation, and modification of an environment (ρ) as semantic manipulation, then licensing of substitution amounts to the same thing as the commutativity of syntactic and semantic transformation as described below.

6.4.2 As a homomorphism between algebras

Compositionality may be formalized as a homomorphism between a pair of algebras $\mathcal{G} = (\mathcal{Y}, R_1^{\mathcal{G}}, R_2^{\mathcal{G}}, \ldots)$, the syntactic algebra, and $\mathcal{H} = (\mathcal{Z}, R_1^{\mathcal{H}}, R_2^{\mathcal{H}}, \ldots)$, the semantic algebra [Partee *et al.*, 1990, pages 335ff]. These algebras may also be viewed as grammars, $\mathcal{G}$ for creating the set of well-formed syntactic constructs, using operations $R^{\mathcal{G}}$, and $\mathcal{H}$ for creating the set of well-formed semantic constructs, with rules $R^{\mathcal{H}}$. Since algebraic operators and grammatical rules can take various numbers of arguments, we assume that each rule applies to a finite set x of arguments and that all are applied correctly (using the set-theoretic implementation of tuples, or sequences, when order matters). For a semantic function ν to be a homomorphism, it must be structure-preserving with respect to corresponding rules $R_i^{\mathcal{G}}$ and $R_i^{\mathcal{H}}$; that is:

$$\nu\,(R_i^{\mathcal{G}}(x)) = R_i^{\mathcal{H}}\,(\nu(x))$$

In other words, the diagram in Figure 6.1 would be a commutative one, for every pair of corresponding syntactic and semantic recursive rules $R_q^{\mathcal{G}}$ and $R_q^{\mathcal{H}}$.

Is it necessary that each semantic rule $R_q^{\mathcal{G}}$ be so tightly coupled with a counterpart rule in the semantic domain $R_q^{\mathcal{H}}$? Could compositionality be captured, in other words, by requiring that every sequence of syntactic combinations $R_i^{\mathcal{G}}, R_j^{\mathcal{G}}, \ldots R_n^{\mathcal{G}}$ resulting in an object $b \in \mathcal{Y}$ have some corresponding realization in $R_k^{\mathcal{H}}, \ldots R_m^{\mathcal{H}}$ resulting in $z = \nu(b)$, but not necessarily in tandem? No—unless the correspondence is predictable, unless $R_i^{\mathcal{H}}$ is antecedently mapped to $R_i^{\mathcal{G}}$, compositionality is lost [Partee *et al.*, 1990, page 337]. This is Fodor and Lepore's requirement of *isomorphism* [Fodor and LePore, 1991].

The properties of algebras and homomorphisms entail that a semantic function $\nu : \mathcal{Y} \longrightarrow \mathcal{Z}$ is *compositional* if and only if it has all of the following **Compositional Properties**:

CP1: The semantic function ν is defined over all $y \in \mathcal{Y}$ that appear in some interpretable compound, by the requirement of closure of the operations $R_i^{\mathcal{G}}$ and $R_i^{\mathcal{H}}$. Stated rigorously,

$$\nu(y)\downarrow\ \Rightarrow\ \forall a \in \{j \mid R_n^{\mathcal{G}}(j) = y\}[\nu(a)\downarrow]$$

CP2: The domain $\mathcal{Y}$ is partitioned into a non-empty collection $\mathcal{A}$ of atomic objects and a collection $\mathcal{B}$ of compound objects; $\mathcal{Y} = \mathcal{A} \cup \mathcal{B}$.

$$\begin{tikzcd}
\mathcal{Y} \arrow[r, "R_q^{\mathcal{G}}"] \arrow[d, "\nu"'] & \mathcal{Y} \arrow[d, "\nu"] \\
\mathcal{Z} \arrow[r, "R_q^{\mathcal{H}}"'] & \mathcal{Z}
\end{tikzcd}$$

Figure 6.1: The elements of compositionality and their arrangement

CP3: The sets $\mathcal{Y}$ and $\mathcal{Z}$ are recursively enumerable, by the existence of the rules $R^{\mathcal{G}}$ and $R^{\mathcal{H}}$ that produce them. If either of the syntax subsets $\mathcal{A}$ or $\mathcal{B}$ is recursively enumerable—if, for instance, the set of linguistic atoms $\mathcal{A}$ is finite—then so is the other.

CP4: For every $r \in R^{\mathcal{G}}$ and $y_1, y_2 \in \mathcal{Y}$, if $y_1 = y_2$, then $\nu(r(y_1)) = \nu(r(y_2))$; the semantics of a compound object depends only on its constituents and the method of their combination.

6.4.3 The existence of a compositional function

Many times the question seems to be not whether a particular known function ν is compositional as defined above, but whether a compositional function is available to do what we want. These cases are those in which the domain $\mathcal{Y}$ is known, the range $\mathcal{Z}$ is known, and the mapping between the two is known (to some degree). Such appears to be the goal in natural language studies. We have some grasp of both the syntactic algebra $\mathcal{G}$ and the semantic algebra $\mathcal{H}$. How they work together to determine the semantics of sentences is partially known, or guessed, as a compositional function ν'. This is not to say that ν' is partially compositional, but rather, that ν' is fully compositional and ν' approximates ν; perhaps $\nu' \subseteq \nu$. Visualize two semantic realms, one the commutative diagram shown in the previous section, with the down-arrows given by the semantic function ν' that is our best approximation, and the other realm, of indeterminate structure, comprising a function $\nu : \mathcal{Y} \longrightarrow \mathcal{Z}$, which is represented by such a diagram only if it is indeed compositional.

For example, it's not too controversial to maintain that ν contains a syntactic rule $R_a^{\mathcal{G}}$ that somehow allows the construction of "Women generally voted for it" from the syntactic elements 'women', 'voted', 'generally', and so forth (other rules in $R^{\mathcal{G}}$ having allowed their construction, in turn, from 'woman', 'vote', etc.), and we assume that there is a corresponding semantic rule $R_a^{\mathcal{H}}$ that assembles the meanings, the values returned by ν for the items 'women', 'voted', 'generally', and so forth, into a coherent meaning, or ν-value, for the whole sentence. Our approximation of these processes of the true semantic function ν, in various natural-language understanding programs or other linguistic formalisms, is ν'. The activities of linguists—including discovery of exceptions, revision of rules, ongoing discussion, and the consensus that no complete grammar for any natural language has yet been written—demonstrate that two premises are held in linguistics: (1) ν' *differs* from ν, and (2) there is some ν which is the goal.

Call ν the "natural" semantic function (e.g., English, as spoken and understood by the competent Anglophone) and ν' the "formal" semantic function (represented, say, by any large English-language machine-understanding project), and distinguish these possible relationships between the two:

- The formal function is "conservative," including only semantic pairings already deemed natural, and maybe not all of those.

$$\nu'(y) = z \;\Rightarrow\; \nu(y) = z$$

- The formal function is "liberal," including all semantic pairings deemed natural, and possibly more.

$$\nu(y) = z \;\Rightarrow\; \nu'(y) = z$$

- The formal function obeys the law of the excluded middle in the natural semantics.

$$(\nu(y) = z_1 \;\&\; z_1 \;\Rightarrow\; \neg z_2) \;\Rightarrow\; \nu'(y) \neq z_2$$

  This is meant to be an example of the successful capture of some particular natural rule of ν in the formal ν'. Many similar constraints are possible.

- The formal function corresponds exactly to the natural function.

$$\nu = \nu'$$

It is not clear, for any given formal attempt ν', exactly which situation holds. If our ν' cannot accommodate "Susan dated an occasional sailor" because our rules only allow the *adjective* ('occasional') to modify the *noun* following it ('sailor'), then is ν' too conservative? Should we add a syntactic/semantic rule pair:

> $\mathcal{R}_k^{\mathcal{G}}$: S $\longrightarrow$ NP VP ADJ NP
> $\mathcal{R}_k^{\mathcal{H}}$: Turn the ADJ into an ADV and interpret it as a modifier for the VP, the first NP as the subject, and the second NP as the object.

And would ν' then become too liberal? It would accept and interpret sentences such as "Susan dated an often sailor," which are (probably) not valid.

6.5 Compositionality in SNePS

The design and development of SNePS is driven, to some extent, by the requirements of natural language (English) competence [Shapiro and Rapaport, 1991]. In fact, according to Smith, it can hardly be helped:

> So the metaphysical problem for semantical theorists is not one of *referring* to the world by using theoretical language, but rather something closer to the opposite: there is no way of referring to the world *except* by using language. [Smith, 1987, page 16; emphasis his]

The considerations given for natural language apply to SNePS, insofar as the meaning of a node is intended to be expressible. The effects of μ on the compositionality of SNePS semantics needs to be considered.

6.5.1 The elements of compositionality provided by μ

Consider the diagram arranging the domains and mappings, the commutativity of which establishes compositionality, Figure 6.1. Assume that the sets of nodes BASE and SENSORY (those that make up individual networks in SNets *other* than the molecular nodes given as the set SNodes) are provided or generated in such a way that both sets are recursively enumerable, and ignore the questions of subnetworks defined by SCOPE or by δ that would force us to consider, for instance, a separate set MOLATOMS. We have this correspondence between the elements of compositionality and the machinery of μ:

$\mathcal{Y}$: Semantics are provided for all classes of nodes, so the syntactic domain is SNodes $\cup$ BASE $\cup$ SENSORY.

$\mathcal{Z}$: The semantic domain is hypersets over the atomic nodes, $\mathcal{V}_{\mathrm{SENSORY}}$.

$\mathcal{R}^{\mathcal{G}}$: The syntactic rules that make new SNodes from old, by definition, are the rules SR.i. We are somewhat at a loss to point to the mechanism that provides new BASE or SENSORY nodes, relying on the assumption above for that.

$\mathcal{R}^{\mathcal{H}}$: The semantic rules that make new hypersets from old are bundled in a single mechanism, the solving of systems of equations $\mathcal{E}$, one for each indeterminate, or new node. The abbreviation $SOL_{\mathcal{E}}$ shall denote this mechanism.

ν: The function ν that assigns semantic values to syntactic values is, of course, μ. We will abuse notation to the extent that μ can take a set of nodes as arguments and return a set of respective hyperset values (computed independently of each other).

The result is the diagram shown in Figure 6.2, and the question whether the diagram is commutative is the question whether, given $n \in$ SNodes and a set x of new nodes, this equality holds:

$$\mu\,(\mathrm{SR.i}\ (n, x)) = SOL_{\mathcal{E}}(\mu(n, x)) \tag{6.1}$$

A reader who has carefully considered the Solution Lemma will expect the answer to be "yes", having recognized that the Solution Lemma itself is some guarantee of a form of compositionality. That sense could be stated as "If you can define new hypersets in terms of old hypersets and each other, then you get what you would have gotten, definitions in terms of the original contents, if the new hypersets had been part of the original system; you can build and then interpret, or you can interpret and then build." And that expectation is borne out by examination. Traveling first across the top of Figure 6.2 from left to right, let SR.i $(n, x) = y$ (for one or more node-formation rules SR.i), creating a new syntactic item y that contains all the proper connections between parts of the structure dominated by n and the new nodes (and arc labels) included in x. Therefore $\mu(y)$, represented by the traversal down the right side, will reflect all of the connections and relationships. Now consider the other traversal from the top left to the bottom right, first traveling down the left side of the diagram in the evaluation of $\mu(n)$, resulting in a collection of hypersets z, followed by the solution SOL of a set of equations $\mathcal{E}$ that define the unknowns x (in terms of $x \cup z$). That solution will, by the Solution Lemma, result in one or more hypersets that reflect all of the connections and relationships captured by $\mu(n)$ and $\mu(x)$. So $\mu(y) = SOL_{\mathcal{E}}(z)$, verifying Equation 6.1.

There is a troublesome aspect to this, however. The system of equations $\mathcal{E}$ used in the solution step $SOL_{\mathcal{E}}$ is not necessarily going to accord with the original hyperset definitions. Some equations in $\mathcal{E}$ may have to override old ones. The definition of circularity chosen, in effect adding out-arcs to base nodes, means that new molecular nodes that point into a base node change the semantics of the base node (and consequently change the semantics of all molecular nodes that use it). An example follows.

$$\begin{tikzcd} \text{SNodes} \arrow[r, "\text{SR.i}"] \arrow[d, "\mu"'] & \text{SNodes} \arrow[d, "\mu"] \\ \mathcal{V}_{SENSORY} \arrow[r, "SOL_{\mathcal{E}}"'] & \mathcal{V}_{SENSORY} \end{tikzcd}$$

Figure 6.2: Compositionality of μ

6.5.2 Example testing compositionality

Recall the interchange regarding the table shown in Figure 4.1 and its extension with the next sentence shown in Figure 4.10. Let us test for the property in Equation 6.1 by computing $\mu(\texttt{b2})$ after the new propositional node `m32` and its subordinate structure has been added (through some unspecified SR.i rules), and comparing that to the hyperset that results from solving the system of equations defining the nodes of the old structure and the nodes of the new structure. No circumscription of the network is done.

The S^* network that results from incorporation of the new sentence's concepts is shown in Figure 6.3, and, by earlier results, the hyperset $\mu(\texttt{b2})$ is the apg rooted at `b2`; it is unnecessary to go through the decoration and solution procedure. So we have the value of the expression on the left-hand side of Equation 6.1. To obtain the other side, a system of equations $\mathcal{E}$, one for each unknown in $\mathcal{X}$ expressing it in terms of a hyperset from $\mathcal{V}_{\mathcal{A}\,\cup\,\mathcal{X}}$, must be written and solved. There are two problems: What is $\mathcal{A}$? What is $\mathcal{X}$?

For $\mathcal{A}$, the obvious choice is the atoms already in the network:

$$\mathcal{A} = \{\texttt{Nancy}, \texttt{ask-whether}, \texttt{Tom}, \texttt{inamimate-object}, \texttt{force}, 2, \texttt{exert}, \texttt{table}\}$$

But this means that the solution would be hypersets over these elements only, not including the new sensory data **`say-that`** and **`think`**. Under strict adherence to the development in §3.4, those two items, like all the new "unknowns", should be members of $\mathcal{X}$, but that would entail equations in $\mathcal{E}$ defining them as hypersets over $\mathcal{A} \cup \mathcal{X}$, and there are no such equations. The only reasonable method of incorporating them is to let $\mathcal{A}$ be automatically extended by new sensory items—call them $\mathcal{A}^+$—before computation of the solution. In other words, the step labeled $SOL_{\mathcal{E}}$ is amplified by a "preprocessor" that sets $\mathcal{A}$ to $\mathcal{A} \cup \mathcal{A}^+$. This policy does not violate the theory of non-well-founded sets—we can define $\mathcal{A}$ to suit our convenience—although it somewhat degrades the elegance of its application to SNePS. The whole problem could be avoided by assuming that $\mathcal{A}$ consists of all possible sensory atoms from the start, but such a policy would clash with what we know of cognition and with SNePS principles. It is not reasonable to assume that a semantic mechanism has access to all possible future inputs. Such a policy may be reasonable to explore, but is beyond the present

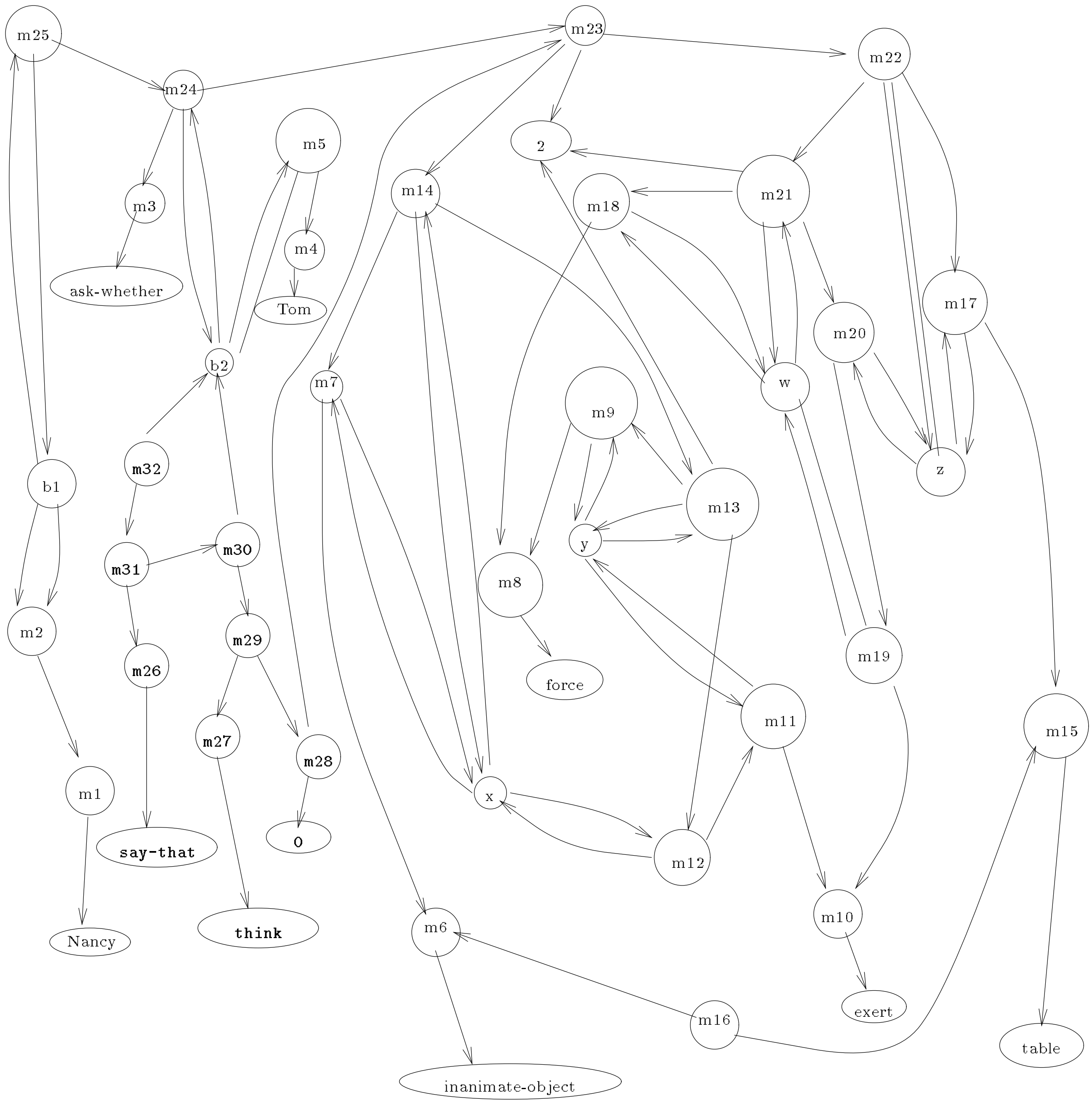


Figure 6.3: S^* of TABLE interchange after Tom's reply

context. Therefore, for the derivation of the solution:

$$\mathcal{A} = \{\texttt{Nancy}, \texttt{ask-whether}, \texttt{Tom}, \texttt{inamimate-object}, \texttt{force}, 2, \texttt{exert}, \texttt{table}, \texttt{say-that}, \texttt{think}\}$$

Now what is $\mathcal{X}$? According to the rules, each new node should be considered an unknown and get a defining equation of its own, so let $\mathcal{X} = \{\texttt{m32}, \texttt{m31}, \texttt{m26}, \texttt{m30}, \texttt{m29}, \texttt{m27}\}$ and write equations for each node, with each expression on the right being a hyperset from $\mathcal{V}_{\mathcal{A}\,\cup\,\mathcal{X}}$:

$$\begin{aligned}
\texttt{m32} &= \{\texttt{b2}, \texttt{m31}\} \\
\texttt{m31} &= \{\texttt{m26}, \texttt{m30}\} \\
\texttt{m26} &= \{\texttt{say-that}\} \\
\texttt{m30} &= \{\texttt{b2}, \texttt{m29}\} \\
\texttt{m29} &= \{\texttt{m27}, \texttt{m28}\} \\
\texttt{m28} &= \{\texttt{m23}, 0\} \\
\texttt{m27} &= \{\texttt{think}\}
\end{aligned}$$

But `b2` appears on the right-hand side of the equation defining `m32` (as it must, since `m32`$\rightarrow$`b2`), so it must be in $\mathcal{X}$ or $\mathcal{A}$. It does not make sense to treat it as an atom, since we are interested in its membership structure, so it must be treated as an unknown. The equation that defines it can easily be added to $\mathcal{E}$:

$$\texttt{b2} = \{\texttt{m24}, \texttt{m5}, \texttt{m32}, \texttt{m30}\}$$

This definition of `b2`, however, contradicts the one already in force:

$$\texttt{b2} = \{\texttt{m24}, \texttt{m5}\}$$

Definitions of the hypersets already in the network are critical to the assignment of hypersets to `m32` and the other new nodes. Assume that they are used as is, except that the original definition of `b2` is replaced with the new one. Then the system $\mathcal{E}$ would enable construction of a solution that looks, of course, exactly like Figure 6.3, since the network is the source of the equations. Therefore we have verification that:

$$\mu\,(SR.i\,(\texttt{b2}, \texttt{m32})) = \text{SOL}\ \ (\mu\,(\texttt{b2}, \texttt{m32}))$$

At what cost?—the reconstruction of whichever of the original hypersets are redefined by $\mathcal{E}$. The solution mechanism requires a second preprocessor that revokes the original definition of any hyperset that is redefined. This will occur any time that the congnitive agent acquires a new concept that dominates a base node, where the semantic influence goes both ways. Again, non-well-founded set theory can accomodate the preprocessing, but its compositonal legitimacy is more dubious. Both the redefinition of given hypersets and the redefiniton of the original $\mathcal{A}$ are instances of the "backward refinement" that compositionality is, strictly speaking, supposed to avoid. In other words, we see that μ can be shown to be compositional under Equation 6.1, but only with careful qualification of the mechanism. There is precedent, however, for such an idea, as follows.

6.6 Compositionality with Qualification

While disavowing any ability to provide one, Fodor and Lepore suggest that a *graded* notion of compositionally might solve the problem. Smith makes a similar suggestion:

> ...we should license a full range of types of correspondence, kinds of circumstantial dependence, and varieties of registration (continuous, discrete, compositional), in terms

> of which subsequently to characterise pictures, maps, graphs, schedules, models, images, and so forth, as well as sentences, formulae, and elements of language. [Smith, 1987, page 14]

Fodor and Lepore ask "But what would a graded notion of compositionality be like? And, in particular, how would such a notion do what compositionality is required to do; *viz.* account for systematicity, isomorphism and productivity?" [Fodor and LePore, 1991, page 341]. Since they define ***systematicity*** as the ability to express propositions that are semantically close to P given the ability to express P, then μ provides it in a literal sense. The meaning $\mu(n_1)$ of a node n_1 that is physically close, in a network, to another node n_2, will be semantically close—in terms of set contents—to $\mu(n_2)$. For *isomorphism*, they require that if a sentence S expresses the proposition that P, then syntactic constituents of S express the constituents of P. While the set-theoretic construction of $\mu(n)$ for some node n appears to reflect this nicely, their additional statement that the structure of sentences should be isomorphic to the structure of propositions they express leaves μ at a loss if "structure" is intended to be more than set membership. As for ***productivity***, the ability to express an unbounded set of propositions, μ has that by virtue of the easy generation of non-well- founded sets from others, with or without new atoms.

In the foregoing example, it appears that the static semantics is compositional, but the dynamic semantics is not. More investigation is required to elaborate on this conclusion, and measure its verisimilitude for other formal semantics.

6.7 Summary

Compositionality of semantics is a strict requirement, best expressed rigorously by the commutativity of a diagram showing that syntactic composition of syntactic units into a syntactic aggregate followed by semantic interpretation is the same as semantic interpretation of syntactic units followed by composition into a semantic aggregate. Yet, as noted by Partee *et alia* at the beginning of this chapter, there are plenty of elements to make precise, and therefore plenty of room for argument. What are the syntactic units? What is the semantic interpretation? What are the two types of composition? All these questions, and others, reflect the difficulty of partitioning the semantic process—even if formally defined—into reasonable discrete steps.

In the case of μ, to claim compositionality is to claim that it can be applied to SNePS nodes either before or after they are combined into SNePS networks, and that the hypersets returned as values by μ are the same. This is so, as long as the context, the set of atoms, $\mathcal{A}$, and the set of equations to solve, $\mathcal{E}$, are adjusted appropriately, a policy which may be criticized as *ad hoc* in the the same way that others discussed here are criticized. The use of non-well-founded sets, if nothing else, gives a richer flavor of compositionality than is found in the world of inductively defined or well-founded structures.

Chapter 7

Alternative Semantics and Other Issues

7.1 Introduction

In this chapter, a variety of issues, nagging questions, and hunches regarding SNePS are briefly considered, mostly concerning other aspects of semantics and possible extensions to or modifications of μ. These alternatives are discussed independently, so the reader should not assume that the modification of one section builds on that of the previous one.

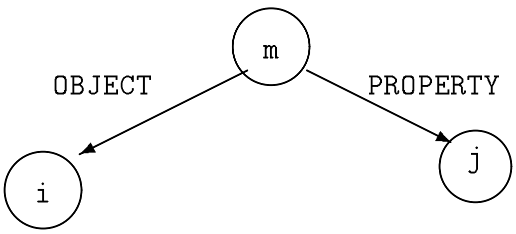
7.2 Alternative Semantics to Handle Arc Labels

What is the role of arc labels, and what exactly is the nature of their "punctuation" function? In [Shapiro and Rapaport, 1987], the two networks given by syntactic rules SR.4 and SR.5 differ only in that one has an arc labeled PROPERTY and the other has an arc in the same position labeled PROPER-NAME; the two networks have different semantics, as expressed in the semantic rules SI.4 and SI.5.

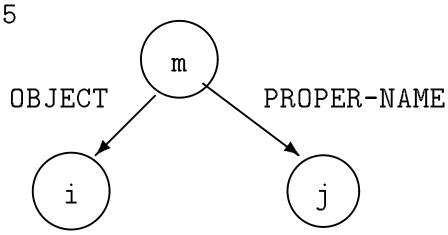
SI.4 m is the Meinongian objective corresponding to the proposition that i has the property j.

SI.5 m is the Meinongian objective corresponding to the proposition that Meinongian objectum i's proper name is j. (j is the Meinongian objectum that is i's proper name; its expression in English is represented by a node at the head of a LEX-arc emanating from j.)

SR.4



SR.5



If arcs have no semantic import, but the meaning of a node is the entire network in which it is embedded, in what principled way can the "structural" contribution of an arc be distinguished from the "semantic" contribution of a node? If arcs make fixed contributions to the meanings of molecular nodes, they should be involved in the semantic function μ. What follows in this section are some ways it might be done, using the arc labels in their formal sense as the set of relations $\mathcal{R}$ given in a SNePS system.

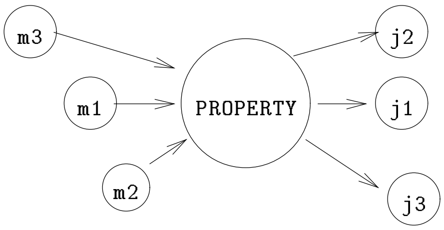
7.2.1 Nodes capturing relations

The relations are concepts in their own right, and therefore nodes. How could the "nodification" of $\mathcal{R}$ be done in a way that minimizes violation of the SNePS paradigm, explicitly maintaining the pairwise relations between nodes somewhere else than in the arc label?

1. They are a special type of node with indegree one and outdegree one:

   These are reminiscent of Conceptual Graphs [Sowa, 1984], but do not belong here. Such an alternative is unprincipled in its violation of the Uniqueness Principle. A SNePS network cannot contain multiple nodes standing for "`PROPERTY`", i.e., multiple nodes n such that $[\![n]\!] =$ `PROPERTY`.

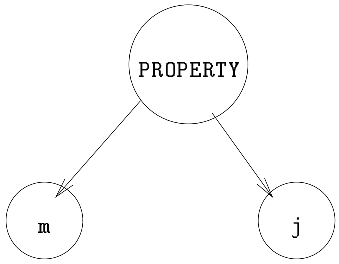
2. Perhaps they are molecular nodes, one for each possible relation.



   Then a node representing a particular relation r has an in-arc from every molecular node that is currently construed as containing that relation r in its cableset, and an out-arc to every subordinate node that is currently construed as the node n in a wire $\langle r, n\rangle$. But where do they come from? In the SNePS paradigm, molecular nodes are built as bundles of subordinate concepts. That means that the arc-node standing for `PROPERTY` would be created before any use of it is made, i.e., before any arcs point into it. This does not seem right.

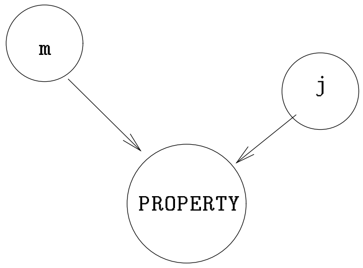
   Furthermore, a "crosstalk" problem results. If the original three sets of connections were `m1` $\longrightarrow$ `j1`, `m2` $\longrightarrow$ `j2`, and `m3` $\longrightarrow$ `j3`, each labeled with `PROPERTY`, the creation of a single `PROPERTY` node now has `m1` also dominating `j2` and `j3`, and so forth. This is certainly not right.

3. So maybe arc-nodes r are best thought of this way, dominating both the original parent and child:



But again, what is the cause? Under what circumstances does a cognitive agent build the molecular node for `PROPERTY`? And is it to be a unique node? If so, there would be no distinction between the sets of `OBJECT` nodes and the set of `PROPERTY` nodes, let alone between the propositions that "`m1` has property `j1`" and "`m1` has property `j2`" (crosstalk again).

4. Perhaps they are not to be "nodified" as molecular nodes at all, but as base nodes.



Insofar as they behave as primitives, this is appropriate. Note, again, the incredibly high degree of connectivity required. *All* nodes that currently have a `PROPERTY` out-arc would dominate the unique `PROPERTY` node. No SNePS researcher should shrink from an incredibly high out-degree or in-degree, after consideration of the effects of the Uniqueness Principle on the node for, say, the concept 'mother', but that rampant domination would, again, lead to the "crosstalk" problem. Computation of semantics under μ, treating `PROPERTY` as a base node would not distingush the assignment of the property `j1` to `m1` from the assignment of the property `j2` to `m1`; computation of the meaning of `m1` with any other semantics would suffer the same ambiguity.

These failed attempts show that "nodifying" relations, to construct from a SNePS network a graphical structure of only nodes and homogeneous connections for Aczel's theory, does not work.

7.2.2 Case frames

It is case frames that are considered to be the units of meaning in the semantic interpretation rules SI.i of [Shapiro and Rapaport, 1987], which give the semantic domain $\mathcal{Z}$ as English statements.

$$\text{SI} : \{ \text{ SNePS case frames from SR rules } \} \longrightarrow \{ \text{ English statements } \}$$

Although appealing in its own way, it is a different semantics. The question where and how the base nodes appear would have to be clarified in order to make the semantics compositional as seems to be intended, and it is not clear that μ lends itself well to expression as English statements, the range of the case frame semantics. The function μ, in its dependence on the notion of the hereditary set, does not respect case frames as units.

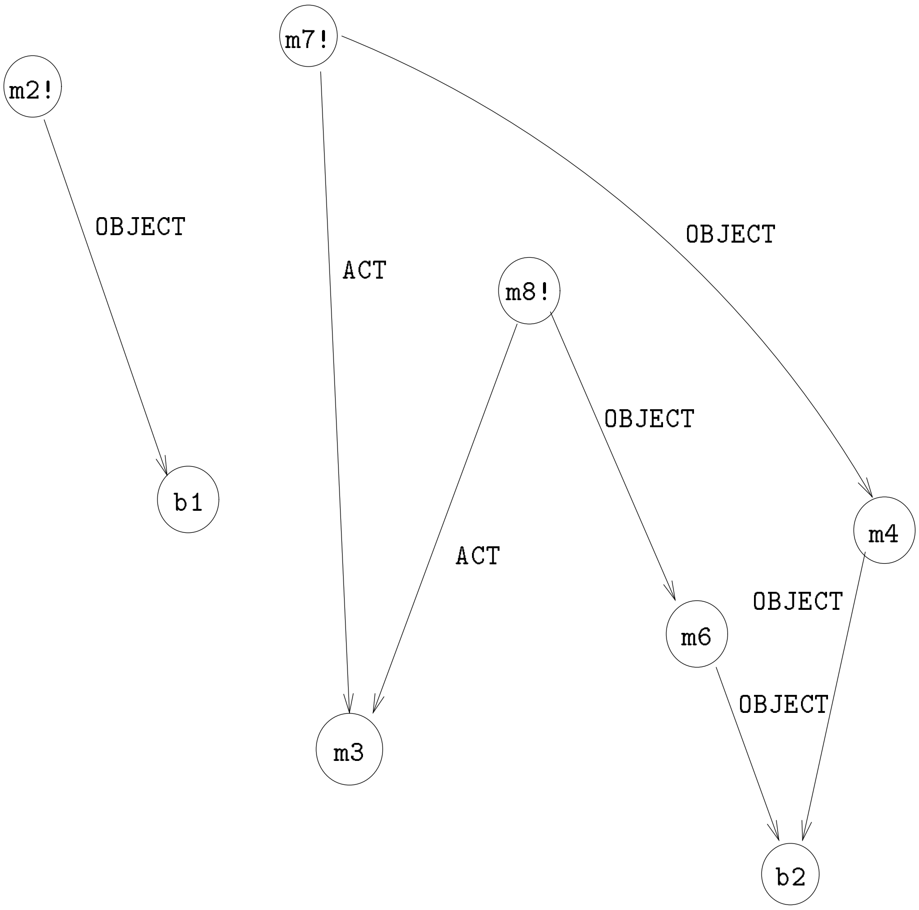


Figure 7.1: Subnetwork of Figure 2.1 induced by { **OBJECT**, **ACT** }

7.2.3 Subgraphs induced by relations

Another approach to the semantic consideration of the relations given by arc labels is to view the part of the SNePS network labeled by that relation as the semantics of that relation. Given one or more relations, the subnetwork induced by the arcs labeled with those relations shows the (collective) meaning of those relations to the cognitive agent. So to see, for example, the semantic contribution of the relation **PROPERTY**, extract from the full network all connections made with arcs labelled **PROPERTY**. Consider the example SNePS network from Chapter 2, Figure 2.1. The subnetwork induced by the relation set { **OBJECT, ACT** } is shown in Figure 7.1. This makes the semantics of relations as arc labels strictly extensional. To say that Figure 7.1 shows the meanings to this (very limited) cognitive agent of the relations **OBJECT** and **CLASS** is to say that the meaning is simply the set of instances of the relations.

Would any such subnetworks be *connected* graphs? Consider, for example, the **MEMBER/CLASS** arcs, corresponding to what appear to be robust and fundamental notions. Perhaps a fully-developed "mind" implemented in SNePS would have these arcs running through the bulk of the concepts, giving this approach a richer notion of the meaning of an arc than appears in the small example above. This is a question that must be settled by study of full cognitive agents.

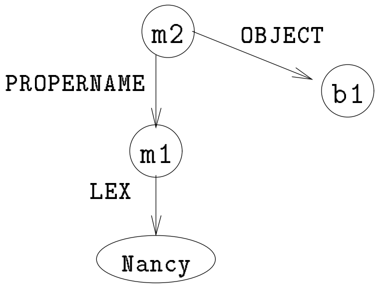


Figure 7.2: The small subnetwork Z

7.2.4 Relations as constituents of molecular nodes

The definition of a molecular node is a set of a wires, and wires are ordered pairs ⟨relation, head-node⟩, and ordered pairs can be expressed as sets:

$$\langle a, b\rangle = \{\{a\}, \{a, b\}\}$$

These properties can be used to enhance the original μ semantics of Chapter 4 to provide a much richer value for the semantics $\mu(n)$ of a node n, which treats the relations $\mathcal{R}$ as *atoms*, along with the sensory data.

Let us return to the examples of §4.5. We take the network context to be the very restricted Z, shown in Figure 7.2, from which Z^* of Figure 4.6 was derived (through Z', by ignoring parallel arcs and arc labels). Instead of the semantically sterile labels for molecular nodes, we use their definitions, given in §2.5, as sets of wires, which are ordered pairs, and convert the ordered pairs to sets.

$$\begin{aligned}
\mathtt{m1} &= \{\langle \mathtt{LEX}, \mathtt{Nancy}\rangle\} \\
&= \{\{\{\mathtt{LEX}\}, \{\mathtt{LEX}, \mathtt{Nancy}\}\}\} \\
\mathtt{m2} &= \{\langle \mathtt{PROPERNAME}, \mathtt{m1}\rangle, \langle \mathtt{OBJECT}, \mathtt{b1}\rangle\} \\
&= \{\{\{\mathtt{PROPERNAME}\}, \{\mathtt{PROPERNAME}, \mathtt{m1}\}\}, \{\{\mathtt{OBJECT}\}, \{\mathtt{OBJECT}, \mathtt{b1}\}\}\}
\end{aligned}$$

So for this small example, the set of atoms, extended to include the relations that participate in the semantics, is $\mathcal{A} = \{\mathtt{Nancy}, \mathtt{LEX}, \mathtt{PROPERNAME}, \mathtt{OBJECT}\}$. To apply the Solution Lemma to the same task as before—finding assignments to the selected set of indeterminates, $\mathcal{X} = \{\mathtt{m2}, \mathtt{b1}\}$—we need a system of equations expressing them as hypersets over $\mathcal{A} \cup \mathcal{X}$.

$$\begin{aligned}
\mathtt{m2} &= \{\{\{\mathtt{PROPERNAME}\}, \{\mathtt{PROPERNAME}, \{\{\{\mathtt{LEX}\}, \{\mathtt{LEX}, \mathtt{Nancy}\}\}\}\}\}, \\
&\quad\ \{\{\mathtt{OBJECT}\}, \{\mathtt{OBJECT}, \mathtt{b1}\}\}\} && (7.1) \\
\mathtt{b1} &= \{\mathtt{m2}\} && (7.2)
\end{aligned}$$

Compare these equations to the system given in Equations 4.1 and 4.2, where the set of atoms was $\mathcal{A} = \{\mathtt{m1}\}$. In Equations 7.1 and 7.2, m1 no longer exists as an object, having been superseded by its definition as a set of wires.

As the solution f, of course, we want hypersets over $\mathcal{A}$, that is, hypersets from the universe $\mathcal{V}_{\mathcal{A}}$, such that the following relationships are maintained:

$$\begin{aligned}
f(\mathtt{m2}) &= \{\{\{\mathtt{PROPERNAME}\}, \{\mathtt{PROPERNAME}, \{\{\{\mathtt{LEX}\}, \{\mathtt{LEX}, \mathtt{Nancy}\}\}\}\}\}, \\
&\quad\ \{\{\mathtt{OBJECT}\}, \{\mathtt{OBJECT}, f(\mathtt{b1})\}\}\} && (7.3) \\
f(\mathtt{b1}) &= \{f(\mathtt{m2})\} && (7.4)
\end{aligned}$$

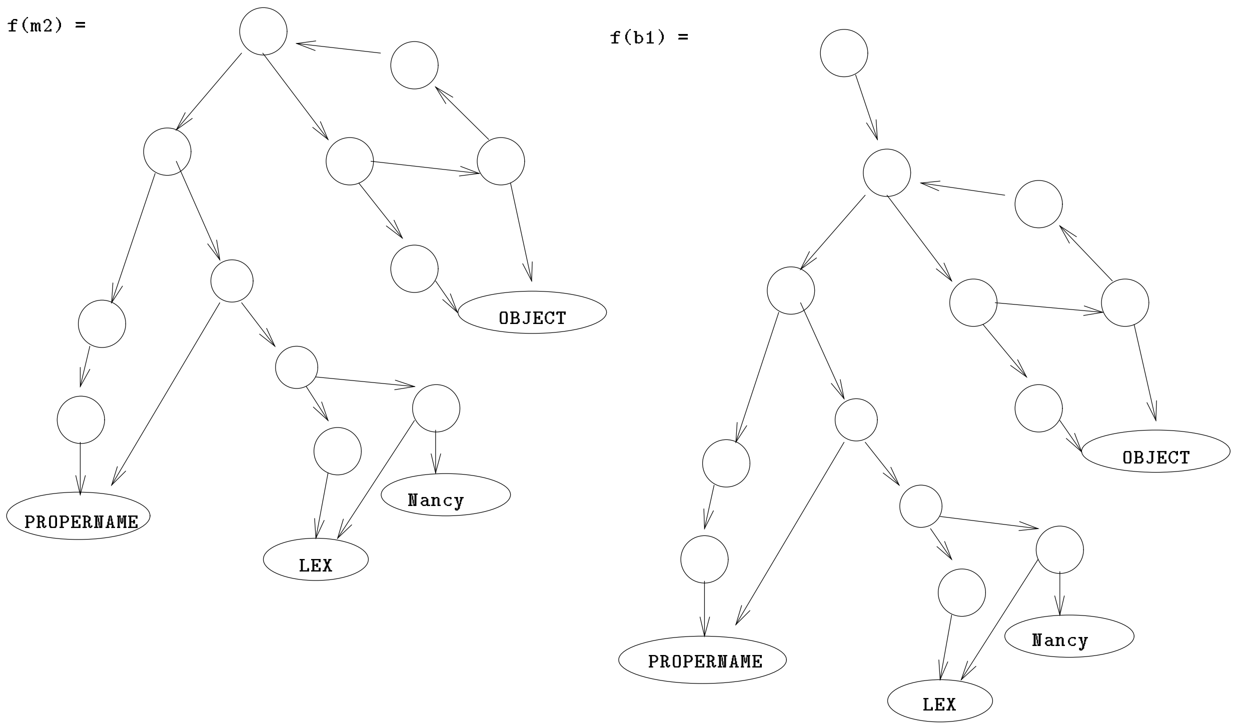


Figure 7.3: Assignments to $\mathcal{X}$ under solution f

The proposed solution uses the set

$$x = \{\{\{\texttt{PROPERNAME}\}, \{\texttt{PROPERNAME}, \{\{\{\texttt{LEX}\}, \{\texttt{LEX}, \texttt{Nancy}\}\}\}\}\}, \{\{\texttt{OBJECT}\}, \{\texttt{OBJECT}, \{x\}\}\}\}$$

in these assignments to the inderminates $\mathcal{X}$:

$$\begin{aligned} f(\texttt{m2}) &= x & (7.5) \\ f(\texttt{b1}) &= \{x\} & (7.6) \end{aligned}$$

and is shown graphically in Figure 7.3. Compare these results to those using Z^*, in Equations 4.5 and 4.6 and Figure 4.7.

Verification is tedious but straightforward:

$$\begin{aligned} f(\texttt{m2}) &= x \\ &\quad \text{(proposed solution)} \\ &= \{\{\{\texttt{PROPERNAME}\}, \{\texttt{PROPERNAME}, \{\{\{\texttt{LEX}\}, \{\texttt{LEX}, \texttt{Nancy}\}\}\}\}\}, \{\{\texttt{OBJECT}\}, \{\texttt{OBJECT}, \{x\}\}\}\} \\ &\quad \text{(definition of } x\text{)} \\ &= \{\{\{\texttt{PROPERNAME}\}, \{\texttt{PROPERNAME}, \{\{\{\texttt{LEX}\}, \{\texttt{LEX}, \texttt{Nancy}\}\}\}\}\}, \{\{\texttt{OBJECT}\}, \{\texttt{OBJECT}, f(\texttt{b1})\}\}\} \\ &\quad \text{(Equation 7.3 is verified)} \\ f(\texttt{b1}) &= \{x\} \\ &\quad \text{(proposed solution)} \\ &= \{f(\texttt{m2})\} \\ &\quad \text{(Equation 7.4 is verified)} \end{aligned}$$

We have computed the semantics of nodes in a standard SNePS network Z, not the derived Z'. Given a SNePS network C, the set of molecular nodes MOLFULL is not unstructured primitives, but consists of cablesets, sets of ordered pairs $\langle r, n\rangle$, and our solution assigns hypersets over the atoms $\mathcal{A} = \mathcal{SA} \cup \mathcal{LA} \cup \mathcal{R}$. We have a new semantic function, to be called μ_r to note the inclusion of relations:

$$\mu_r : \text{MOLFULL } (C) \cup \text{ BASE } (C) \longrightarrow \mathcal{V}_{(\mathcal{SA} \cup \mathcal{LA} \cup \mathcal{R})}$$

Even multiple arcs from one (molecular) node m to some other node are not special cases, since they must have distinct labels and therefore form distinct constituents of m. The definition of the decoration $\mathcal{D}_r$ is the same for nodes $s \in$ SENSORY, $a \in$ MOLATOM, and $b \in$ BASE (except for the use of $\mathcal{C}'$), but differs for $m \in$ MOLFULL:

Definition 7.2.1

$$\begin{aligned}
\mathcal{D}_r(s) &= tag(s) \\
\mathcal{D}_r(a) &= tag(a) \\
\mathcal{D}_r(m) &= \{\langle r, \mathcal{D}_r(n)\rangle \mid \langle r, n\rangle \in m\} \\
\mathcal{D}_r(b) &= \{\mathcal{D}_r(m) \mid \mathcal{C} \textit{ has an arc from } m \textit{ to } b\}
\end{aligned}$$

Since, for full molecular nodes, μ depends exclusively on the decoration, the definition of μ_r is essentially unchanged.

Definition 7.2.2

$$\begin{aligned}
\mu_r(m) &= \mathcal{D}_r(m) \\
\mu_r(b) &= f(b), \textit{where } f \textit{ is the solution to this system of equations:} \\
&\qquad\qquad b = \{\ \mathcal{D}_r(m) \mid b \in \mathcal{D}_r(m)\} \\
&\qquad\qquad m_i = \{\ \mathcal{D}_r(m_i)\},\ \forall m_i \ \textit{ such that } b \in \mathcal{D}_r(m_i)
\end{aligned}$$

Notice that this method turns out to be like the subnetworks induced by a set of relations, discussed in the previous section. All uses of the `PROPERNAME` relation, for example, show as arcs with heads at a single node, the atom `PROPERNAME`, as Figure 7.3 reveals. It is not the derived SNePS network S^* that has this structure, however, only the assigned hypersets.

Though complex, this enhancement of μ is significant for reasons of the integrity of the semantics. It seems obvious that the two arcs labeled `PROPERTY` and `PROPERNAME` in SR.4 and SR.5 above have something to do with the establishment of distinct meanings for their respective dominating molecular nodes. In fact, they could both occur in the same cognitive agent, as discussed in Chapter 4, since the BUILD command of SNEPSUL would not judge them to violate the Uniqueness Principle. In other words, Theorem 3, which states that $n = m \Leftrightarrow \mu(n) = \mu(m)$, would no longer have to be qualified by the exclusion of the case where n and m dominate exactly the same structure but have different arc labels. Theorem 2 would no longer hold, however, since the SNePS network itself does not show the hyperset structure rooted at nodes if relations from the arc labels are to be atoms along with the sensory nodes.

The treatment given above distinguishes SNePS from other semantic network approaches that have explicitly-named relations between nodes, but no way to build them into nodes themselves at a fundamental level. The definition of the SNePS object "wire" as a node and relation is the key here. (Of course, any semantic network treatment could have such a definition added to it.) On the other hand, the original semantic function μ of Chapter 4, which ignores arc labels in favor of node identifiers and connectivity, shows *what* participates in the meaning of a node (that is, what other nodes) without making a commitment as to *how*, and could be applied (with its handling of circularity) to any graphically structured knowledge/belief representation—even those that do not allow propositions modelled to have multiple arguments in a single position (see [Shapiro, 1991, page 138ff.] for comparison).

7.3 Acquisition, Reference, and Retention of a Concept

Let us attempt to reconsider some questions of § 1.4 in light of the foregoing analysis:

- Suppose $[\![n]\!]$ is C's concept of Perdita's old car. Suppose $[\![m]\!]$ is D's concept of Perdita's old car. What makes $[\![n]\!]$ and $[\![m]\!]$ the same thing, the same external object?
- Suppose $[\![n]\!]$ is C's concept of Hugo's dog and that this interchange takes place between C and D:

  C: She always hated Hugo's dog.
  D: But Hugo doesn't have a dog.
  C: Well, maybe it was Manuel's dog.
  D: Oh, you mean Manuel's *cat*.

  What happens to $[\![n]\!]$?

A group of sensory nodes that serve as atoms to two different cognitive agents, or to the same agent at different times, will define hypersets that have the bulk of their structure in common. Agent C's base node (or molecular node) representing Perdita's old car will be much like D's, because they assign very similar hypersets to the sight, the memory, the words, or whatever other sensory associations are in effect at the proper times.

How are later references to Perdita's old car bound to the same node? The BUILD mechanism searches to some SCOPE, just as a dictionary definition carries out its duty to some reasonable point.

Since old beliefs are not deleted, a cognitive agent retains its experience, its network grows larger, and the meaning of a node n expands over time. This is because its dominating network (corresponding to its *assertional* status) expands, of course; since arcs cannot be added, its *structural* status does not change. Whatever n was conceived as dominating is what it always dominates, even if virtually all of the logical content is overridden by qualification and contradiction due to further domination of n. Such a phenomenon changes Hugo's dog into Manuel's cat; both C and D would presumably claim that they are "talking about the same thing" throughout.

But the *content* of the past changes constantly. The node n may participate in the meaning of many new nodes, and its own contribution in terms of *meaning* will change, since it is dependent on its relation to the entire network.

It seems that part of the cognitive significance of a concept, and therefore part of the meaning of a node, should be *the circumstances under which it is acquired.* In other words, the *full* meaning of concept x to cognitive agent C will inevitably be different from the full meaning of x to cognitive agent D, unless they acquire x under exactly the same circumstances. Since that is virtually impossible, we can conclude that the respective meanings of x are different from agent to agent. This is not as outrageous as it may seem (see [Fodor and LePore, 1991] for scornful remarks), since the *full* meaning of a node is irrelevant to virtually any application, which will necessarily be limited to calculation of μ to some SCOPE short of the full semantics. In other words, C's concept of Perdita's old car and D's concept of Perdita's old car will never be *-equivalent, but they will be s-equivalent, to some SCOPE s, or δ-equivalent, to some distance of limited semantics μ^{δ}.

An ontological consequence of this principle would be that all cognitive agents are created with a context already in place. A graph-theoretic consequence of this principle would be that all SNePS networks are connected. Since the Uniqueness Principle requires that two instances of the same concept be represented by the same node, even if conceived at different times under different circumstances, incorporating this context of acquisition entails that an existing node (representing, say, the New York State Legislature) be found when the cognitive agent needs it even if it were originally acquired under quite different circumstances, and that all new contextual elements be built in with new nodes and arcs.

Accidental juxtaposition of nodes because of circumstances of acquisition may be significant. Such juxtaposition will affect the SCOPE-wise meaning of a node. But is there such a thing as "accidental juxtaposition"? SNePS has no provision for connections on that basis, nor does any developed cognitive model known to this author. But she suspects that accident and coincidence—like circularity, usually abhorred—should rather be embraced. How can the rich texture of the world's semantics be approached without it? The semantic network representaton is a good model for this, providing a natural way for, say, the scent of lilacs to be forever associated, in some mind, with certain revelations of the past because of the setting of a family reunion. It would be interesting to see how this might play out in "lifesize" examples of SNePS cognitive agents.

7.4 Conscious versus Subconscious Processing

In an avowedly intensional system such as SNePS, the distinction between transparent and opaque contexts must be known to belief revision processes, in order that assertions be updated correctly. Suppose the cognitive agent C, with separate nodes for the Morning Star and the Evening Star, learns that they are physically the same and, indeed, are both the planet Venus, which it knows as a gaseous mass. Then the correct update would also attach "is a gaseous mass" to the Morning Star and Evening Star nodes, since that predicate is a transparent context, but would not attach "was shining in the sky when we fell in love" (originally attached to the Evening Star) to the other two nodes, since that predicate is an opaque context.[1] How much, if any, of this complex processing need be conscious, or nodebased? (How much is opacity in the eye of the beholder? And should "transparent vs. opaque context" be a concept represented by a node?)

In fact, the general interaction between conscious and subconscious processing is an intriguing issue. How much does a cognitive agent know about its own reasoning? Observers of human nature agree that we use inference rules without being able to formulate them (as we seem to use natural-language grammars without "knowing" the grammar or even knowing what a grammar is); is the converse true? What are the semantic processing implications of knowing an inference rule without using it? Are there any semantic processing implications of adding nodes that represent the same concepts used in arc labels, such as `CLASS`, `PROPERTY`, `MAX`, etc.?

7.5 Assertion-Dependent Semantics for Base Concepts

Suppose we wished to consider a more personal meaning for a base node, a meaning that depends not on its location in the entire cognitive network of asserted and unasserted nodes, but its location in the web of *beliefs* held by that cognitive agent, given by asserted nodes.

Let the small SNePS network of Figure 2.1, in which nodes `m2, m7` and `m8` are asserted, the others not, be called J. Node `m6` represents the proposition that "(the concept represented by) `b2` is sweet", a proposition to which the cognitive agent modeled, CASSIE, has not committed a belief, but which CASSIE knows John to believe, as shown by `m8`. In this section, we consider a semantics for `b2` that does not include `m6` directly, on the theory that if CASSIE doesn't believe it, it is not part of the meaning, to her, of `b2`.

The approach will be to trace back from the base node to the nearest dominating asserted nodes, and treat those nodes as the members of the base node's set for purposes of the decoration and subsequent computation of μ. Applying this to the current example yields $\{$`m8`,`m7`$\}$ as the decoration of `b2`, rather than its immediate parents $\{$`m6`,`m4`$\}$. The decorations of `m7` and `m8` will, however, still include the unasserted `m6` and `m4`. What we are after is inclusion of the unasserted nodes only under the aegis, so to speak, of some beliefs held by CASSIE.

[1] Or so we claim for the purposes of this example. A *transparent* context allows substitution of equivalents, while an *opaque* context does not.

The system of equations to be solved, taking `m9` as an atom, is:

$$
\begin{aligned}
\mathtt{b2} &= \{\mathtt{m8}, \mathtt{m7}\} \\
\mathtt{m8} &= \{\mathtt{b1}, \mathtt{m3}, \mathtt{m6}\} \\
\mathtt{m7} &= \{\mathtt{b1}, \mathtt{m3}, \mathtt{m4}\} \\
\mathtt{b1} &= \{\mathtt{m2}, \mathtt{m7}, \mathtt{m8}\} \\
\mathtt{m2} &= \{\mathtt{m1}, \mathtt{b1}\} \\
\mathtt{m1} &= \{\mathtt{John}\} \\
\mathtt{m3} &= \{\mathtt{believe}\} \\
\mathtt{m6} &= \{\mathtt{m5}, \mathtt{b2}\} \\
\mathtt{m4} &= \{\mathtt{b2}, \mathtt{m9}\} \\
\mathtt{m5} &= \{\mathtt{sweet}\}
\end{aligned}
$$

Note that although `b2` does not include `m6` or `m4`, both `m6` and `m4` include `b2`. Figure 7.4 shows the distinction between the old J^* derived to show the backward arcs depicting the set membership of base nodes, and this new approach to the J^*. The base node `b1` will get the same decoration under the new approach because its parent molecular nodes are all asserted.

Formalization of belief-semantics for base nodes is accomplished by a revised definition of the decoration $\mathcal{D}$, to be called $\mathcal{D}_a$, to involve 'assertion'.

Definition 7.5.1

$$
\begin{aligned}
\mathcal{D}_a(s) &= tag(s) \\
\mathcal{D}_a(a) &= tag(a) \\
\mathcal{D}_a(m) &= \{\mathcal{D}_a(c) \mid \mathcal{S}' \textit{ has an arc from } m \textit{ to } c\} \\
\mathcal{D}_a(b) &= \{\mathcal{D}_a(m)) \mid m \in db(b) \textit{ in } \mathcal{S}'\}
\end{aligned}
$$

The function $db(n)$, for "dominating beliefs" of a node n, yields a set of molecular nodes.

Definition 7.5.2

$$db(n) = \{m \mid m \rightarrow n \textit{ and } m \textit{ asserted}\} \cup \{db(m) \mid m \rightarrow n \textit{ and } m \textit{ not asserted}\}$$

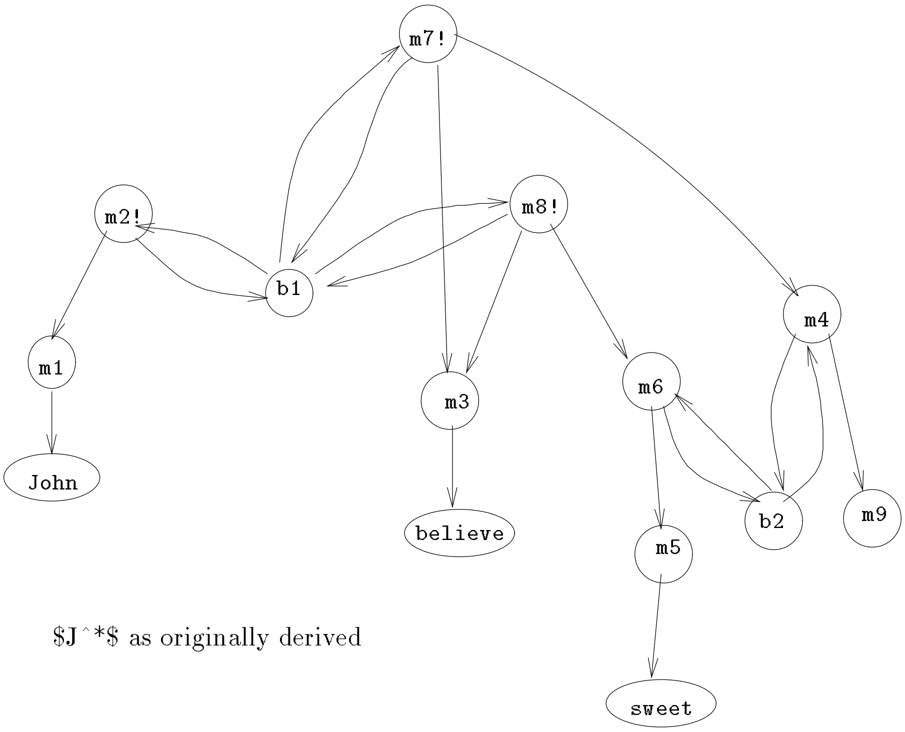
The definition of μ_a is the same as μ, except that it uses the decoration $\mathcal{D}_a$.

Definition 7.5.3

$$
\begin{aligned}
\mu_a(m) &= \mathcal{D}_a(m) \\
\mu_a(b) &= f(b), \textit{where } f \textit{ is the solution to this system of equations:} \\
&\qquad b = \{\ \mathcal{D}_a(m) \mid b \in \mathcal{D}_a(m)\} \\
&\qquad m_i = \{\ \mathcal{D}_a(m_i)\},\ \forall m_i \ \textit{ such that } b \in \mathcal{D}_a(m_i)
\end{aligned}
$$

The computation of μ_a(`b2`) yields the apg in Figure 7.5.

This semantics seems to have the properties desired, as there is no way to reach `m6` from `b2` without "going through" `m8`. CASSIE's semantics of the concept known as the girl next door does not include the idea that she is sweet, but does include the idea that John thinks she is sweet. This is certainly a viable semantics, to be favored over the assertion-independent version of Chapter 4 if semantics is to be *completely* situated, that is, dependent on only the cognitive agent's point of view.



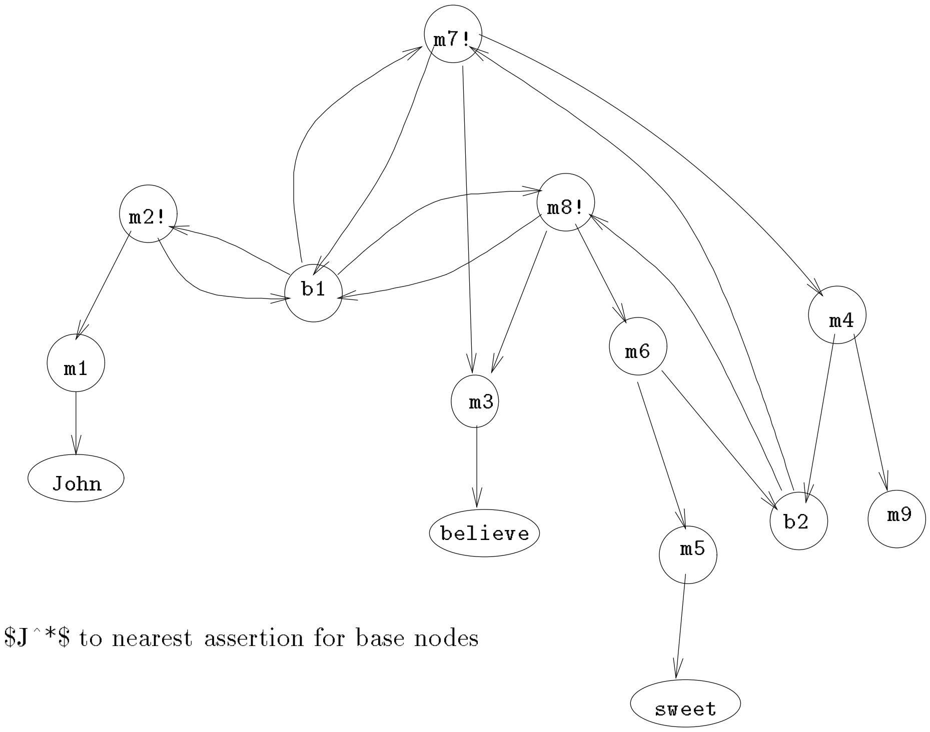


Figure 7.4: New base node semantics that incorporates nearest dominating assertion

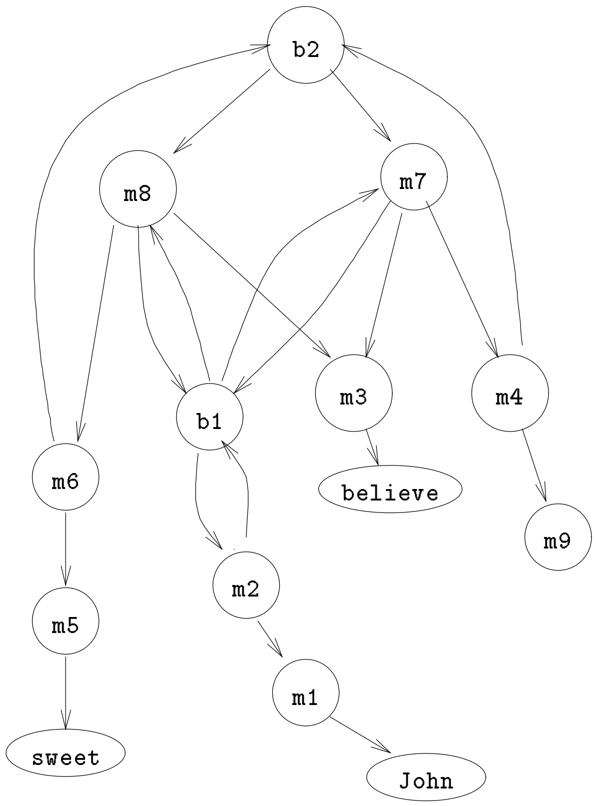


Figure 7.5: Hyperset serving as $\mu_a(\mathtt{b2})$ under assertion-dependent μ

7.6 Assertional Status at a Node vis-a-vis Meaning

A more pointed question, related to the one explored above, is whether a node should be taken to have the same meaning whether believed or not. Incorporation of the assertional operator '!' as an added facet of meaning would amount to inclusion of it as an atom in the set constituting the meaning of the node. But, as discussed above in terms of the arc labels/relations question, μ is a plain, old-fashioned, static semantics, based only on hierarchical set membership. The current theory of SNePS supports neglect of the semantic influence of assertion: "...the '!' does not affect the identity of the node, nor the proposition it represents" [Shapiro, 1991, footnote 2].

Here is the problem: Consider two agents, C and D, with two nodes, n_C and n_D, identical (or at least identical as far as the current scope is defined) except that n_C is not asserted, and n_D is. They both dominate lots of interesting structure and have complicated decorations but it ends up that $\mu(n_C) = \mu(n_D)$. Here is the obvious question, and what seems to be the right answer:

Q. Do n_C and n_D mean the same thing?
A. Yes, except that n_D is asserted.

In other words, the best answer is neither an unqualified "no" nor "yes". But any treatment that includes assertional status as part of the μ-semantics will have to incorporate it right there as part of the meanings of n_C and n_D, before their child nodes are examined, decorations are computed, etc. So the answer to Q will simply (and immediately) be:

A. No.

This violates what we mean by "meaning". We could compute the meaning and then handle the assertional question, but that leaves μ just as it is now. The answer to Q, in other words, that is currently provided by μ is:

A. Yes. And you want to know their respective assertional statuses? Look elsewhere.

Assertional status of an arbitrary node n would be important to $\mu(n)$ if assertional status affected the computation of $\mu(m)$ for some other node m connected to m. But to make the semantic mechanism sensitive to the current knowledge/belief status of the agent would be something of a violation of intensionality, insofar as it might limit the first-class standing of a concept. So μ, as developed here, has nothing to offer in response to Question 2 in § 1.4.

Nevertheless, SNePS researchers may not feel comfortable ignoring assertion in the computation of the semantics μ, as they might not at ignoring arc labels. If it is not appropriate to build in assertional status of a node through hypersets, there remains another way, highly SNePS-compatible: Conceptualize the assertional mechanism. Use nodes to do assertion. How? Here are several possible degrees, under the SNePS paradigm.

1. Consider a node *standing for* 'belief'. Is it necessary? A cognitive agent should not be required to "know" about *belief*, the counterpart of *assertion*, in order to have beliefs. Although it is a fundamental notion, there is no evidence that cognitive agents must have that notion before they can assert propositions. In SNePS, the necessary reasoning is provided by path-based inference [Shapiro, 1991], which does not rely on any conceptualization of its mechanism.

   But human cognitive agents do eventually acquire the notion of belief, presumably as a node, while entertaining actual beliefs, asserted propositions, also. Shouldn't the full meaning of that `BELIEF` node somehow take into account those assertions? The most telling argument that a node for 'belief' must exist in a (fully-developed) cognitive agent is that a cognitive agent "must be able to represent other cognitive agents, both as objects and agents of belief" [Shapiro and Rapaport, 1991, page 217].

2. Now consider a node *labeled* 'belief'. Is the lexical assignment necessary? By the Uniqueness Principle, if there is a node standing for 'belief' *at all*, it must be the one to which the label (via **LEX** arc) is attached. Can we allow conceptualization before the naming of it? Cognitive verisimilitude would entail that a cognitive agent have a node for the concept 'belief' independently of the acquisition of the term. Note that later lexical assignment cannot be done with the addition of the **LEX** arc; mechanisms such as **EQUIV/EQUIV** must suffice. The semantics μ is no different in this regard from any other semantics defined over nodes. The assignment (somehow) of the word 'belief' to the node affects the meaning only insofar as it then encompasses the word in addition to all pre- existing elaboration.

3. Now consider a node *performing* belief. In other words, if assertion is to be conceptualized as a node, shouldn't there be some relationship between it and node-based reasoning? And even between it and path-based reasoning? Else where would path-based inference come from, given that its placement in the mind of the cogitive agent would make it a node, and given that nodes are the only raw material available? And, of course, if it (or any component) is a node, then by the Uniqueness Principle, that node is also the locus of any concepts *about* belief.

4. For a possibility that is speculative well beyond the current SNePS paradigm, consider a SNePS mechanism that spontaneously *creates its own node (or other mechanism) for belief*, independently of any sensory data input. A truly intelligent system would seem to call for it, but such a capacity is not to be considered here.

Assertion grounds inference and the establishment of a commitment to propositions. Only in the context of such research in SNePS, not here, can the primary question of Fodor and LePore be addressed—whether the three properties (1) the inferential role of meaning, (2) the rejection of the analytic/synthetic distinction, and (3) compositionality, can be maintained simultaneously under the semantics μ.

7.7 Full Location Incorporation

The semantic function μ puts circularity, or mutual semantic influence strictly between a base node and its parent, leaving the semantics of a molecular node more or less dependent only on the nodes it dominates, the exception being that a base node it dominates may well include another parent molecular node in its semantics, thereby causing the semantics of the first molecular node to include the second even though neither dominates the other. An obvious extension would have all nodes influence the meanings of their parents, as would be depicted by the addition of backward arcs everywhere except between sensory nodes and their parent nodes. Let us call this semantics μ^+. As a variation on the semantic structure of the TABLE sentence shown in Figure 4.1, the result of this is shown in Figure 7.6. Here, clearly, all nodes participate in the semantics of all other nodes, and the meaning of a node depends in a direct way on its location in the entire network. The special status of base nodes is lost, and the question whether one node is a member (at some level) of the hyperset assigned to another node is vacuous, the only question remaining being the level, or distance, of that membership. The atoms are still the sensory nodes, and the lack of membership arcs between them and their parents prevents the entire network from being a picture of Ω. Figure 7.7 shows the hyperset assigned to the node `b2` under μ^+ (to the same restricted scope as W^* of Figure 4.5, so this $f($`b2`$)$ should be compared to that shown in Figure 4.8).

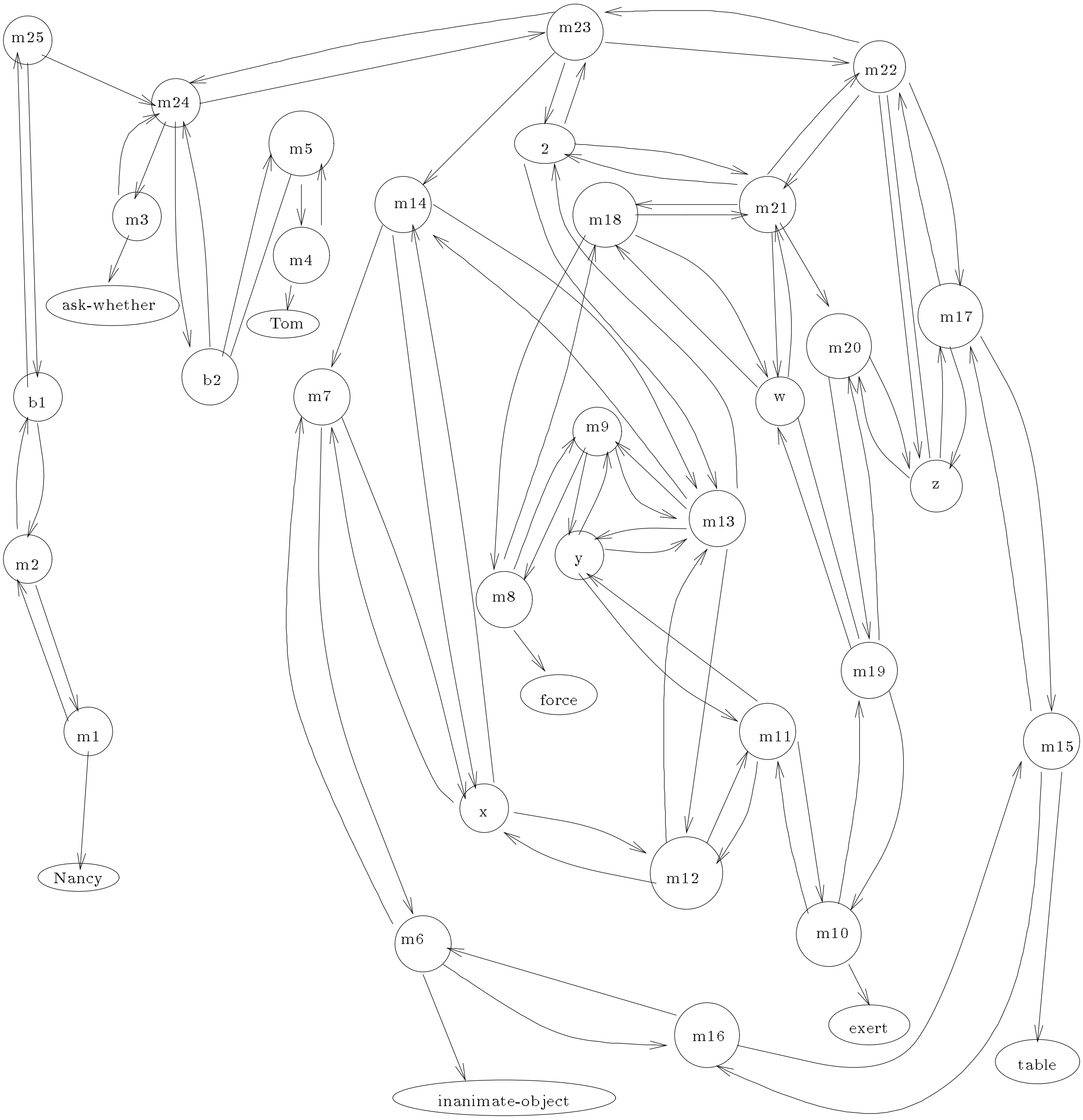


Figure 7.6: TABLE sentence showing semantics of nodes under μ^+

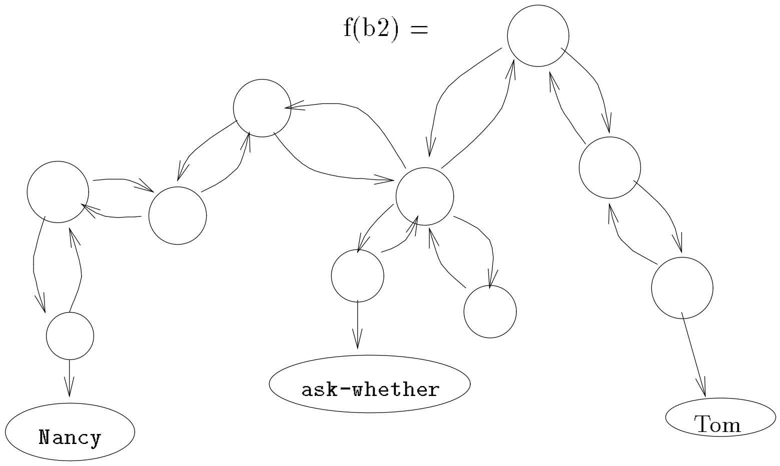


Figure 7.7: Assignment to indeterminate `b2` under μ^+

7.8 Conclusions

There are reasonable alternatives to the “plain” non-well-founded set semantics μ of Chapter 4 that incorporate relations (used as arc labels), that depend on the beliefs of the cognitive agent, and that include the *entire* surrounding network in the meaning. Unresolved issues include the degree of influence of context on meaning and the distinction between directed, or conscious, processing of new nodes and involuntary, or unconscious, acquisition.

The best SNePS semantics based on non-well-founded set theory may well be some combination of the enhancements noted here, to be determined by further research.

Chapter 8

Conclusions

"Reason, or the ratio of all we have already known, is not the same that it shall be when we know more."

— William Blake, 1788[1]

8.1 Introduction

In this chapter, the major points of the work are reviewed and summarized. Additional suggestions for research are offered, especially to clarify the relationship of SNePS with μ to other issues and theories in artificial intelligence.

8.2 A Formal Semantics for SNePS

SNePS needs a semantics for base nodes and a theoretical foundation providing the circularity in its semantics that is referred to in [Shapiro and Rapaport, 1991, pages 221–222]. The non-well-founded set theory of Peter Aczel offers a rigorous mathematical development of set-theoretic objects that may contain themselves. The semantic function μ offered herein provides a semantics for SNePS, based on non-well-founded set theory, which incorporates the notion of circularity in base nodes. Feasibility of actual processing is provided by μ^{δ}, which renders nodes out of reach of the search area atomic, thereby incorporating them in the computation of μ-semantics as placeholders whose values are not known, or perhaps not yet known.

The expression of the semantic value of a node in SNodes as a hyperset over some set of atoms $\mathcal{A}$ is its expression in terms of complex combinations of direct and indirect references to the particular sensory data that make up the world as it appears to the cognitive agent. The μ-value of a node for *grandmother* will be a set **not** of concepts such as *grandparents'-house*, *lilac-eau-de-cologne*, and *blood-being-thicker-than-water*, but a set of the sensory stimula that count—the sight of the house, the voice, the scent of lilacs, the feeling of shock at hearing of a death in the family, and so forth. There remain no identifiers or other artificial references to intermediate structures.

There is much more to investigate in μ, such as formulation of its solutions as the greatest fixed points of some operator using *coinduction*, the non-well-founded set counterpart of induction. A version of Aczel's Special Final Coalgebra Theorem should be developed specifically for SNePS. [Barwise and Etchemendy, 1987, page 56 ff.]

The reader in search of an appealing *informal* semantics will be disappointed. Hypersets derived from SNePS networks seem even farther from English glosses or other paraphrases than are the

[1] *MINOR PROPHECIES*, There Is No Natural Religion II

SNePS networks themselves. As discussed in the introductory chapter, however, that closeness between semantic networks and human expression is misleading in that it implies the successful crossing of a bridge that has not yet been built. The semantic investigation culminating in μ is meant to provide some foundation for it.

8.3 Legitimizing Circularity in Semantics

Shapiro and Rapaport, after describing the circularity of SNePS, refer to sensory nodes as "a major escape from this circularity" [Shapiro and Rapaport, 1991, page 222]. Evidently, circularity is to be acknowledged but held in check, a reasonable attitude. As Smith puts it, "Making interpretation dependent on use, at least at first blush, therefore gives one every reason to suppose that the notion of soundness is rendered circular, hence vacuous" [Smith, 1991, page 266]. The first challenge, then, to any semantic theory that embraces circularity is to counteract the instinct to fear it as destructive.

The problem with circularity is the lack of a basis for computation. To say that a set is equal to itself, is Ω, is to say that no other statement can be made about that set; it can have no other properties or attributes or membership predicates. To say, on the other hand, that a set contains *itself and something else* is to have circularity, but saved from viciousness. It is possible to make meaningful claims about such sets—which claims are the material of Aczel's theory, of course. The semantic function μ gives us circularity in the form of mutual influence between a molecular and base node, and grounding in the form of sensory data as atoms. What it does not give is a full explanation of cognitive phenomena like "thoughts going around in circles".

8.4 Graphical Semantic Models

In §1.3.2, we saw that Lenhart Schubert calls for a knowledge representation scheme based on semantic networks that is qualitatively different from the predicate calculus. It would make "essential use of nontrivial graph-theoretic properties" [Schubert, 1991, page 106]. Note that his complaint applies to well-founded semantic networks, insofar as their semantics depends on recursive enumerability. But we now have a difference in meaning that depends on "genuine structural difference" between graphs, since only such a difference suffices to distinguish one hyperset from another [Barwise and Etchemendy, 1987, page 40].

As for furthering of the theory of SNePS itself, the only result that my come as a surprise to SNePS researchers is Axiom 4:

> Given a network $S \in$ SNets, every molecular node $m \in$ MOLFULL(S) heads a semipath $\langle m, n_1\rangle, \langle n_1, n_2\rangle, \ldots \langle n_k, t\rangle$, where $t \in$ SENSORY(S) and for every arc $\langle n_i, n_{i+1}\rangle$, either (1) there is an arc from n_i to n_{i+1} in S, or (2) $n_i \in$ BASE(S) and there is an arc from n_{i+1} to n_i in S.

Its requirement that base and molecular nodes be assembled to allow a certain type of semipath to sensory nodes may affect further development of case frames and other structural properties.

8.5 A Limited Semantics

Along with the informal mechanism called SCOPE, in which an arbitrary subnetwork of a SNePS network is delineated as the context of μ, a variant that computes semantics to a given semipath length δ was defined as the function μ^{δ}. Some semantic network researchers view the measure of distance between nodes (number of arcs, or, equivalently, length of the nrn-path) as suspect, and any attribution of significance to that distance as a misuse of the network model. (See, for example, [Brachman *et al.*, 1985, pages 415–416].) But μ^δ explicitly makes this obvious notion of distance between nodes into a feature of the semantic theory. Any node n that includes two others c_1 and c_2, that are at different distances away, will have that distance reflected in the nesting of membership of c_1 and c_2 in n. For example, in Figure 2.1, $\mu(\texttt{m8})$ includes both `m6` and, through `b2`, `m4`, but at different "generations of descendants" of hereditary membership:

$$\mu(\texttt{m8}) = \{\ldots\texttt{m6},\ldots\{\{\texttt{m4}\}\},\ldots\}$$

The nesting shows that the distance from `m8` to `m6` is one, and the distance from `m8` to `m4` is three. This way of incorporating distance is not grafted on independently of the semantics μ, but is inherent in the hereditary set approach.

The present work does not offer any argument for the utility of such a numerical measurement of semantic influence, but the material is there. A visual property of graphical models (semantic networks), which may have been dismissed as accidental, may in this way contribute to its disciplined use in cognitive research. There is an obvious analogue to connectionist theories here, which should be explored in further research. Perhaps SNePS with μ^δ is a static version of the phenomenon manifested dynamically in connectionist network processing, spreading activation.

8.6 Compositionality

The function μ is compositional, but with some reservations. The semantics of an expression (a member of SNodes) is indeed computable from the semantics of its parts and the way they are put together, but the semantics of some parts have to be redefined. This can be done without obvious violation of the principle of compositionality by simply defining the semantic combination rules so that they do so. No means of providing circularity (in base nodes, to which further concepts can be attached) would escape that necessity.

Little has been said about the "semantic interpretation" rules SI.i of [Shapiro and Rapaport, 1987], which offer a semantics of SNodes as English glosses. They are, by contrast, strictly compositional in that no interpretation done on a node is affected by the addition of others. (They do not provide circularity.)

8.7 Intensionality

Semantics based on non-well-founded sets supports intensionality by ensuring that different concepts, represented as mandated by different nodes, also have different meanings. The Uniqueness Principle for μ shows that the meanings of no two distinct nodes, construed as hypersets under μ, are the same. Furthermore, no two distinct hypersets can be assigned to the same node. If their meanings are closely related in some way, however, two distinct nodes will have a great deal of hyperset composition in common. Their semantic influence on other nodes will be close, therefore, and furthermore, the difference between their semantic contributions diminishes, relative to other contributions, as δ increases in μ^δ.

8.8 Significance for Artificial Intelligence

Theorem 2 makes a virtue of necessity in acknowledging that, except for its addition of virtual semantic arcs extending from base nodes to dominating molecular nodes, μ adds nothing to the SNePS network that was not already visible. The semantic value is more or less the syntactic value, thinly disguised. Thus this research is open to the charge often leveled at artificial intelligence, that some piece of work is a futile omphaloskeptic exercise—a charge with some merit.

Such refutation as can be made is left largely to the discussion of contributions above, but since it is Smith who calls for AI research that positions itself properly along the spectrum of possible attacks on its problems, it is is fitting to end by reviewing his twelve questions for those which are affected by the semantics μ for SNePS. Recall that the answers for EC ("embedded computation") are what he considers to be the goals of knowledge representation through AI.

1. **Does the system focus primarily on explicit representation?** [EC—No.] The Uniqueness Principle of SNePS establishes a certain explicitness of representation, in that concepts and nodes are in one-to-one correspondence. It is not entirely clear (to Smith *or* his reader) what types of representation are not explicit, but the semantics of SNePS nodes as possibly infinite sets may qualify.

2. **Is representation content contextual (situated)?** [EC—Yes.] No change; SNePS cognitive agents were already known to rely on a point of view.

3. **Does meaning depend on use?** [EC—Yes.] Recall that Smith's dynamic notion of meaning "can't be separated from the whole complex of inferential, conversational, social, and other purposes to which it is put". That is certainly true of $\mu(n)$ for any SNePS node n, through its inclusion of sensory nodes as atoms. If "use" can be construed as the incorporation of other meanings, eventually dependent on sensory data, then "Yes". If "use" requires a more active cognitive agent than can be embodied in a semantic network diagram, then "No".

4. **Is consistency mandated?** [EC—No.] No change; a cognitive agent may still have logically contradictory intensions.

5. **Does the system use a single representational scheme?** [EC—No.] Well ...yes. According to μ, SNePS does indeed have a single representational scheme; all nodes stand for hypersets. On the other hand, his examples of phenomena that resist capture by the Lenat and Feigenbaum frame-and-slot system—mass nouns, plurals, images [Srihari and Rapaport, 1990; Cho, 1992]—are easily captured as nodes in SNePS, with any peculiarity of property provided by, of course, connection to other nodes capturing those peculiarities.

6. **Are there only discrete propositions (no continuous representation, images, ...)?** [EC—No.] Sets are relentlessly discrete objects, with membership a relentlessly binary predicate. Hypersets, however, with the ability to contain infinite objects, may be regarded as analogous to infinite decimal representations of real numbers using the standard discrete objects, the ten digits.

7. **Do the representations capture all that matters?** [EC—No.] Does SNePS with μ meet this standard or not? Is it supposed to? According to Smith,

   > ...the full significance of an intentional action (not just a communicative one) can crucially involve *non-representational phenomena*, as well as representational ones. [This] is a claim that the millenial story about intelligence won't consist solely of a story about representation, but will inevitably weave that story together with analyses of other non-representational aspects of an intentional agent. [Smith, 1991, page 273]

   It appears that the "right" answer to this question would only be provided by a proof that there are phenomena of intelligence that SNePS with μ not only does not, but *cannot*, capture. Insofar as SNePS with μ is a cognitive representation scheme, its job is to do all and only representation, and the answer is "Not applicable". Insofar as SNePS with μ is a model of intelligence in the large, the answer is beyond the scope of the present inquiry.

8. **Are reasoning and inference central?** [EC—Yes.] Yes. No change, as the assertional mechanism, though important, is excluded from μ.

9. **Are participation and action crucial?** [EC—Yes.] This research really sheds no light on that question (but see the following one). Incorporation of fully-defined *act* nodes into the set subject to μ will, however. The door has been left open in a way described by Smith in his discussion of question 3: "...much of the structure of argument and discourse—even the *raison d'être* of rationality—involves negotiating in an intentional space where meanings are left fluid by our linguistic and conceptual schemes, ready to be grounded in experience." [Smith, 1991, page 267] Grounding in sensory data, and soon, it is to be hoped, in act nodes, is an important feature of μ.

10. **Is physical embodiment important?** [EC—Yes.] It is sensory nodes that serve as atoms over which to form hypersets that become the meanings of the nodes. The semantics in terms of the values returned by μ are utterly dependent on sense data. We can have well-defined hypersets without atoms, however, when δ does not "reach" any sensory nodes for some use of μ^δ, so the μ^δ mechanism is still meaningful without any manifestions of physical embodiment. Both this question and the previous one, however, must be qualified by the possibility that Smith wants something captured that is, by definition, out of reach of the sensory acquisition of knowledge.

11. **Does the system support "original" semantics?** [EC—Yes.] For SNePS with μ, very much so. Nodes have μ-semantics without any interpretation from outside the agent, indeed, without any outside interpreter at all.

12. **Room for a divergence between the representational capacities of theorist and agent?** [EC—Yes.] No change. Still clearly "yes", as such independence was always a principle of SNePS.

The author hopes that SNePS with μ provides a footing on Smith's middle ground of "intermediating conceptual structure" in AI research, fostering further work there.

Bibliography

[Aczel, 1988] Peter Aczel. *Non-Well-Founded Sets.* CSLI Lecture Notes; Number 14. Center for the Study of Language and Information, Stanford, California, 1988.

[Barwise and Etchemendy, 1987] Jon Barwise and John Etchemendy. *The Liar: An Essay on Truth and Circularity.* Oxford University Press, New York, New York, 1987.

[Barwise, 1989] Jon Barwise. *The Situation in Logic.* Center for the Study of Language and Information, Leland Stanford Junior University; Stanford, California, 1989.

[Brachman and Levesque, 1985] Ronald J. Brachman and Hector J. Levesque, editors. *Readings in Knowledge Representation.* Morgan Kaufmann, Los Altos, California, 1985.

[Brachman and Schmolze, 1985] Ronald J. Brachman and James G. Schmolze. An overview of the KL-ONE knowledge representation system. *Cognitive Science*, 9:171–216, 1985.

[Brachman *et al.*, 1985] Ronald J. Brachman, Richard E. Fikes, and Hector J. Levesque. KRYPTON: A functional approach to knowledge representation. In Ronald J. Brachman and Hector J. Levesque, editors, *Readings in Knowledge Representation*, pages 411–429. Morgan Kaufmann Publishers, Inc., Los Altos, California, 1985.

[Brassard and Bratley, 1988] Gilles Brassard and Paul Bratley. *Algorithmics: Theory and Practice.* Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1988.

[Cantor, 1952] Georg Cantor. *Contributions to the Founding of the Theory of Transfinite Numbers.* The Open Court Publishing Company, La Salle, Illinois, 1952. Work of 1895–1897, translated by Philip E. B. Jourdain.

[Cho, 1992] Sung-Hye Cho. Collections as intensional entities and their representations in a semantic network. In *Proceedings of the Second Pacific Rim International Conference on Artificial Intelligence*, pages 388–394, 1992.

[Dionne *et al.*, 1992] Robert Dionne, Eric Mays, and Frank J. Oles. A non-well-founded approach to terminological cycles. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 761–766, Menlo Park, California, 1992. American Association for Artificial Intelligence, AAAI Press.

[Ehrlich and Rapaport, 1992] Karen Ehrlich and William J. Rapaport. Automatic acquisition of word meanings from natural-language contexts. Technical Report 92–03, Center for Cognitive Science, SUNY at Buffalo, 1992.

[Encyclopedia Americana, 1980] Encyclopedia americana. Americana Corporation; Danbury, Connecticut, 1980. Entry for “Mentor”.

[Fodor and LePore, 1991] Jerry Fodor and Ernest LePore. Why meaning (probably) isn't conceptual role. *Mind & Language*, 6(4):328–343, 1991.

[Gamut, 1991] L. T. F. Gamut. *Logic, Language, and Meaning; Volume II: Intensional Logic and Logical Grammar.* University of Chicago Press, Chicago, Illinois, 1991.

[Garey and Johnson, 1979] Michael R. Garey and David S. Johnson. *Computers and Intractability.* W. H. Freeman and Company, New York, 1979.

[Gibbons, 1985] Alan Gibbons. *Algorithmic Graph Theory.* Cambridge University Press, Cambridge, Great Britain, 1985.

[Grandy, 1990] Richard E. Grandy. Understanding and the principle of compositionality. In James E. Tomberlin, editor, *Philosophical Perspectives Vol. 4: Action Theory and Philosophy of Mind*, pages 557–572. Ridgeview, Atascadero, California, 1990.

[Harary, 1972] Frank Harary. *Graph Theory.* Addison-Wesley Publishing Company, Reading, Massachusetts, 1972.

[Hirst, 1991] Graeme Hirst. Existence assumptions in knowledge representation. *Artificial Intelligence*, 49:199–242, 1991.

[Hull, 1986] R. G. Hull. A new design for SNIP the SNePS inference package. SNeRG Technical Note 14, Department of Computer Science, SUNY at Buffalo, 1986.

[Jacobson, 1974] Nathan Jacobson. *Basic Algebra I.* W. H. Freeman and Company, San Francisco, 1974.

[Kumar and Shapiro, 1991] Deepak Kumar and Stuart C. Shapiro. Architecture of an intelligent agent in SNePS. *SIGART Bulletin*, 2(4):89–92, August 1991.

[Kumar *et al.*, 1988] Deepak Kumar, Syed Ali, and Stuart C. Shapiro. Discussing, using and recognizing plans in SNePS preliminary report—SNACTor: An acting system. In P V S Rao and P Sadanandan, editors, *Modern Trends in Information Technology: Proceedings of the Seventh Biennial Convention of South East Asia Regional Computer Confederation*, pages 177–182. Tata McGraw-Hill, New Delhi, India, 1988.

[Kunen, 1980] Kenneth Kunen. *Set Theory: An Introduction to Independence Proofs*, volume 102 of *Studies in Logic and the Foundations of Mathematics.* North-Holland Publishing Company, New York, 1980.

[Lacey, 1976] A. R. Lacey. *A Dictionary of Philosophy.* Routledge and Kegan Paul, London, 1976.

[Lenat and Feigenbaum, 1991] D. B. Lenat and E. A. Feigenbaum. On the thresholds of knowledge. *Artificial Intelligence*, 47:200–250, 1991.

[Levy, 1979] Azriel Levy. *Basic Set Theory.* Springer Verlag, Berlin, 1979.

[Linguist List, 1991] Linguist list. Electronic bulletin board in mailing list form, September 1991. Vol-2-507, Vol-2-514, Vol-2-523.

[Maida and Shapiro, 1982] A. S. Maida and S. C. Shapiro. Intensional concepts in propositional semantic networks. *Cognitive Science*, 6(4):291–330, 1982. Reprinted in R. J. Brachman and H. J. Levesque, eds. *Readings in Knowledge Representation*, Morgan Kaufmann, Los Altos, CA, 1985, 170–189.

[Martins and Shapiro, 1988] J. P. Martins and S. C. Shapiro. A model for belief revision. *Artificial Intelligence*, 35(1):25–79, 1988.

[Maurer and Ralston, 1991] Stephen B. Maurer and Anthony Ralston. *Discrete Algorithmic Mathematics*. Addison-Wesley Publishing Company, Reading, Massachusetts, 1991.

[Minsky, 1981] Marvin Minsky. A framework for representing knowledge. In J. Haugeland, editor, *Mind Design*, pages 95–128. The MIT Press, Cambridge, Massachusetts, 1981. As printed in Brachman and Levesque: 245–262.

[Morgado, 1986] Ernesto J. M. Morgado. Semantic networks as abstract data types. Technical Report 86–19, Department of Computer Science, SUNY at Buffalo, 1986.

[Nebel, 1991] Bernhard Nebel. Terminological cycles: Semantics and computational properties. In John F. Sowa, editor, *Principles of Semantic Networks*, chapter 11, pages 331–361. Morgan Kaufmann Publishers, Inc., San Mateo, California, 1991.

[Partee *et al.*, 1990] Barbara H. Partee, Alice ter Meulen, and Robert E. Wall. *Mathematical Methods in Linguistics*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1990.

[Partee, 1984] Barbara Partee. Compositionality. In F. Landman and F. Veltman, editors, *Varieties of Formal Semantics*. Foris, Dordrecht, 1984.

[Quillian, 1968] M. Ross Quillian. Semantic memory. In Marvin Minsky, editor, *Semantic Information Processsing*, pages 227–270. The MIT Press, Cambridge, Massachusetts, 1968.

[Rapaport, 1978] William J. Rapaport. Meinongian theories and a Russellian paradox. *Noûs*, 12:153–180, 1978. Errata in Vol. 13.

[Rapaport, 1981] William J. Rapaport. How to make the world fit our language: An essay in Meinongian semantics. *Grazer Philosophische Studien*, 14:1–21, 1981.

[Rapaport, 1985a] * W. J. Rapaport. Logic and artificial intelligence. Technical Report 85–16, Department of Computer Science, SUNY at Buffalo, 1985.

[Rapaport, 1985b] W. J. Rapaport. Meinongian semantics for propositional semantic networks. In *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics*, pages 43–48, Morristown, NJ, 1985. Association for Computational Linguistics.

[Rapaport, 1986] W. J. Rapaport. Logical foundations for belief representation. *Cognitive Science*, 10:371–422, 1986.

[Rapaport, 1988a] W. J. Rapaport. A knowledge-representation challenge for SNePS. SNeRG Technical Note 20, Department of Computer Science, SUNY at Buffalo, 1988.

[Rapaport, 1988b] W. J. Rapaport. Syntactic semantics: Foundations of computational natural-language understanding. In J. Fetzer, editor, *Aspects of Artificial Intelligence*, pages 81–131. Kluwer Academic Publishers, Dordrecht, Holland, 1988.

[Rapaport, 1992a] W. J. Rapaport. Logic, predicate. In *Encyclopedia of Artificial Intelligence, Second Edition*, pages 866–873. John Wiley and Sons, Inc., 1992.

[Rapaport, 1992b] W. J. Rapaport. Logic, propositional. In *Encyclopedia of Artificial Intelligence, Second Edition*, pages 891–897. John Wiley and Sons, Inc., 1992.

[Rapaport, forthcoming] William J. Rapaport. Meinongian semantics and artificial intelligence. In P. Simons, editor, *Essays on Meinong*. Philosophia Verlag, forthcoming.

[Schank and Rieger, 1974] Roger C. Schank and Charles J. Rieger. Inference and the computer understanding of natural language. *Artificial Intelligence*, 5:373–412, 1974. As reprinted in Brachman and Levesque, 1985: 120–139.

[Schubert, 1991] Lenhart K. Schubert. Semantic nets are in the eye of the beholder. In *Principles of Semantic Networks*, pages 95–107. Morgan Kaufmann Publishers, Inc., 1991.

[Shapiro and Group, 1989] S. C. Shapiro and The SNePS Implementation Group. *SNePS-2 User's Manual*. Department of Computer Science, SUNY at Buffalo, 1989.

[Shapiro and Rapaport, 1987] S. C. Shapiro and W. J. Rapaport. SNePS considered as a fully intensional propositional semantic network. In N. Cercone and G. McCalla, editors, *The Knowledge Frontier*, pages 263–315. Springer–Verlag, New York, 1987.

[Shapiro and Rapaport, 1991] Stuart C. Shapiro and William J. Rapaport. Models and minds: Knowledge representation for natural-language competence. In Robert Cummins and John Pollock, editors, *Philosophy and AI: Essays at the Interface*, pages 215–259. MIT Press, Cambridge, MA, 1991.

[Shapiro *et al.*, 1982] S. C. Shapiro, J. Martins, and D. McKay. Bi-directional inference. In *Proceedings of the Fourth Annual Conference of the Cognitive Science Society*, pages 90–93, Ann Arbor, MI, 1982. the Program in Cognitive Science of The University of Chicago and The University of Michigan.

[Shapiro, 1978] S. C. Shapiro. Path-based and node-based inference in semantic networks. In D. Waltz, editor, *Tinlap–2: Theoretical Issues in Natural Languages Processing*, pages 219–225, New York, 1978. ACM.

[Shapiro, 1979] S. C. Shapiro. The SNePS semantic network processing system. In N. V. Findler, editor, *Associative Networks: The Representation and Use of Knowledge by Computers*, pages 179–203. Academic Press, New York, 1979.

[Shapiro, 1991] Stuart C. Shapiro. Cables, paths and "subconsious" reasoning in propositional semantic networks. In John F. Sowa, editor, *Principles of Semantic Networks*, chapter 4, pages 137–156. Morgan Kaufmann, San Mateo, CA, 1991.

[Siskind, 1990] Jeffrey Mark Siskind. Acquiring core meanings of words, represented as jackendoff-style conceptual structures, from correlated streams of linguistic and non-linguistic input. In *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 1990.

[Smith, 1982] Brian Cantwell Smith. Reflection and semantics in a procedural language. Technical Report MIT/LCS/TR-272, MIT Laboratory for Computer Science, 1982. Prologue reprinted in Brachman and Levesque, 1985: 32–39.

[Smith, 1987] Brian Cantwell Smith. The correspondence continuum. Technical Report CSLI–87–71, Center for the Study of Language and Information, Stanford University, 1987.

[Smith, 1991] Brian Cantwell Smith. The owl and the electric encyclopedia. *Artificial Intelligence*, 47:251–258, 1991.

[Sowa, 1984] John F. Sowa. *Conceptual Structures: Information Processing in Mind and Machine*. Addison Wesley, Reading, Massachusets, 1984.

[Srihari and Rapaport, 1990] R. K. Srihari and W. J. Rapaport. Exacting visual information from text: Using captions to label human faces in newspaper photographs. In D. Kumar, editor, *Current Trends in SNePS–Semantic Network Processing System: Proceedings of the First Annual SNePS Workshop*, pages 85–96, Buffalo, NY, 1990. Springer-Verlag.

[Stoy, 1977] Joseph E. Stoy. *Denotational Semantics: The Scott-Strachey Approach to Programming Language Theory.* The MIT Press, Cambridge, Massachusetts, 1977.

[van der Waerden, 1991] B. L. van der Waerden. *Algebra: Volume I.* Springer Verlag, New York, 1991. Seventh Edition.

[Wilks, 1984] Yorick Wilks. Is Frege's principle trivial or false? In *Proceedings of the Annual Conference of the Linguistics Association of Great Britain.* Linguistics Association of Great Britain, 1984.

[Woods, 1975] William A. Woods. What's in a link: Foundations for semantic networks. In Daniel G. Bobrow and A. M. Collins, editors, *Representation and Understanding*, pages 35–82. Academic Press, 1975. Reprinted in Brachman and Levesque, pages 217–241.