ABDUCTIVE INTERPRETATION AND REINTERPRETATION OF NATURAL LANGUAGE UTTERANCES

by

Susan W. McRoy

Department of Computer Science
University of Toronto
Toronto, Ontario, Canada

A Thesis submitted in conformity with the requirements
for the degree of Doctor of Philosophy
in the Department of Computer Science, University of Toronto,
July 1993

Abstract

To decide how to respond to an utterance, a speaker must interpret what others have said and why they have said it. Speakers rely on their expectations to decide whether they have understood each other. *Misunderstandings* occur when speakers differ in their beliefs about what has been said or why. If a listener hears something that seems inconsistent, he may reinterpret an earlier utterance and respond to it anew. Otherwise, he assumes that the conversation is proceeding smoothly. Recognizing an inconsistency as a misunderstanding and generating a new reply together accomplish what is known as a *fourth-position repair*.

To model the repair of misunderstandings, this thesis combines both intentional and social accounts of discourse, unifying theories of speech act production, interpretation, and repair. In intentional accounts, speakers use their beliefs, goals, and expectations to decide what to say; when they interpret an utterance, speakers identify goals that might account for it. In sociological accounts provided by Ethnomethodology, discourse interactions and the resolution of misunderstandings are normal activities guided by social conventions. The approach extends intentional accounts by using expectations deriving from social conventions in order to guide interpretation. As a result, it avoids the unconstrained inference of goals that has plagued many models of discourse. A unified theory has been developed by using default reasoning to generate utterances and using abduction to characterize interpretation and repair.

The account has been expressed as a logical theory within the Prioritized Theorist Framework. The theory includes relations on linguistic acts and the Gricean attitudes that they express. It also contains an axiomatization of speakers' knowledge for generating socially appropriate utterances and for detecting and repairing misunderstandings. The generality of the approach is demonstrated by re-enacting real conversations using the theorem-proving capabilities of Prioritized Theorist.

Acknowledgements

Firstly, I thank my supervisor, Graeme Hirst, for his support and guidance throughout this thesis work.

I also thank my external appraiser, James Allen of the University of Rochester, my external examiner, James Heap of Ontario Institute for Studies in Education, and the members of my committee from the Department of Computer Science at the University of Toronto: Derek Corneil, Hector Levesque, Marilyn Mantei, and Ray Reiter.

A number of other people read and commented on this thesis in its various stages. I am especially grateful to Mark Steedman for his detailed review and to Chrysanne Di-Marco, Mike Gruninger, Diane Horton, David McDonald, Sheila McIlraith, Ray Perrault, and Evan Steeg for their comments.

I also benefited from many discussions that helped both in developing and clarifying my ideas and also in making the work a lot more fun to do. I am indebted to Jim Aragones, Nicolas Ascher, Susan Brennan, Graham Button, Jean Carletta, David Chapman, Adam Grove, Elizabeth Hinkelman, Jerry Hobbs, Eduard Hovy, Lynn Lambert, Diane Litman, Javier Pinto, Rich Thomason, David Traum, Lyn Walker, Bonnie Webber, and Jan Wiebe.

Financial support for this work was given by the University of Toronto, the Natural Sciences and Engineering Research Council of Canada, and my parents.

Finally, this thesis would not exist if not for the support and patience of friends and family members. To Jim, Evan, Carol, Graeme, Nadia, Diane, Tom, Chrysanne, Suzanne, my parents, and my friends in the NLU and KR groups and DCS at large, I (perhaps again) express my sincerest gratitude.

Contents

List of Figures

List of Examples

List of Definitions

Chapter 1

Introduction

In conversation, speakers must interpret what the other speakers have said. To decide how to respond, they must determine *why* others have said what they did. Most computational accounts of this task have assumed that once listeners have interpreted an utterance, they never change this interpretation. However, human interactions routinely violate this assumption.

This is because people are necessarily limited in how much information they can make explicit. As a result, *misunderstandings* might occur—discourse participants might differ in their beliefs about the meaning of what has been said or about its relevance to the discourse. To address this possibility, participants rely in part on their expectations to determine whether they have understood each other. If a speaker fails to notice anything unusual, she may assume that the conversation is proceeding smoothly. But if she hears something that seems inconsistent with her expectations, she may hypothesize that there has been a misunderstanding and attempt to reinterpret part of the discourse, initiating a *repair*.

In other words, speakers' inferences about discourse are nonmonotonic, because speakers may learn things that conflict with their earlier reasoning and cause them to re-evaluate what happened before. Because their utterances can explicate only part of what a speaker means, discourse participants' can only surmise—*abduce*—each other's intentions. They must reason from utterances to causes or goals that might account for them. As a result, one might later discover (or be told) that some earlier interpretation was wrong, or that wrong inferences were drawn from it.

1.1 The importance of detecting misunderstandings

Failing to monitor for differences in understanding can lead to trouble. Listeners expect that a speaker's utterances will have some coherent relation to the discourse so far. So, when they consider the many possible interpretations of an utterance, they may rule out alternatives that are inconsistent with the discourse. But misunderstanding can lead to a breakdown in the conversation in which one participant cannot interpret some utterance. Misunderstanding can also cause participants to form different judgements about what they have accomplished; this problem is illustrated by Example 1.1 (from Raudaskoski, 1990).

The example involves an interaction between an experimental speech-recognition system and its user. The user is left with the mistaken impression that her message for Eve has been successfully recorded, although it has not. The user was misled because she did not recognize the system's attempts to initiate a repair (T3, T5, and T7). The system, because it could not detect errors in the understanding of either itself or the user, could only try to make the user rephrase her request. This unexpected limitation, coupled with the ambiguity of the system's utterances, then prevented the user from detecting the misunderstanding herself. For example, the system's reply T7 is consistent with the user's interpretation of T5 as a request for the message and her reply, T6, as compliance with it.

Example 1.1 'Leave a message'

T1	**System:**	So you want to leave a message for Eve—is that right?
T2	**User:**	A message for Eve that's correct.
T3	**System:**	What is it that you want to do then?
T4	**User:**	Can you say that Judy is at home?
T5	**System:**	Do what?
T6	**User:**	Judy is at home.
T7	**System:**	You can leave messages or listen to messages—select either.
T8	**User:**	I want to leave a message for Fred.
T9	**System:**	So you want to leave a message for Fred—is that right?
T10	**User:**	Yes, for Fred that's correct.
T11	**System:**	Please give the message.

The problem underlying this example is that the system fails to use expectations. When people speak, they expect to receive a coherent response and, when they respond, they expect to have their action taken as evidence of understanding. Misunderstandings can occur when people form different expectations. To recognize and repair them, discourse participants must be able to detect that their expectations have been violated.

1.2 The defeasibility of interpretation

Participants in a dialogue bring to it different beliefs and goals. These differences can lead them to make different assumptions about one another's actions, construct different interpretations of discourse objects, or produce utterances that are either too specific or too vague for others to interpret as intended. These problems may in turn lead to misunderstandings. Some of these can be circumvented, for example, by a hearer's identifying and correcting others' apparent misconceptions about objects or concepts mentioned in the discourse or by a speaker's adding disambiguating information to her own descriptions before uttering them (Goodman 1985; McCoy 1985; Calistri-Yeh 1991; Eller and Carberry 1992; Zukerman 1991). Not every misunderstanding can be avoided, however, because no speaker ever has enough information, by herself, to determine how her utterance will be understood, even if perfectly cooperative and considerate of the context. Nor can she know whether she herself has understood. Maxims for cooperative behavior are insufficient, because in general neither participant can know ahead of time what will be informative or relevant for the other.

Discourse participants compensate for this limitation by using the evidence provided by their utterances to verify and revise their understanding of the conversation (*cf.* Clark and Schaefer 1989; Brennan 1990). If either participant disagrees with the other's interpretation, they can challenge it. Alternatively, participants may accept an interpretation and respond with an utterance that shows their understanding and acceptance of it. In effect, speakers *negotiate* the meaning of utterances. This is illustrated by Example 1.2 (Gumperz, 1982) where the negotiation is made explicit. In this exchange, Husband intends T1 as a request for the location of the newspaper, whereas Wife takes it as a request to fetch the paper—even after being told that he just wants the information. Alternatively, Husband might have accepted her interpretation by simply saying "Thank you".

Example 1.2 'Where is the paper?'

T1	**Husband:**	Do you know where today's paper is?
T2	**Wife:**	I'll get it for you.
T3	**Husband:**	That's okay. Just tell me where it is. I'll get it.
T4	**Wife:**	No, *I'll* get it.

1.3 The need for both intentional and social information

The model of dialogue to be presented in this thesis accounts for the interpretation and production of utterances and the detection and repair of misunderstandings. To consider possible misunderstandings in addition to intended interpretations would explode the number of alternatives that an interpreter would need to consider, unless there were adequate constraints. However, predominant computational approaches to dialogue, which are based on intention, already have difficulty constraining the interpretation process. Proposed sociological accounts are more constrained, but none are computational. The new model combines the strengths of both intentional and social accounts of discourse.

In intentional accounts, speakers use their beliefs and goals to decide what to say;

when hearers interpret an utterance, they try to identify goals that might account for it. This sort of reasoning is difficult to constrain because, although beliefs can narrow the search for an interpretation, there is no principled way of constraining the depth of the search. For each motivation that a hearer considers, he must also consider any higher-level motivations that it might support. To make such an approach workable, many simplifying assumptions have to be made, including the assumption that previous parts of the conversation have been understood correctly. However, there is another way to address misunderstanding that avoids this unconstrained inference of goals: use expectations deriving from social conventions (rather than intention) to guide interpretation.

In sociological accounts provided by Ethnomethodology, both coherent discourse interactions and repairs of misunderstandings are normal activities guided by social conventions (Garfinkel 1967; Schegloff 1992). There are conventions regarding the expected range of responses to every action, for example. People then can assume that others are behaving as generally expected, unless they have reason to believe otherwise. In this way, the conventions give speakers a guide to possible interpretations. Reasoning is also limited, because conventions do not depend on the psychological characteristics of particular participants. What these accounts lack that computational accounts provide is an explanation of how people can identify the convention that is relevant, especially when there is no pre-existing expectation.

The theory to be presented combines both intentional and social sources of knowledge. Together, these sources enable discourse participants to form expectations, to identify inconsistent actions, and to repair misunderstandings. Intentional information will be captured by two relations: one between utterances (input forms) and speech acts, and one between utterances and the attitudes that they express. These relations will be the basis for deciding whether a set of utterances is consistent. To capture socially-derived expectations, the theory will include a relation on the speech acts—for each act, which acts are expected to follow. It will also contain an axiomatization of speakers' knowledge for generating appropriate utterances and for detecting and repairing misunderstandings. The model will show how these decisions depend on interactions among discourse participants' beliefs, intentions, previously expressed attitudes, and knowledge of social conventions.

1.4 Thesis of the thesis

In this thesis, I present a model of communicative interaction that supports the negotiation of meaning discussed in Section 1.2. According to the model, speakers form expectations on the basis of what they hear, and thus monitor for differences in understanding. If necessary, they also reinterpret utterances in response to new information and generate repairs. Beliefs about the discourse context and conventions for interaction are used to select speech acts that are appropriate for accomplishing the speakers' goals. Interpretation and repair attempt to retrace this selection process abductively—when speakers attempt to interpret an utterance, they try to identify the goal, expectation, or misunderstanding that might have led the other agent to produce it.

The model that I present includes a representation of the information that enables speakers to negotiate understandings. It also provides a characterization of how they form beliefs and expectations on the basis of an utterance, allowing that this reasoning may depend on current beliefs and expectations. This reasoning is clearly nonmonotonic; the model will demonstrate that it can be characterized quite naturally as abduction.

The key features of the model that distinguish it from previous ones are the following:

- An account of the detection and repair of speech act misunderstandings and its relation to generation and interpretation. Although there has been work on identifying *potential* sources of misunderstanding, none of it addresses the problem of identifying and repairing *actual* misunderstandings. Also, unifying these tasks requires that linguistic knowledge and processing knowledge be kept distinct, improving the clarity of the model and permitting general knowledge about language to be reused.

- An integration of the socially-determined, structural conventions that have been identified by Ethnomethodology with the use of belief and intention that has been popularized within Artificial Intelligence. As a result, the model does not do extended inference about goals when it is not necessary.

- An account of the nonmonotonicity of discourse reasoning. In particular, the interpretation of utterances and the detection of misunderstandings are both characterized as abduction problems; speech act generation is characterized as default reasoning. As a result, all three processes can be specified within a single theory of communicative interaction.

- A reification of expectation. According to the model, agents form expectations on the basis of social conventions. They filter these expectations by considering the consistency of the intentions that they have expressed. By contrast, previous models of discourse attempt to eliminate interpretations by using some (necessarily incomplete) set of felicity conditions.

- An axiomatization in Prioritized Theorist (Poole, Goebel and Aleliunas 1987). Theorist is a declarative framework for default and abductive reasoning. Thus, linguistic knowledge and processing knowledge are kept distinct.

The scope of this thesis will be limited to explicating the proposed architecture for discourse reasoning and demonstrating how, within this framework, beliefs and goals of discourse participants, social conventions for language use, the dynamic interactive context can account for the kinds of understanding, misunderstanding, and repair that we find. In particular, it will not address several important issues, such as sentence processing, belief revision, user modeling, and extra-linguistic planning and problem solving; however, I will speculate about what would be required to account for them.

1.5 Overview of the thesis

The next chapter reviews intentional and sociological approaches to dialogue, previous work on (preventing) misunderstanding, and the use of abduction to solve problems that resemble the detection of misunderstanding. Chapter 3 then lays out the new theory of dialogue, showing how it extends the intentional and social accounts to account for both coherent utterances and misunderstandings. Chapter 4 describes Prioritized Theorist, a framework for default and abductive reasoning. Chapter 5 describes a formal language that I will use to describe the theory within this framework. Chapter 6 gives an axiomatization of the theory using this language. Chapter 7 applies the reasoning capabilities of Prioritized Theorist to show how the axiomatization accounts for real examples of generation, interpretation, and repair. Chapter 8 discusses some possible extensions to the model, including speculations on how this work might be done. Chapter 9 summarizes the contributions of the thesis and directions for future work.

Chapter 2

Background

In this chapter, we shall consider how previous work provides a basis to begin solving the problems addressed in this thesis. There are four areas of previous work that we will draw upon. First, proposed characterizations of pragmatic force will help answer the fundamental philosophical questions "What is one really *doing* when one speaks?" and "What is the role of intention in communication?". Second, previous accounts of speech act interpretation will provide insight into how people reason about intention and understanding. Third, there has been work on categorizing and preventing misunderstanding that offers insight into how a speaker might detect and correct it. And last, previous applications of abduction to diagnosing faults and recognizing agents' plans suggest that there may be an analogous account of interpretation and repair. The following sections will discuss each of these areas of research.

2.1 Characterizing pragmatic force

2.1.1 Grice versus Searle

In his definition of *non-natural meaning*, Grice (1957) introduces the notion of reflexive intention as intention "to produce some effect in an audience by means of the recognition of this intention" (p. 385). In other words, communication works because speakers intend that their listeners identify reasons for what they utter and listeners recognize that speakers have this intention. According to Grice, a speaker might intend for the hearer to adopt certain beliefs or intentions or perform certain actions. However, Grice's account runs into trouble at this point, because a hearer may fail to adopt the beliefs that a speaker expresses, even after recognizing the speaker's intention (Searle 1969).

Searle (1969) argues that a better way to define speech acts would be to provide, for each act, a set of necessary and sufficient conditions for its (literal, nondefective) performance. Such conditions would need to hold in order for an utterance of the act to be considered appropriate. And, to the extent that one assumes that an act has been performed appropriately, the conditions would also specify the effects of the act on the discourse context. For example, in order for a speaker S to have sincerely and successfully promised to do something, the following conditions must hold (Levinson 1983, p. 238–239):

1. S said S would perform the future action.
2. S intends to do it.
3. S believes S can do it.
4. S thinks S wouldn't do it anyway, in the normal course of action.
5. S thinks the addressee wants S to do it (rather than not do it).
6. S intends to obligate S to do it by uttering the promise.
7. Both S and addressee comprehend the utterance.
8. They are both conscious, normal human beings.
9. They are both in normal circumstances—not, for example, acting in a play.
10. The utterance contains some illocutionary[1] device that is only properly used if all the appropriate conditions obtain.

One difficulty in trying to formalize speech acts in this way is that there are many reasons that the utterance of a speech act might be defective, most of which no agent would consider unless communication fails.[2] As a result, one might need to consider a plethora of speech act types, corresponding to minor variations of conditions. For example, there might be a special type of promising in which S does not really believe that the action is possible. However, even if one could enumerate every possible variation, the generalizations among speech acts (*e.g.,* conditions 7–10 are common to all types) would likely be lost.

Another problem with Searle's approach is that it assumes that conditions for literal, non-defective performance do not depend on the social context, although, in fact, they do. For example, when a speaker says "I will meet you on Thursday", the speaker and the listener normally both understand that there is an unstated condition, "unless there is a good enough reason not to". What would count as a sufficiently compelling reason for canceling the meeting depends on the social relationship between the two—while a friend might consider "unless I have a date" sufficient, an employer probably would not. Similarly, although a speaker might intend to utter a promise, if the hearer chooses to interpret it as something else (*e.g.,* a threat or an insult), the speaker will be accountable for that interpretation.

An alternative solution to Grice's problem would be to provide a more careful account of the *content* of reflexive intentions. For example, the intended effect of an utterance might not be for the hearer to *believe* the speaker, but for the hearer to take the speaker's utterance as a *reason* to believe that the speaker has a particular belief or

[1]'Illocutionary' refers to actions that an agent directly performs in making an utterance, such as requesting or promising. They are distinguished from physical acts of uttering, which are called 'locutionary acts', and the effects that an act might have on another agent, such as frightening or convincing. The latter are called 'perlocutionary' effects. This three-way distinction was first made by Austin (1962), who also introduced the notion of treating utterances as actions.

[2]This is a case of the qualification problem in artificial intelligence.

intention (Bach and Harnish 1979). Whether the hearer actually adopts the speaker's belief might depend on the context and the hearer's other beliefs. Successful recognition of reflexive intentions, however, would not need to depend on one agent being able to convince another. For example, if Sally tells Tom that α is true, then she expresses that she believes α and that she intends for him to believe α. Moreover, by her utterance she expresses these attitudes *even if* she does not really believe α or does not really expect him to believe it. Thus, a statement counts only as an *attempt* to make the hearer adopt these beliefs and is successful exactly when the hearer recognizes it as an attempt.

2.1.2 Inferential approaches

The so-called "inferential" approaches to speech act theory, such as those of Bach and Harnish (1979), Cohen and Levesque (1985a; 1990b), Perrault (1987; 1990), and Shelley (1992), can be seen as a bridge between the purely intention-based account suggested by Grice and the more convention-based one described by Searle. These approaches reject Searle's notion of felicity conditions as primitive, and instead derive their effects from Gricean intention, some general principles of communicative interaction, and a representation of participants' mental states. According to this view, conversants plan and recognize speech acts by calculating how their actions affect each other's beliefs and intentions.

For example, Cohen and Levesque (1985a, 1990b) attempt to formalize the nature of communicative intentions. Their account treats speech act types as types of committed attempts by a speaker to bring about a state of affairs where it is mutually believed by the dialogue participants that the speaker wants some goal. For example, they define a request as follows:

> A request is an attempt to achieve some condition while being committed to making that goal public. The goal in question has two parts: first, that some action should eventually take place; and second, that the addressed party should intend to do that action relative to the speaker's wanting it done and to the hearer's being helpfully disposed toward the speaker. (Cohen and Levesque, 1990, p. 241)

According to this view, if Sally requests that Tom buy her a soda, they will form a mutual belief that she sincerely has the persistent goal of getting him to do it, and if he is helpful, the model entails that he will eventually buy her one. (A persistent goal is one that an agent will not drop unless she or he comes to believe that it has become true, that it can never become true, or that the reason for its adoption no longer holds.)

Complementary to this effort to specify the effects of communicative intentions is Bach and Harnish's (1979) attempt to characterize the inference strategies that achieve them. They suggest that a listener will attempt to account for an utterance by reasoning backward from the utterance to a speech act type that would provide a plausible explanation for it. The basis for this inference is the communicative intentions expressed by the utterance, the participants' mutual contextual beliefs, and the following set of communicative principles:

The linguistic presumption
The discourse participants share knowledge of a language and the possible meanings of sentences of that language.

The communicative presumption
In a communicative context, a speaker's utterance has some recognizable illocutionary intent.

The presumption of literalness
If, under the circumstances, a speaker could be speaking literally then she or he is indeed speaking literally.

The literal act type of a sentence is determined by the sentence mood (declarative, imperative, or interrogative) and by the objects to which an utterance of the sentence would refer (including times, properties and relations). Mutual contextual beliefs are the beliefs that the speaker and hearer share about each other that are both relevant to and activated by the context. Listeners use these beliefs to find acceptable explanations:

- If, for some utterance, a listener finds that the corresponding direct literal act does not seem relevant according to the participants' mutual contextual beliefs, he will seek an indirect account.

- If the literal act seems inconsistent with these beliefs, then he will seek a nonliteral direct account (*e.g.,* irony), and, if necessary, a nonliteral indirect account.

The use of a standardized act or a marker (such as "please") can short-circuit this reasoning.

According to all inferential approaches, speakers can chain arbitrarily deeply through their knowledge to find an interpretation that is relevant according to their mutual (contextual) beliefs. This introduces two problems: deciding what beliefs are mutual and determining which of them are relevant to the context. Representing mutual belief is a problem because, by most accounts, it corresponds to an infinite conjunction of beliefs:

$$\mathbf{B}_{x,t}\ p \wedge\ \mathbf{B}_{y,t}\ p \wedge\ \mathbf{B}_{x,t}\ \mathbf{B}_{y,t}\ p \wedge\ \mathbf{B}_{y,t}\ \mathbf{B}_{x,t}\ p \wedge\ \mathbf{B}_{y,t}\mathbf{B}_{x,t}\ p \wedge\ \mathbf{B}_{y,t}\mathbf{B}_{x,t}\mathbf{B}_{y,t}\ p \wedge \ldots$$

As a result, one must either treat it as primitive, deriving it from situations of copresence, or make use of some finite subset of alternating beliefs. (See Clark and Marshall (1981)).

A more serious concern with the inferential approach is how to assess relevance in a particular context. However, these accounts offer almost no help in this regard. Most point to Gricean maxims for cooperative communication (see Grice 1975), suggesting, for example, that speakers determine the relevance of an utterance by assuming that it is an act whose type is appropriate to the current stage of the exchange, that it is sincere, and that it provides the appropriate quantity and quality of information. Thus, if a speaker produces a directive, the listener can assume that the speaker provides the requisite information for compliance, believes that compliance is possible, and has reasons for wanting compliance. However, to specify these conditions for each act would reintroduce the problems associated with Searle's felicity conditions. In more recent work, Bach (1990) suggests that some integration of the inferential approach with planning theory (which does adopt Searle's approach) might help delimit the context; however, as we shall see in Section 2.2.1, traditional plan-based accounts face the same difficulty in determining where to stop inference. Something more powerful is needed.

2.2 Interpreting speech acts: Ethnomethodology versus Artificial Intelligence

Artificial Intelligence has traditionally taken the notion of intent very seriously; much of the research in planning and communication revolves around the idea of agents forming intents and identifying the intents of others (*cf.* Cohen, Morgan, and Pollack 1990). On the other hand, Ethnomethodology and Conversation Analysis (see Garfinkel 1967) try to describe observable behavior as objectively as possible, and avoid any ascription of intent. Both of these approaches offer certain advantages.

2.2.1 Plan-based accounts

Starting with Cohen (1978) and Allen (1979; 1983), most AI accounts of discourse have been plan-based; they treat speech acts as STRIPS operators, defined by sets of preconditions, effects, and steps for achieving those effects.[3] For example, Hinkelman (1989)[4] defines questions (`Askif`) and answers (`Informif`) as shown in Figure 2.1.

STRIPS-based accounts implement Searle's (1969) conditions for non-defective speech acts. These conditions are implemented both explicitly, as components of a particular plan, and implicitly, as side effects of the planning strategy itself. For example, Searle's **preparatory conditions** and **essential conditions** are typically modeled explicitly as preconditions and effects.[5] Searle's **sincerity conditions** and **propositional content conditions**, on the other hand, are normally left implicit; for example, such systems typically assume that a speaker would not plan an act if she believes that its effects already hold.

According to plan-based accounts, agents form plans by chaining together actions with matching preconditions and effects. Similarly, agents determine the illocutionary purpose of an utterance by plan recognition. That is, to interpret a speech act, they will chain from subaction to action, from actions to effects of other actions, and from preconditions to actions to identify a plan (*i.e.,* a set of actions) that includes the observed act (see Allen and Perrault 1979; Sidner and Israel 1981; Litman 1986; Carberry 1985).

[3] Here I use "plan-based" and "STRIPS-based" interchangeably to describe all systems that use a representation language similar to STRIPS (Fikes and Nilsson 1971) and NOAH (Sacerdoti 1977). That is, speech acts and discourse plans are specified by a header that names an action (and functions as a frame type) and a specification of the preconditions, effects, and steps in the decomposition for an act of the given type (typically represented as slots). However, the operators of different plan-based systems are not necessarily compatible with each other, because formulae in a specification admit several possible interpretations (Pollack 1986 discusses this problem). These systems also employ a variety of algorithms to index actions and compute their effects.

[4] We shall consider Hinkelman's work in detail in Section 2.4.3.

[5] The essential conditions specify what constitutes the performance of a given type of speech act. For example, a request counts as an attempt to get the hearer to perform the requested act.

```
Askif(S:Agent, H:Agent, P)
     Constraints:  Able(H, Informif(H, S, P)),
                   Knowif(H, P)
                   Want(S, Knowif(S, P))
     Body:  Do(S, Believe(H, Want(S, Informif(H, S, P))))
     Effects:  Informif(H, S, P)

Informif(S:Agent, H:Agent, P)
     Constraints:  Knowif(S, P)
     Body:  Believe(H, Want(S, Knowif(H, P)))
     Effects:  Knowif(H, P)
```

Figure 2.1: Hinkelman's (1989) STRIPS-based speech plans for `askif` and `informif`.

Note that the effects of STRIPS operators are defined in terms of lists of formulae to be added or deleted from a database. Here, where the syntax allows some action *action*`(H, S)` as an effect, this is understood as adding the proposition `Believe(H, Want(S,` *action*`(H, S)))` to the database that represents `H`'s mental state. The type coercion is performed by an inference rule.

Heuristics are applied to discriminate among alternatives (*e.g.,* they might favor longer chains of inference over shorter ones). In addition, a plan-based reasoner might use the state of the ongoing task and of the participants' knowledge and goals to eliminate or deprecate interpretations.

Allen and Perrault

Allen and Perrault (1979; 1980) show how plans can be used to understand and generate appropriate replies to indirect speech acts (such as the use of "Can you pass the salt?" as a polite request to pass the salt). The first link into their planning system is the surface (literal) speech act. From this act, the system attempts to reconstruct the underlying plan and identify any obstacles to that plan. It recognizes the surface speech act using conventionalized features of the utterance such as the sentence mood and verb type. Inference rules are used to identify the plan. To decide among possible interpretations, it evaluates each one according to a set of rating heuristics and performs only tasks that are associated with the highest-rated alternative. Figure 2.2 contains paraphrases of some of these rules.

To see how this approach works, we will consider how it might interpret "Do you know the secret?" as a request to tell the secret.[6] Figure 2.3 shows the inferences that the listener, S2, would need to make in order to figure out that the speaker, S1, might

[6] This example is discussed in Allen (1983).

Plan Construction

- **Action-Precondition** If X wants act A and P is a precondition of A, infer that X wants P.
- **Action-Body** If X wants act A and B is the body of A, infer that X wants B.
- **Know-Rule** If X wants P, infer that X wants to know if P holds.

Plan Interpretation

- **Precondition-Action** If the speaker believes the hearer wants P (which we abbreviate as SBAW(P)), and P is a precondition of ACT, infer SBAW(ACT).
- **Body-Action** If SBAW(B) and B is the body of ACT, infer SBAW(ACT).
- **Know-Positive** If SBAW(A knowif P), infer SBAW(P).

Rating Heuristics

- **Effects true** Decrease the rating of a path if it contains an action whose effects are true at the time the action starts.
- **Intersection found** Increase the rating of a path if it represents an intersection between the paths generated by plan inference and those generated by plan construction.
- **Inference rule applied** Increase the rating of a path each time an inference rule is applied (*i.e.,* give more support to longer inference chains).

Figure 2.2: Examples of Allen and Perrault's (1983) plan-inference and construction rules.

Here "wants P" should be interpreted as "intends to do or achieve P" rather than just "desires" it to be true.

want to know the secret.

However, this account is implausible in two ways. First, it ascribes to S1 (the unlikely) intention to *achieve* that S2 know the secret (step 5). Second, it assumes that hearers reason as long as possible to uncover every intention that a speaker might possibly have. A more realistic account would include as part of the hearer's knowledge that it is conventional to express an `askref` with a question of the form "Do you know X?". Alternatively, it should consider social norms. That is, a hearer should know that the expected reply to the speaker's question would be an `informif(s2, s1, knowref(s2, theSecret))`, expressing the same beliefs necessary for the coherent use of an `askref`. In either case, `intend(s1, knowref(s1, theSecret))` is a linguistic intention expressed by `askref`.

The approach treats the interpretation of speech acts as a bidirectional search over

1. S2 infers from the syntactic form of the utterance that S1 wants to perform a `surface-request`: `believe(s2,want(s1, surface-request(s1, s2, informif(s2, s1, knowref(s2, theSecret)))))`.
2. S2 infers from the definition of `surface-request` that S1 wants the effect of this act: `believe(s2, want(s1, mutualBelief(s2, s1, want(s1, informif(s2, s1, knowref(s2, theSecret))))))`.
3. S2 matches this surface-level effect to the body of a discourse-level request to infer that S1 wants the request: `believes(s2, want(s1, request(s1, s2, informif(s2, s1, knowref(s2, theSecret)))))`.
4. S2 infers that S1 wants the effect of the request: `believes(s2, want(s1, knowif(s1, knowref(s2, theSecret))))`.
5. S2 infers that S1 wants S2 to know the secret, because this is a likely inference from `knowif`: `believes(s2, want(s1, knowref(s2, theSecret)))`.
6. S2 infers that S1 wants S2 to tell the secret, because S2's knowing it is a precondition of S2's telling it: `believe(s2, want(s1, informref(s2, s1, theSecret)))`.
7. S2 infers that S1 wants to know the secret, because this would be the effect of S2's telling it: `believe(s2, want(s1, knowref(s1, theSecret)))`.

Figure 2.3: Allen's (1983) account of "Do you know the secret?"

a network of plan operators. The inference rules define possible search paths. The plan-inference rules chain backward from the utterance, while the plan-construction rules search forward from a mental state. The rating heuristics prune unlikely paths and control the order in which paths are considered. The range of possible inferences is therefore limited to the few proposed relationships between utterances and domain plans. These context-independent rules are insufficient, because the relevance of a set of propositions can depend on the context or the state of the ongoing task. (See McCoy (1986).) However, adding sufficient rules would be difficult (if not impossible), because of the potentially explosive number of paths that the rating system would then need to distinguish. (The rating mechanism would be difficult to extend, because it incorporates experimentally-derived values that may interact in unpredictable ways.)[7]

Litman

Work by Litman (1986) attempts to overcome some of the limitations of Allen and Perrault's inference rules by extending the plan hierarchy to include *meta-plans*. Meta-plans are actions that refer to domain plans or to other meta-plans. For example, a domain

[7]It also makes it difficult to distinguish linguistic or domain knowledge from control knowledge, because there seem to be both types of heuristics.

plan may be introduced, continued, or modified. Whenever a meta-plan is considered, it triggers a recursive search for the plans in its parameters. The identified plans provide background information that enables the system to interpret elided utterances and interruptions.

Although the meta-plans add flexibility by increasing the number of possible paths, they also add to the problem of pruning and ordering the paths, requiring additional heuristics. For example, there are heuristics to choose among alternative meta-plans on the basis of the following:

- the presence of clue words,
- inconsistencies with current implicit expectations (*e.g.,* eliminating plans whose effects are already true or whose constraints cannot be satisfied), and
- a default preference ordering on meta-plans (*e.g.,* preferring continuations to clarifications).

Litman also adds a new general heuristic: stop chaining if an ambiguity cannot be resolved. As with Allen's system, these heuristics combine linguistic knowledge with knowledge about the search mechanism itself.

Another disadvantage of Litman's meta-plans is that they introduce control knowledge into the content of the knowledge base, creating links between states of the search and domain objects. For example, the attitude "`want(H,Plan)`" is not distinguished from the processing state "`next(Act,Plan)`" or from the stack operation "`pop(closure(OldPlan))`". (See Figure 2.4.) By mixing processing states with domain plans and machine operations with beliefs, expectations, and temporal constraints, characterizing discourse inference becomes even more difficult.

Carberry

Carberry's (1985; 1987) work takes a different approach to linking utterances to background knowledge, incorporating expectations into a plan-based account of interpretation with explicit processing rules. For example, if the system poses a question to a user, it will expect the user to identify and answer the question. Processing rules associate discourse expectations with an ordered set of discourse goals that the user might pursue; each utterance is assumed to correspond to exactly one of these goals (*cf.* Mann, Moore, and Levin's (1977) dialogue games). For example, the rule for accepting a question suggests the following list of goals:

1. Seek-Confirm (*i.e.,* confirm the surface content of the question)
2. Seek-Identify (*i.e.,* request help in resolving references)
3. Seek-Clarify-Question (*i.e.,* request additional constraints)
4. Express-Surprise-Question (*i.e.,* refuse to accept the question)

To guide the matching process, associated with each discourse goal there is also a processing rule that specifies tests to perform on the input string and the context

```
Introduce-Plan(S, H, Act, Plan)
  Decomposition:  Request(S, H, Act)
  Effects:        Want(H, Plan)
                  Next(Act,Plan)
  Constraints:    Step(Act, Plan)
                  Agent(Act, H)

Continue-Plan(S, H, Step, NextStep, Plan)
  Prerequisites:  Last(Step,Plan)
                  Want(H,Plan)
  Decomposition:  Request(S, H, NextStep)
  Effect:         Next(Act,Plan)
  Constraints:    Step(Step, Plan)
                  Step(NextStep, Plan)
                  After(Step, NextStep)
                  Agent(NextStep, H)
                  CanDo(H, NextStep, Plan)

Modify-Plan(S, H, Change, Changee, NewAction, OldAction, OldPlan, NewPlan,
                  Stack, OldStack)
  Prerequisites:  Want(H,OldPlan)
  Decomposition:  Request(S, H, NewAction)
  Effects:        Pop(Closure(OldPlan))
                  Next(NewAction,NewPlan)
  Constraints:    Parameter(OldAction,Changee)
                  Step(OldAction, OldPlan)
                  Step(NewAction, NewPlan)
                  Equal(NewAction, SUBST(Change,Changee,OldAction))
                  Equal(TYPE(Change), TYPE(Changee))
                  not Equal(Change,Changee)
                  Replace(Stack,OldStack)
```

Figure 2.4: Meta-planning using STRIPS operators: Litman (1986).

model, which is a tree of all the plans and variable bindings recognized so far. These processing rules can be quite specific, testing syntactic properties of the dialogue, the context model, or the model of the information seeker's beliefs. Figure 2.5 shows a rule for matching the goal *express-surprise–obtain-corroboration*, where CONTEXT-PROPS is a conjunction of parameter constraints introduced by each of the discourse goals that lie along a given path of the context model, including any constraints introduced by the goal under consideration.

Rule-DG-Express-Surprise–Obtain-Corroboration-1:
Check that the following conditions are satisfied:

1. The information seeker's elliptical fragment terminates in a "?".
2. The fragment highlights a proposition P in the context of a conjunction of propositions CONTEXT-PROPS in the information seeker's underlying task-related plan. That is, the fragment unifies with or subsumes a relation or parameter of the plan.
3. It is mutually believed that the information seeker already knows the information provider's beliefs about the truth of the proposition (P $\wedge$ CONTEXT-PROPS).

Figure 2.5: Carberry's (1989) discourse-goal rule for *express-surprise*.

Unlike previous approaches, Carberry's model does not consider all possible interpretations and then filter all but the best one. Instead, it considers each expectation and each goal in a fixed order, expanding the first goal that matches the input. Although the system must still apply filtering heuristics to determine the best plan to expand the goal, presumably the set is much smaller than it would be if all possible goals were considered. Once a goal and plan are selected for expansion, the model chains up the plan hierarchy (equivalent to Allen's *action-body* rule) until either it reaches a distinguished top-level discourse plan or it reaches a branch in the hierarchy, so that it can no longer unambiguously infer a plan. Carberry's model demonstrates that it is possible to reduce the number of explicit filtering heuristics required in a plan-based framework by assuming a fixed preference ordering over combinations of actions. The sacrifice here is a loss of generality; the mechanisms for recognizing goals are specific to her particular implementation. The search strategy is narrow, but still arbitrarily deep. It is also clear that using the model to *generate* utterances would require a number of new heuristics for filtering unnecessary goals.

Conclusions

Thus, a major weakness of the both the inferential and the plan-based approach is that there does not seem to be a principled way of constraining inference. From only a germ of content, a tremendous number of goals could potentially be inferred. Ad hoc heuristics must be applied to stop agents from chaining through their knowledge further than necessary. However, as we shall see, the ethnomethodological approach suggests that there may be a way around this problem. According to this approach, there are conversational rules that specify an expected range of responses to every action, giving speakers a guide to possible interpretations. These rules do not depend on the psychological characteristics of particular participants; that is, they are independent of speakers' various desires and motivations (see Heritage 1984, pp. 1–2).

In addition, the STRIPS-based models of discourse that have been proposed have not given us a well-defined theory of dialogue. Even for conventional cases, the key linguistic knowledge sources and strategies that they capture are difficult to identify, because the information is distributed over operator definitions, search algorithms, and heuristics. Moreover, these mechanisms are only weakly constrained in expressiveness (if at all). The lack of constraint makes the wrong predictions about why linguistic reasoning is hard: these models permit an arbitrary amount of reasoning about agents' beliefs and goals, when, as we shall see, a simpler, expectation-driven strategy would not. Instead, extended reasoning would be better reserved for bridging the occasional gaps between expectation and non-conventional communication.[8] At the same time, previous models do not account for the defeasibility of inference that underlies repair. Similarly, they waste effort trying to verify some ad hoc subset of the potentially limitless number of Searlian felicity conditions, when they could derive them from a more limited set of Gricean intentions and a few, more general, principles of interaction (Cohen and Levesque 1985; Bach and Harnish 1979). Moreover, we shall see that even in situations where communication has broken down, to repair the trouble it suffices to identify *how* it has broken down, without performing extended reasoning as to *why*.[9]

2.2.2 Conversation Analysis

In the ethnomethodological account of human communication, known as Conversation Analysis, people design their behavior knowing that they will be held *accountable* for it. The appropriateness of a speaker's actions will be judged by others according to the degree to which it aligns with some (culturally determined) set of conversational rules, such as "when an offer has been made, one must accept or refuse it". Speakers, aware of some rule or norm that is relevant to their current situation, choose to follow (or not follow) the rule depending on how they view the consequences of their choice. They must consider that their utterances will be taken to display their understanding of the rules of conversation and the situation prior to the utterance, and will be subject to acceptance or correction by others. A key issue for analysts (which is still the subject of much debate, *e.g.,* Button 1990, Hirst 1991, and Chapman 1992) is the nature of the agent's choice: the "rule" that maps a situation onto an appropriate action is represented by an agent only implicitly—in terms of the expectable response(s) to a hypothesised

[8]An interesting question would be how certain types of utterances become conventionalized, while others do not. It would seem that some compilation of belief-based inferences occurs, but there remains the question of why not all of them are accepted (Mark Steedman, personal communication).

[9]One might argue that the STRIPS-based approach is flexible enough that these limitations could each be overcome; if it is, one could conceivably use the definition of discourse reasoning developed here as the specification for a new STRIPS-based account.

action and perhaps the reasons for its (their) failing to occur—and not as any sort of internalized condition–action pair, as we might find in a system of production rules. Interpretation is thus a process of reconstructing the choice: determining why a speaker thought her action was appropriate; and what she is expecting now.

The level of description used by Conversation Analysis treats the *adjacency pair* as the primary unit. Adjacency pairs are sequentially constrained pairs of utterances (such as question–answer) in which an utterance of the first type creates an expectation for one of the second.[10] There is a (small) range of expected responses to a given utterance type, ordered by their relative expectedness.[11] Utterance types can be distinguished according to their expected range of replies. A speaker may produce a reply that is out of this range (Schegloff and Sacks 1973); however, that speaker can expect that the producer of the first part will try to infer an explanation. Thus, if a speaker decides to produce a reply that she knows will be outside the expected range, she might use a discourse marker such as "well" to acknowledge the deviance of the response and to avoid appearing to have misunderstood (Schiffrin 1987).

Within an adjacency pair, the typology of units is very similar to the illocutionary acts proposed by speech-act theorists. But, it is considered an important feature of Conversation Analysis that these utterances are classified without reference (by the analyst) to such unobservable features as the speaker's intention. Whether or not one agrees that this restriction is desirable (or possible), an illocutionary level of description is necessary for an expectation-based account, because social norms depend on behaviors that can be sanctioned, observed, and verified (Shimanoff 1980).

Where Conversation Analysis is weakest is in its explanation of how the recipient of an utterance is able to understand it if there is no pre-existing expectation (*i.e.,* no incomplete adjacency pair). It also does not explain why the producer of an utterance should expect the intended pair to be recognized. For this, speakers need linguistic knowledge that links the features of an utterance to a range of speech acts that form adjacency pairs. Speakers also need to have some idea of the beliefs and intentions that particular actions can express, so that they can make judgments about their appropriateness in the context. Thus, some synthesis of Conversation Analysis and traditional plan-based approaches appears to be necessary.

[10] Here we are talking about *normative* expectations. In other words, under normal circumstances, the first part of a pair makes the second a *conditionally relevant* or appropriate reply.

[11] Such rankings need not follow standard accounts of cooperativeness; for example, when a speaker attributes some property to another agent (such as by a compliment or accusation) the expected reply is a *disagreement* with the attribution. Differences in expectedness, known as *preferences* in the literature, lead to a notion of "relevant absence": if the preferred response is absent, then a less preferable one is assumed. Thus, if a disagreement does not follow some attribution, then speaker has cause to believe that the listener agrees with it (Bilmes 1988).

2.3 Understanding misunderstanding

A *misunderstanding* occurs in conversation when one participant's interpretation of some part of the conversation is different from another participant's.[12] Misunderstandings can result from ambiguities and from defects or deficits in a participant's knowledge of the world. They can also result from acoustic difficulties and inattentiveness. Because different sources of misunderstanding can manifest themselves in a similar way, identifying potential sources of misunderstanding is useful mostly as an aid to preventing them. To address the problem of *detecting* misunderstandings, whatever their initial cause, one also needs to characterize their symptoms.

Misunderstandings normally involve either problems of reference or problems of relevance (Remler 1978; Schegloff 1987). *Reference* problems are those in which the discourse participants have a different understanding of the identity of an object or idea mentioned in the discourse, as in the following exchange (from Remler 1978, p. 392), where agent B is mistaken about which dress A is talking about:

Example 2.1 'I like that dress'

T1 **A:** I really like that dress.

T2 **B:** Thanks. I just bought it today.

T3 **A:** Oh. Yours is nice too, but I was talking about the one L has on.

T4 **B:** Oh. I thought you meant mine.

Relevance problems are those in which the participants have a different understanding of the relationship between actions or objects. In Example 2.2 (from Schegloff 1987, p. 208), two sisters are talking on the telephone, and one sister (A) misunderstands an attempt by the other (B) to initiate a *closing* in T1, taking it instead as a *complaint* and hence requiring her to offer an *excuse* in T4.

Reference and relevance problems may also co-occur. In the following example (from Beach 1983, p. 204), W's misunderstanding of "wok" as "walk" in T1 leads to W's misunderstanding A's *assertion* in T1 as a *suggestion*:[13]

[12]This definition assumes that both participants form some interpretation of the conversation, albeit not the same one. This is different from *nonunderstanding*, which is a failure to find any explanation.

[13]In many North American dialects, the words "wok" and "walk" are virtually homophonous.

Example 2.2 'I will see you'

T1	**B:**	Well honey? I'll probably see you one of these days.
T2	**A:**	Oh God yeah.
T3	**B:**	Uh huh.
T4	**A:**	I just couldn't get down there.
T5	**B:**	Oh I know. I'm not asking you to come down.
T6	**A:**	Jesus. I mean I just didn't have five minutes yesterday.

Example 2.3 'We need a wok'

T1	**A:**	We need to get a wok.
T2	**W:**	A walk? Right now?
T3	**A:**	No, later.
T4	**W:**	After dinner?
T5	**A:**	Maybe next week.
T6	**W:**	What are you talking about?
T7	**A:**	A wok, you know, Chinese cooking?
T8	**W:**	Oh. A *wok*.

To avoid breakdown of the conversation, agents must *actively* monitor each other's utterances for symptoms of misunderstanding and take action when it is appropriate. Determining when to act is the key issue. There is a tradeoff between acting at the first sign of trouble, and possibly disrupting the conversation unnecessarily, and acting only when the trouble becomes an impediment, and possibly creating a need for a larger and more difficult repair (see Carletta 1991). Most previous work on misunderstanding takes the first approach, attempting to resolve ambiguities or apparent conceptual errors.[14]

[14]Conceptual errors correspond to believing of some false proposition that it is true. In everyday language, such states of misbelief is also called "misunderstandings", which may lead to some confusion. Here, we shall use the term "misconception" for this sense, while reserving "misunderstanding" for the 'acting on the basis of false premises' sense introduced earlier.

2.3.1 Avoiding ambiguity

One approach to preventing ambiguity from becoming a barrier to understanding is for agents to monitor their own utterances for ambiguities and add disambiguating information when necessary. For example, Zukerman (1991) postulates that expertise-related impairments, such as an inability to evoke the intended concept or some relational aspect of the concept, can occur any time a description is generated. Her solution is to extend all plans to include goals to "invalidate" [*sic*] any potential impairments. (A stereotypical model of the hearer as either "talented" or "mediocre" provides heuristics for filtering unnecessary extensions and for selecting the appropriate devices and content to prevent the impairment.) This amounts to avoiding misunderstandings at the output stage; one can also work to prevent them during the interpretation stage, by monitoring the utterances of other agents for ambiguities or conceptual errors.

2.3.2 Reconciling misconceptions

Misconceptions are a deficit in an agent's knowledge of the world; they can become a barrier to understanding if they cause an agent to evoke an unintended concept or relation. To prevent misconceptions from triggering a misunderstanding, agents can check for evidence of misconception and try to resolve apparent errors. The symptoms of misconception include references to entities that do not map to previously known objects or operations (Webber and Mays 1983) or requests for clarification (Moore 1989). Errors are corrected by replacing or deleting parts of the problematic utterance so that it makes sense. A number of correction strategies have been suggested:

- Generalize a description by selectively ignoring some constraints (see Goodman 1985; McCoy 1985, 1986, 1988; Carberry 1988; Calistri-Yeh 1991; Eller and Carberry 1992),
- Make a description more specific by adding extra constraints (see Eller and Carberry 1992), and
- Choose a conceptual "sibling", by combining generalization and constraint operations. For example, if there is more than one strategy for achieving a goal, then an entity that corresponds to a step from one strategy might be replaced by one corresponding to a step of one of the other strategies (see Carberry 1985, 1987; Eller and Carberry 1992; Moore 1989).

Although these approaches do quite well at preventing certain classes of misunderstandings, they cannot prevent them all. The impossibility of discourse participants knowing all of each other's prior beliefs, and hence anticipating every pitfall, makes it inevitable that even their best efforts to achieve understanding will sometimes fail. These approaches do not, for example, consider the possibility of the other agent applying similar repair heuristics. Moreover, these approaches may actually trigger misunderstandings because they always find *some* substitution, and yet they lack any mechanisms for detecting when one of their own previous repairs was inappropriate.[15] Thus, a conversational

[15] Evaluation of these approaches is also a problem. Most rules for finding a substitution seem unprincipled and implementation dependent. Since most seem to be the kind of thing an agent would learn

participant will still need to be able to address actual misunderstandings.

2.3.3 Maintaining multiple interpretations

At the opposite extreme from trying to prevent all misunderstandings is treating anything other than the "optimal" understanding as a mistake that warrants revision. This approach is most common in text-based systems. For example, Eiselt's (1989) strategy is to maintain all possible interpretations of the discourse and alternate among them whenever there is a change in their rank ordering. As processing proceeds, the system records the current best paths between all the objects mentioned, while preserving the others in an inactive state; then, if an inactive path is rediscovered (*i.e.,* one of its nodes is reactivated), it is re-evaluated and, if it has sufficient rank, substituted for a previously active path. The danger of such extreme flexibility is that it would bring any real conversation to a standstill, each change in interpretation necessitating an interruption of the current discourse to acknowledge the change (so that the other participant does not mistakenly believe the *current* turn has been misunderstood).

2.3.4 Repairing misunderstandings

So, conversation requires more than just trying to prevent all misunderstandings or vacillating among all possible interpretations. Although agents are not likely to act unless there is a clear need for repair, they must be able to handle situations that require it. When it becomes obvious that a discourse warrants repair, an agent will publically alter his or her understanding of it. Fox (1987) refers to such reconstructions of a discourse as *interactional reconstruction.* She characterizes this as follows:

1. Interpret utterance U, disambiguated by the context, as I.
2. Keep I as long as subsequent utterances are "consistent" with it.
3. If subsequent utterances are inconsistent with I, then reinterpret U to try to get rid of the inconsistency.

From an AI perspective, these reconstructions resemble the operation of a truth-maintenance system upon an abductive assumption that has been found to be incorrect.

Most cues to identifying actual misunderstandings are indirect—an agent does something that is incoherent or inconsistent with what she or he might be expected to do. It is useful to divide speech act misunderstanding into two types: misunderstandings that are made and detected by oneself, *self-misunderstanding*, and misunderstandings that are made by one participant, but detected by the other, *other-misunderstanding.* The first type arises when a hearer finds that he cannot incorporate an utterance into the discourse consistently, unless he interprets one of the speaker's utterances differently. The second type occurs when a hearer recognizes that if one of his own acts had been

over time, it might be that the underlying principles derive from a theory of acquisition. Other cognitive evidence would come from examining the types of errors that result from inappropriate substitutions (*cf.* Brown and Van Lehn's (1980) work on Repair Theory).

interpreted differently, the speaker's utterance would have been the expected response to it. The hearer might then attempt to change the speaker's interpretation, by performing a repair. For example, he might restate his intended goal or explicitly tell the speaker that she has misunderstood. Alternatively, the hearer might choose not to make the misunderstanding public, because certain forms of third-turn repairs can easily be mistaken for a challenge (Schegloff, Jefferson and Sacks 1977).

After a speaker detects a misunderstanding by either participant, she may initiate a repair. Conversation analysts have identified three important types of discourse-level repair, distinguished by the number of turns between the misunderstood turn and the start of the repair (Schegloff 1992). The most common type is second-turn (or next-turn) repair. These repairs occur immediately after the misunderstood turn, before the hearer has produced any other reply to it, as in the following example:

Example 2.4 Second-turn repair

T1 **B:** Do you know where Mr. Williams is?

T2 **A:** What?

The next most common type of repair involves correcting another speaker's interpretation of the discourse. In the simplest case, a speaker makes an utterance displaying her misunderstanding in the turn immediately following the one she misunderstood. If the other speaker then recognizes the misunderstanding and initiates a sequence to resolve the misunderstanding, this is a *third-turn* (or *third-position*) repair, so called because the repair is initiated in the third turn of the top-level sequence, counting from the misunderstood utterance.[16] Consider Example 2.5 from Coulthard and Brazil (1984). In this example, B has responded to T1 with an *acknowledgement*, interpreting T1 as an *inform*.

Example 2.5 Third-turn repair

T1 **A:** So the meeting's on Friday.

T2 **B:** Thanks.

T3 **A:** No, I'm *asking* you.

However, A intended T1 to be an *askif* (presumably with an *informif* as the expected reply). Recognizing B's misunderstanding, A produces a third-turn repair in T3, telling

[16] Schegloff(1992) distinguishes nth-turn from nth-position repairs, where the former correspond to repairs that begin exactly $n-1$ turns after the problematic utterance while the latter allow an arbitrary number of intervening pairs of turns. I shall use the terms interchangeably, allowing intervening exchanges.

B what action A had intended in T1. A could have also told B the intended goal (*e.g.,* "No, I want you to tell me.")

The least-favored type of repair involves producing a new reply to a turn that one has apparently misunderstood. Although there is a preference for repairing one's own misunderstandings (Schegloff, Jefferson and Sacks 1977), these repairs are deprecated because the number of potential targets for the repair increases with each intervening exchange, making locating the target increasingly difficult (Schegloff 1992). If a conversant hears an utterance that seems inconsistent with her expectations (perhaps because she has misunderstood some previous utterance) and the inconsistency leads her to reinterpret an earlier utterance and produce a new response to it, this is a *fourth-turn* (or *fourth-position*) repair (Schegloff 1987). Such repairs not only display the alternative interpretations, but also indicate some of the information that may underlie a participant's decision to favor one of them over another. Consider the fragment of conversation shown in Example 2.6 (Terasaki 1976).

Example 2.6 Fourth-turn repair

T1	**Mother:**	Do you know who's going to that meeting?
T2	**Russ:**	Who?
T3	**Mother:**	I don't know.
T4	**Russ:**	Oh. Probably Mrs. McOwen and probably Mrs. Cadry and some of the teachers.

In this dialogue, Russ initially interprets T1 as expressing Mother's desire to tell, that is, as a *pretelling* or *preannouncement*, but finds this interpretation inconsistent with her next utterance. In T3, instead of telling him who's going (as one would expect after pretelling), Mother claims that she does not know (and therefore could not tell). Russ recovers by generating a fourth-turn repair; he reinterprets T1 as an indirect request and producing T4 as an attempt to satisfy it. This example also demonstrates agents' reluctance to repair the problems in the utterances of others (Schegloff, Jefferson and Sacks 1977); although Mother might have produced an explicit third-turn repair at T3 (*e.g.,* "No, I'm *asking* you."), the manifestation of a misunderstanding provided her with an expectable option that allowed her to avoid doing so.

Once an agent recognizes a misunderstanding, she or he can initiate a repair sequence to resolve it. During the repair sequence, either participant might address what the speaker said, what the speaker meant (*i.e.,* the speaker's goal), what the hearer thought was said, what the hearer thought was meant, or how the hearer responded. A previous utterance may be completed, repeated, emphasized, amplified (elaborated), explained, queried, or refuted. The intended goal of an utterance may be explained, queried, or refuted. Repair devices can also be implicit—an agent might simply change his or her interpretation of a previous utterance and respond to the expectations induced by the reinterpretation (Humphrey-Jones, 1987).

2.3.5 Models incorporating coordination and repair

Clark and Schaefer (1989) propose a model for describing the structure of meaning negotiation in discourse. In the model, each exchange or *contribution* comprises two phases: a *presentation* phase and an *acceptance* phase. Every turn is a presentation. A contribution is complete (or *grounded*) only when it receives an acceptance that the presenter in turn accepts.[17] In the simplest case, each turn is also an acceptance of the turn that precedes it. However, an acceptor may opt not to accept a presentation; likewise a presenter may opt not to accept an acceptance. The acceptance might be explicit (*e.g.,* some *backchannel* device, such as nodding or saying "okay") or it might be implicit (*e.g.,* the next relevant utterance), as in Example 2.7, where T2 is an implicit acceptance of T1 and T3 is an implicit acceptance of T2.

Example 2.7 'Where is Dan?'

T1	**User:**	Where's Dan?
T2	**System:**	In a meeting.
T3	**User:**	Where's Diana?

If a presenter rejects a proposed acceptance, instead of beginning the next contribution at the same level, the rejecting utterance begins a *side-sequence* (Jefferson 1972) for negotiating acceptance. Normally this involves some sort of repair, as in Example 2.8, where the sequence T2–T4 serves as an acceptance of T1. Once such a repair is completed, the original adjacency pair is resumed. The utterance that completes the side-sequence, such as the word "yes" in T4 in Example 2.8, becomes the start of the next contribution at resumed level. Thus, what constitutes the acceptance of a given presentation, and hence the structure of discourse, is determined retrospectively.

Example 2.8 'Is term okay?'

T1	**A:** Is term okay?
T2	**B:** What?
T3	**A:** Is term all *right*?
T4	**B:** *Yes*, it seems all right so far. Touch wood.

Although Clark and Schaefer's model captures the structure of negotiation, it is unable to account for repair, because it says nothing about *why* participants reject pre-

[17] Speech acts may be spread over several contributions, in which case each piece is separately accepted or rejected.

```
if request(S1, S2, informif(S2, S1, knowref(S2, D))) and
   know(S2, knowref(S1,D))
then know(S2, wants(S1, knowref(S2, D))).

if request(S1, S2, informif(S2, S1, knowref(S2, D))) and
   know(S2, not knowref(S1,D))
then know(S2, wants(S1, knowref(S1, D))).
```

Figure 2.6: Cawsey's rules for inferring speaker's goals

sentations. It also fails to specify how either agent knows before the end of the dialogue whether a presentation has been accepted; according to the model, they must postpone their judgement for as long as it is possible for a rejection to occur that undermines prior acceptances. What is needed is a nonmonotonic model that captures what an agent believes at each point in the conversation. Here, I suggest that this model should also be *abductive*.

To see the value of an abductive approach, consider a recent *deductive* one proposed by Cawsey (1991). The core of Cawsey's model is a set of inference rules that are intended to describe how an advice-giving system could infer the goal of its user from the user's utterances and the system's prior beliefs about the user's knowledge. The rules (see figure 2.6) are based on Perrault and Allen's (1980) speech act definitions and rules for plan inference and construction (see Section 2.2.1). The first rule is intended to describe a pretelling. It says that if `S1` asks `S2` if `S2` knows the referent of some description `D` and `S2` knows that `S1` already knows the referent, then `S2` will know that `S1` wants `S2` to know it as well (*i.e.,* that `S1` wants to tell `S2` the referent). The second rule describes the more literal case of requesting. It says that if `S1` asks `S2` if `S2` knows the referent of some proposition, and `S2` knows that `S1` does *not* know the referent, then `S2` will know that `S1` wants to know it (*i.e.,* that `S1` wants `S2` to tell `S1` the referent). Beliefs such as `know(S2, knowref(S1, D))` are assumed to either be true by default or to follow from the dialogue.

Cawsey's prototype implementation of this model treats all antecedents of a rule (*e.g.,* a precondition of an act) as default assumptions. These assumptions can be either *a priori* beliefs of the system or beliefs that it acquires as a result of observing the speech acts the user performs. The system distinguishes the two by giving them different certainty values: whereas observations have perfect certainty, *a priori* beliefs have somewhat lesser certainty.

The basic operation of the system is to apply all its rules to each new utterance to deduce a set of beliefs. Conversational repairs are then handled by standard mechanisms for belief revision: if the system detects a conflict in its beliefs, then the least-certain assumption whose deletion will get rid of the conflict is retracted and the belief set is recomputed.

Cawsey's work, although never fully developed, is important because it demonstrates how a dialogue reasoner can use inconsistencies in the input to recognize possible misunderstandings and generate necessary repairs. (Previous approaches considered inconsistency only as a filter, or possibly as an indicator of irony—see Perrault 1990.) However, the rules clearly oversimplify the task of identifying a speaker's goal. More importantly, they fail to address the source of the problem: it is not just the incompleteness of `S2`'s knowledge about whether `S1` knows the referent that makes `S2`'s beliefs about `S1`'s wants uncertain. Uncertainty also arises because many different interpretations of an action can be achieved in different belief contexts or through negotiation. For example, a yes-no question can be a warning, a test, an insult, a challenge, or even just a vacuous statement intended to keep the conversation going.[18] As a result, utterances are at most a reason for believing that a speaker has a certain goal.

2.4 Solving problems by abduction

The knowledge to which discourse agents have access is necessarily incomplete, nor can they always consider everything that they do know. Conversation unfolds over time, and as it does so, the range of possible interpretations changes—sometimes through addition, but also through elimination. This type of interaction would seem to favor agents' reasoning to a plausible explanation for the phenomena they observe (rather than attempting to find all possible explanations and comparing them), and then using their hypotheses to form expectations, revising them if necessary.[19] The formal approach to dealing with incompleteness is called nonmonotonic inference; reasoning to an explanation is abduction.

2.4.1 An introduction to nonmonotonic reasoning

Formally, nonmonotonic inference is characterized by the following property: the set of conclusions that can be drawn from a set of premises A might not be a subset of the conclusions that can be drawn from a set B that contains A. Nonmonotonicity results when one allows inferences that are not logically valid, relying on defeasible assumptions or an inability to prove some fact. For example, a nonmonotonic inference might make use of a premise that might not always be true, such as that milk is a liquid. From this belief and a belief that one has just tipped a milk carton over a glass, one normally concludes that some milk has flowed into the glass. However, one would not conclude this if one knew (or found out) that the milk was frozen. In contrast, deductive inference is monotonic; the only way to invalidate a conclusion is to remove some of the premises that underlie it. Nonmonotonic reasoning can be tricky, because default generalizations

[18] In Example 2.6, even if Mother knew who was going, she could still be asking Russ a question, albeit insincerely (not realizing that he believes she knows) or she could be trying to test Russ. Moreover, if we consider the conversation from Russ's perspective, he might actually have chosen to treat Mother's utterance as a pretelling, even while suspecting that she did not know who was going, just to confirm his suspicions.

[19] Schmidt, Sridharan and Goodson's (1978) work on a psychological theory of plan recognition supports this assessment both from an empirical point of view and from the viewpoint of AI problem-solving.

might conflict, might not necessarily be transitive, and might get quite complex when exceptions are involved (Reiter and Criscuolo, 1983).

Intuitively, nonmonotonic reasoning can be seen as "jumping to a conclusion" in the face of incomplete information, or having some context-sensitivity to one's inferences, *i.e.,* being able to come to a different conclusion in the face of additional information. The defeasibility and context-sensitivity inherent in nonmonotonic reasoning reveal its sensitivity to the order in which information is considered: if one jumps to a conclusion before any contrary information is known, one might want to revise one's knowledge if such information becomes known, whereas if it were known beforehand, then one might simply avoid the defeated conclusion in the first place, making one's reasoning context-sensitive.

There are two main approaches to default reasoning: consistency-based approaches (McDermott and Doyle 1980; Reiter 1980) and minimal-model approaches (McCarthy 1986). In a consistency-based approach, conclusions can be drawn when there is no evidence to the contrary. Since this condition may change over time as new information is learned, previous inferences may also change. By contrast, in minimal-model approaches, the set of conclusions to be believed is determined by their presence in a model that is "minimal" according to some ordering. Although minimal-model approaches are more interesting formally because they have a semantics, consistency-based approaches have simpler syntax and thus can be implemented more directly, using a theorem prover. Perhaps for this reason, the consistency-based approach has been successfully applied to problems in computational linguistics, while the minimal-model approach has not.[20] (The consistency-based approach is also more closely related to the approaches to abduction that we shall consider in the next subsection.)

Determining consistency is not as straightforward as it may seem, however, since normally the nonmonotonic axioms are themselves part of the theory that must be checked. The formal solution to this problem is to define the theorems of the logic as the intersection of all fixed-points, where a fixed-point is defined as follows:

DEFINITION 1: Let A be a nonmonotonic theory, M be a modal operator such that Mw is true if w is consistent, and Th denote closure under first-order consequence. Then, T is a *fixed-point* of A if $T = Th(A \cup \{Mw \mid \neg w \notin T\})$ (McDermott and Doyle 1980).

As a practical matter, neither these fixed-points nor the theorems are recursively enu-

[20] However, Thomason (1990) suggests using circumscription to specify how speakers coordinate their representation of the common ground of a conversation. His theory makes the default assumption that unless a conversation is abnormal in some publically observable respect (*e.g.,* the respondent to an utterance expresses disagreement: "No", or confusion: "What") both speakers will perform the "mutually supposed supposition update" and thereby maintain coordination. For example, after an assertion, they will both add the same representation of the meaning of what the speaker said during a given turn. The circumscription minimizes the abnormality predicate of the theory with respect to the axioms that update individual propositional attitudes and close the set of possible agents and response types. Although this is an interesting use of circumscription, experimental evidence, such as that of Brennan (1990) and our own work (see Chapter 7), suggests that coordination does *not* occur by default. Speakers' background beliefs may differ, even when they know each other well. As a result, to maintain coordination, they look for and offer evidence of understanding.

merable. However, there are characterizations that can be used to verify whether a set of formulae E is a maximal (with respect to set inclusion), consistent, and deductively closed set of first-order sentences (Reiter 1980).[21]

Applications to speech act theory Perrault (1987; 1990) presents a characterization in Default Logic (Reiter 1980) of some effects of a declarative utterance on the mental states of discourse participants. Default Logic rules have the form $\alpha(\overline{\mathrm{X}}) : \beta(\overline{\mathrm{X}}) \,/\, \gamma(\overline{\mathrm{X}})$where $\alpha(\overline{\mathrm{X}}), \beta(\overline{\mathrm{X}})$, and $\gamma(\overline{\mathrm{X}})$ are all first-order formulae in some language L. A default rule is interpreted (roughly) thus: "if $\alpha(\overline{\mathrm{X}})$ is true and $\beta(\overline{\mathrm{X}})$ is not known to be false, then assume that $\gamma(\overline{\mathrm{X}})$ is true". Perrault's theory uses only defaults for which β and γ are the same; he abbreviates them as $\alpha \Rightarrow \beta$.[22]

Perrault's theory attempts to account for such phenomena as sincere assertions leading to mutual beliefs and for lies not causing the speaker to believe that the lie is true. Central to the theory are two normal defaults: if one believes that the speaker believes something, then one will believe it oneself (Belief Transfer Rule), and if one says something, then one believes it oneself (Declarative Rule):

1. $\mathbf{B}_{x,t}\mathbf{B}_{y,t}P \Rightarrow \mathbf{B}_{x,t}P$
2. $\mathbf{DO}_{x,t}(\mathit{Utter}\ P) \Rightarrow \mathbf{B}_{x,t}P$

The modal operator $\mathbf{B}_{x,t}$ represents agent x's beliefs at time t and has weak S5 properties (*i.e.*, it lacks $\mathbf{B}_{x,t}\alpha \supset \alpha$). The theory also incorporates the following axioms to describe the behavior of beliefs over time and the consequence of observation:

3. $\mathbf{B}_{x,t+1}\mathbf{B}_{x,t}P \supset \mathbf{B}_{x,t+1}P$ (Beliefs persist.)
4. $\mathbf{B}_{x,t}P \supset \mathbf{B}_{x,t+1}\mathbf{B}_{x,t}P$ (Beliefs are remembered.)
5. $\mathbf{DO}_{x,t}\alpha \wedge \mathbf{DO}_{y,t}(\mathit{Observe}\ x) \supset \mathbf{B}_{y,t+1}\mathbf{DO}_{x,t}\alpha$ (An observer of an act will believe that it occurred.)

Lastly, the theory includes a default-rule schema so that if $p \Rightarrow q$ is a default then so is $\mathbf{B}_{x,t}p \Rightarrow \mathbf{B}_{x,t}q$.

Thus, if speaker S1 utters a declarative sentence asserting proposition P to speaker S2, then S2 will reason that P is true by the following chain of inferences:

1. Given $\mathbf{DO}_{S2,t}(\mathit{Observe}\ S1)$,
2. $\mathbf{B}_{S2,t+1}\mathbf{DO}_{S1,t}(\mathit{Utter}\ P)$ can be derived from 1 and axiom 5;
3. $\mathbf{B}_{S2,t+1}\mathbf{B}_{S1,t+1}\mathbf{B}_{S1,t}P$ can be inferred from 2 using Declarative Rule;

[21] A set E satisfying this property is called an *extension*.

[22] Defaults of the form $\alpha : \beta/\beta$are called "*normal*". Theories containing only normal defaults are desirable because then no default can be applied if its consequent contradicts its own justification, some known fact, or the justification of some other default whose prior application is necessary for the rule's application. As a result, such theories always have an extension.

4. $\mathbf{B}_{S2,t+1}\mathbf{B}_{S1,t+1}P$ is derivable from 3 and axiom 3; and

5. $\mathbf{B}_{S2,t+1}P$ is inferable from 4 using BELIEF TRANSFER RULE.

The main advantage of this approach is that the basic structure of the theory can be specified without enumerating all the preconditions and constraints that might possibly defeat a default (as in the STRIPS-based approaches to representing speech acts that were considered in Section 2.2.1). The theory is far too strong, however, because it predicts that agents never revise their beliefs and that a speaker can convince her/himself of an unsupported proposition simply by uttering it. There also does not seem to be any efficient way for a reasoning system to use BELIEF TRANSFER RULE—if the rule is applied in a forward direction, then all of an agent's beliefs about another agent's beliefs will be transferred, including a very large number that have nothing to do with the utterance, but if it is used backwards then it can apply to its own result (Appelt and Konolige 1988).

Appelt and Konolige propose a solution to some of these problems in another consistency-based approach, known as hierarchical autoepistemic logic. Epistemic approaches (Moore 1984; Moore 1985; Levesque 1989) cast default reasoning in terms of knowledge or belief instead of consistency. Nonmonotonicity is achieved by giving a reasoner the power to derive $\neg\mathbf{B}_{x,\neg\beta}$ whenever $\neg\beta$ is not in agent x's belief set. Hierarchical autoepistemic logic extends classical first-order logic by adding an indexed set of unary modal operators, L_i, with the intended meaning "is an element of subtheory i". Hierarchical autoepistemic logic enables one to represent preferences among defaults; instead of treating all premises as part of a single theory, logical structures are partitioned into a partially-ordered set of subtheories where information at lower levels overrides information at higher levels.

In Appelt and Konolige's (1988) reformulation of Perrault's model, the hierarchy is used to specify situation-dependent defaults that may override more general ones. For example, the first level contains only the strongest information about the speaker's and hearer's mental states. The second level includes default rules related to utterances and the speaker's and hearer's beliefs about the current utterance and the third contains defaults about nested beliefs. They resolve the problem of a speaker not convincing herself by uttering something by placing an axiom to that effect in the second level.

2.4.2 An introduction to abduction

While deductive inference solves the problem of determining *whether* a given statement is true, abductive inference provides hypotheses about *why* a statement is true. A typical abduction problem involves a reasoner trying to incorporate a new observation into a theory of the world by determining what assumptions should be added to the theory so that it accounts for the observation. When the theory consists of a set of causal rules of the form `cause` $\supset$ `effect`, then the task is to find an *explanation* for an observed effect in terms of one of the causes. For example, suppose that one observes that the proposition `oil_spot_on_driveway` holds. Also suppose that one's knowledge of oil and driveways includes the following facts about the origin of spots:

`car_has_oil_leak` $\supset$ `oil_drips_under_car` `oil_drips_under_car` $\wedge$ `car_in_driveway` $\supset$ `oil_spot_on_driveway`

`spill_oil_in_driveway` $\supset$ `oil_spot_on_driveway`

Then one may abduce from this theory any of the following explanations:

1. `oil_spot_on_driveway`
2. `spill_oil_in_driveway`
3. `oil_drips_under_car` $\wedge$ `car_in_driveway`
4. `car_has_oil_leak` $\wedge$ `car_in_driveway`

Since it would not make sense to consider an explanation that is already known to be false, a complete definition of abduction requires that an explanation must not only allow one to deduce the effect, but must also be consistent with the rest of the theory:

DEFINITION 2: Given a theory $\mathcal{T}$ and a goal proposition $\mathcal{G}$ one can *abduce* from $\mathcal{G}$ a set of assumptions Δ such that

$\mathcal{T} \cup \Delta \models \mathcal{G}$; and

$\mathcal{T} \cup \Delta$ is consistent.

Abductive inferences are thus defeasible—the addition of beliefs inconsistent with a potential explanation prevent its being abduced. For example, if the agent had already believed $\neg$`spill_oil_in_driveway`, then explanation 2 above would have been rejected.

Slight variations of this definition are quite common in the literature. In systems that rely on a Prolog-style theorem prover, for example, it is more natural to relate abduction to provability ($\vdash$) rather than truth ($\models$) (see Pople 1973; Eshghi 1988; and Poole 1986, 1989). If one treats belief as provability, the explanations that these models abduce correspond to those that the reasoning agent could believe to be true.

Although there are systems that generate explanations deductively (*e.g.,* MYCIN, Shortliffe (1976)), abductive accounts offer certain advantages. First, the axioms for abductive explanation preserve the normal cause–effect relationship, whereas the axioms supporting deductive explanations must invert it. For example, MYCIN includes rules such as these:

```
If:  (1) the stain of the organism is gram-positive, and
     (2) the morphology of the organism is coccus, and
     (3) the growth conformation of the organism is clumps,
then there is suggestive evidence (0.7) that the identity
of the organism is staphylococcus.
```

Such rules allow it to make tentative conclusions about what disease caused the clinical observations. Not only are such rules counterintuitive, but they require *ad hoc* mechanisms for handling uncertainty and the possibility of competing explanations. The only (deductive) alternative is to write axioms of the form $\texttt{symptoms} \supset \texttt{cause}_1 \vee \texttt{cause}_2 \vee \ldots \texttt{cause}_j$ that specify all the known causes for a given effect. In contrast, an abductive model need not assume that its knowledge of causes and effects is complete.[23] In addition, axioms for abductive explanation can be used both for explaining an observed symptom and for predicting what symptoms might result from a given cause; a comparable deductive model would need to include separate rules to enable it to deduce the effects. Finally, because abductive inference is defeasible, it can provide an adequate account for defaults without extending the syntax of the underlying logic.

An important practical consideration is what to do when there is more than one possible explanation. If a reasoner is only interested in a single best explanation, then an abduction system must apply additional constraints, either during the search or as a separate selection process. In the oil-spot example, a reasoner might reject explanation 1 because it is trivial and explanation 3 because it is entailed by explanation 4. A reasoner might prefer explanation 2 to explanation 4 because it requires fewer assumptions, or might prefer explanation 4 if its antecedents were considered to be the most likely. Not surprisingly, what constitutes a best explanation in one domain may be entirely inappropriate in another.

In the next two subsubsections, we shall consider how abduction has been applied in the areas of diagnosis and planning.

2.4.3 Diagnosing faults by abduction

Diagnosis involves identifying a set of faults that would account for discrepancies between the behavior exhibited by a system and the behavior predicted by a model of how the system is supposed to work. A number of different diagnostic systems (*e.g.,* Ahuja and Reggia (1986)); Poole (1986; 1989)) make use of abductive inference. We shall now consider the predominant abductive approaches to diagnosis—set-covering and logic-(proof-) based diagnosis—and their application to problems in computational linguistics.

Set-covering

In a set-covering model of diagnosis, the goal is to find the smallest possible set of faults such that the set of all possible manifestations of those faults contains all the initial observations. The problem specification includes a finite set D of possible disorders, a finite set M of possible manifestations of those disorders, a relation C between disorders and their manifestations, and a set $M^+ \subset M$ of observed manifestations (*i.e.,* symptoms).

Ahuja and Reggia (1986) describe an application of set-covering to the diagnosis of chemical spills. Their system attempts to build a minimal-cardinality set of chemicals whose presence would be sufficient to account for the symptoms observed at the time of a spill. In the model, disorders correspond to chemicals that might be in a spill. Their manifestations are salient properties that may be true in situations where a chemical has been spilled (*e.g.,* color, pH, or time of year). The relation between these two sets links each chemical to the manifestations that it can cause (*e.g.,* petroleum can cause a black

[23] Horton and Hirst (1992) discusses a parallel problem in assuming completeness in axioms that represent the decomposition of actions.

spill); formally, $(d_i, m_j) \in C$ if chemical $d_i \in D$ can cause manifestation $m_j \in M$. The system's observations are a subset of the possible manifestations. Thus, the abduction picks out a minimal set $\{d_{i_1}, \ldots, d_{i_n}\}$ such that if m_k is an observation, then $(d_{i_k}, m_k) \in C$ for $1 \leq k \leq n$.

In the implementation, the disorder relation is represented as a set of frames for each chemical, where each slot describes a manifestation. The diagnostic algorithm indexes a subset of these frames on the basis of an initial set of observations and then uses their slots to generate queries that will eliminate some of the candidates. If a symptom occurs that cannot be accounted for by *any* of the currently active candidates, the system initiates a separate diagnostic process and indexes additional frames to *cover* the symptom. A diagnosis contains a best candidate from each of these processes.

An application to story understanding A common view of story understanding is as the construction of a single coherent representation of the events of the text as the goal of some character in the story. Charniak and Goldman (1988) use an abductive approach very similar to a set-covering diagnosis to solve the problem of recognizing characters' goals. In their system, the diagnostic task is to reason from the text to a minimal cardinality set of objects and events that it describes. The objects and events are organized into a hierarchy of frames linked by the relations ISA, SLOTS, and ACTS:

- ISA is a specialization relation between two events,
- SLOTS is a thematic relation between an event and a set of objects, and
- ACTS is a decomposition relation between an event and a set of actions that accomplish the event.

The manifestations of these entities are the words and syntactic relations of the text. Two distinct types of causal relationships are defined between these manifestations and the domain: a referring relationship between words in the text and objects or events in the domain, and a specialization relationship between the syntactic relations in the text and the slots of objects and events. For example, a noun refers to an object, a verb refers to an event, and an object relation corresponds to a thematic role (*e.g.,* PATIENT) of some event.

Understanding the text involves linking it to the domain on the basis of these two relationships and then consolidating the information provided by the links to form a single unifying explanation. From an initial link, such as a thematic role, the plan recognizer reasons upward through this network to suggest an event that contains the role as a slot or an event that has a step (one of the ACTS) that has the role as a slot. This process terminates when no new links can be found. Then, the smallest sets of event frames that account for the entire text represent possible interpretations of the story.

An application to speech act understanding Speech act recognition can be seen as a diagnostic task in which different features in the input, such as the mood of a sentence or the presence of a particular lexical item, can be treated as manifestations of different speech acts; for example, "please" can be a manifestation of a request. Hinkelman's (1989) work takes this approach and the result is very much like the set-covering systems we have considered.

In Hinkelman's model, the relation between patterns of features and speech acts is given by a matching procedure that the system applies to the input to determine a set of candidates. The system filters candidate acts on the basis of the consistency of the implicatures of each candidate with its model of prior beliefs. Following Allen (1983), implicatures are given as a set of plan inference rules that specify that belief in the doing of an act entails believing that its preconditions and constraints hold and that its effects are wanted.[24] For example, the following inference rule requires that it would be inconsistent for the knowledge base to include both that a hearer `H` believes that a speaker `S` is performing act `A` and `X` is an effect of `A` and that `H` does not believe `S` wants `X`:

$\texttt{B(H,Do(S,A))} \wedge \texttt{effect(A,X)} \rightarrow \texttt{B(H,W(S,X))}$

Thus, if $\texttt{B(H,} \neg \texttt{W(S,X))}$, `A` will not be considered a possible interpretation. (After the system selects a candidate act, it will update the knowledge base by adding the implicatures to it.)

Proof-based diagnosis

The model of conversation described in this thesis has been developed using Prioritized Theorist (Poole, Goebel and Aleliunas 1987; van Arragon 1990), which incorporates what is known as a "proof-based" approach to abduction. In such approaches to abduction, a backward-chaining theorem prover (such as Prolog) is extended to assume certain goals that might not otherwise be derivable, in order to complete a proof. In proof-based diagnosis, these assumptions include hypotheses about which (if any) of the possible failures defined in the system model would cause a given set of observations.

Determining the best explanation in this framework is less straightforward than in set-covering. Because the cardinality of any explanation is extremely sensitive to the particular syntactic details of the theory (which is undesirable), it is common to use a semantic criterion instead. One basis for comparison is the presumption relationship: A_j *presumes* A_i if $\mathcal{T} \cup A_j \models A_i$. For most diagnostic problems, a more presumptive (*i.e.,* more specific) explanation is preferable to a less presumptive one.

Poole's (1986; 1989) work on diagnosis is a typical proof-based approach. The background theory includes axioms that define the normal behavior for each of the components as well as their possible faults. For example, the representation of a circuit might contain the following axioms about the operation of a multiplier:

$\texttt{multiplierOK(M1,In1,In2,Output)} \wedge \texttt{In1} = x \wedge \texttt{In2} = y \supset \texttt{Output} = xy$

$\texttt{multiplierBad(M1,In1,In2,Output)} \wedge \texttt{In1} = x \wedge \texttt{In2} = y \supset \texttt{Output} \neq xy$

Given a set of observations of the form $\texttt{Output} = z$, the diagnostic engine forms hypotheses about the behavior of the system that would enable it to deduce the value of the observed output.) The diagnostic engine then checks the consistency of each potential diagnosis by attempting to prove that its negation is not entailed by the background

[24] These rules do not allow for insincere or indirect acts.

theory. It also eliminates any diagnosis that is entailed by a competitor. If there is still more than one candidate diagnosis, the engine may request additional tests to distinguish them.

An application to sentence interpretation Sentence interpretation requires disambiguating syntactic and semantic structures and resolving local coherence problems such as coreference and the vague relationships found in nominal compounds. This process can be seen as an abduction problem in which the interpreter attempts to provide an explanation of why the semantic representation of the sentence is true.

Hobbs, Stickel, Martin, and Edwards (1988) describe an application of abduction to solving local pragmatics problems. The system uses a weighted abduction mechanism, a variant on the proof-based approach, to find the best set of assumptions sufficient to allow the system to prove the logical form. The system accounts for the logical form of a sentence by abducing specific relations for each of the vague or unspecified relations it contains. For example, the representation of the following sentence,

`The Boston office called.`

includes two unspecified relations, `rel` (the verb-object relation) and `nn` (the noun-noun compound relation), as shown below:

$(\exists \mathtt{e}, \mathtt{x}, \mathtt{o}, \mathtt{b})\ \mathtt{call'(e,x)} \wedge \mathtt{person(x)} \wedge \mathtt{rel(x,o)} \wedge \mathtt{office(o)} \wedge \mathtt{nn(b,o)} \wedge \mathtt{Boston(b)}$

The background theory includes of an abstraction hierarchy of relations. For example, one possible specialization of a noun-noun compound `nn` is the relation `in`:

$(\forall \mathtt{y}, \mathtt{z})\ \mathtt{in(y,z)} \supset \mathtt{nn(z,y)}$

One possible specialization of `rel` is the relation `work-for`:

$(\forall \mathtt{x}, \mathtt{y})\ \mathtt{work\text{-}for(x,y)} \supset \mathtt{rel(x,y)}$

So, one possible interpretation of the sentence is that there was a call from a person who works in an office in Boston:

$(\exists \mathtt{e}, \mathtt{x}, \mathtt{o}, \mathtt{b})\ \mathtt{call'(e,x)} \wedge \mathtt{person(x)} \wedge \mathtt{work\text{-}for(x,o)} \wedge$
$\qquad \mathtt{office(o)} \wedge \mathtt{in(b,o)} \wedge \mathtt{Boston(b)}$

The weighted-abduction algorithm extends the traditional proof-based approach by associating a cost factors (weights) with each part of a clause. These costs enable the system to assign a total cost to each candidate explanation and then choose the best explanation. The best ones are those that require the least costly combination of assumptions. The costs also determine which predicates are assumable; for example, very high values can prevent assumption. However, unlike other approaches that designate the abducibility of certain predicates *a priori*, dynamically computed costs make abducibility sensitive to what else has been proven and what else would need to be assumed to complete a proof. Weighted abduction can thus capture such notions as the explanatory "power" of an assumption and the mutual support among the set of assumptions in an explanation. However, the conflation of these notions as a single weight makes it difficult to select and interpret their values.

There are two types of costs associated with an axiom. The cost of a conclusion represents the system's preference for the trivial explanation, the goal itself. The cost of an explanation is the sum of the cost of each antecedent assumption that is considered while backchaining from a goal, where:

- a previously proven assumption costs nothing;

- an unproven assumption of weight w_a costs $w_a K$, where K is a system-defined constant;
- a newly proven assumption of weight w_a costs $w_a Q$, where Q is the cost of its proof.

2.4.4 Solving plan-related tasks by abduction

Another area where abduction has been useful is the development of theories of planning. The background theory for a planning problem specifies the relationship between a goal and a set of actions or states that achieve that goal. More precisely, an action designator α is a *plan* if one can prove from the theory that α achieves the goal when executed in the initial state (Genesereth and Nilsson 1987). The theory normally includes a *plan library*, that is, a collection of precomputed plans for achieving goals. Each such plan includes a set of preconditions, a sequence of actions, and a set of postconditions such that whenever the actions are performed in a situation where the preconditions are true, the postconditions will become true.

The two main plan-based tasks are planning, the formation of a plan to achieve a given goal, and plan recognition, the identification of a plan or goal that would account for an agent's observed actions. These tasks are similar to diagnosis in that plans play a role analogous to faults in accounting for an agent's goals and actions (*cf.* Josephson 1987; Kautz 1987). But they differ from the diagnostic problems discussed earlier in that often one cannot observe an agent's (intended) actions directly, but only their effects. This indirect inference is exactly the sort that discourse participants will need to perform in order to decide from their utterances what each of the agents has understood. In the remainder of this subsection we shall examine some applications of abduction to solve plan-based tasks.

Planning

The task of a planner is to determine a sequence of actions and conditions whose satisfaction is sufficient to accomplish some goal $\mathcal{G}$. This goal can be any set of propositions, perhaps restricted to a particular time, t_g. For a plan to accomplish the goal, the postconditions of the actions of the plan must entail the goal and must persist until t_g.

Eshghi (1988) describes an account of plan formation that uses a proof-based approach to abduction.[25] Typical of such models, it reasons backward from the goal and the background theory, assuming $\neg\mathcal{G}$ and then applying a resolution-style inference procedure to derive a negated conjunction of actions, persistence assumptions, and temporal constraints.[26] A plan can then be obtained by negating this conjunction and removing any inconsistent persistence assumptions.

In this model, a goal $\mathcal{G}$ is a predicate of the form `holds(f,FINAL-STATE)`, where `f` is an atomic sentence about the world, and `FINAL-STATE` is a distinguished event that

[25] Allen (1991) presents a similar approach.

[26] Eshghi's system also uses integrity constraints to ensure that for every action in the plan, the background theory and the other assumptions are sufficient to prove the preconditions of the action. Such constraints might introduce additional subgoals. Allen's system incorporates preconditions in the background theory itself.

follows all others. The background theory $\mathcal{T}$ contains a representation of the properties true in the initial state, a set of Horn clauses that specify the preconditions and post-conditions of actions, and a general description of the properties of events and actions. For example, the following pair of rules links goals to the actions that achieve them:

1. For events `e1, e2, e3,` and fact `f,`
   `initiates(e1,f)` $\wedge$ `e1` $<$ `e2` $\wedge$ `persists(f,e1,e3)` $\wedge$ `e2` $\leq$ `e3` $\supset$ `holds(f,e2)`

2. For action `a,` event `e,` and fact `f,` where `action(a,e)` associates the event with an action and `postcon(a,f)` associates a postcondition with an action,
   `action(a,e)` $\wedge$ `postcon(a,f)` $\supset$ `initiates(e,f)`

(The first rule says that a fact `f` holds during an event `e2` if there is an event `e1` occurring before `e2` that makes `f` true, and the fact `f` remains true until some event `e3` that occurs after `e2`. The second says that the event `e` makes a fact `f` true when there is an action `a` that happens during `e` and a postcondition of that action is to make `f` true.)

Plan recognition

Plan recognition involves attributing to an agent a plan that contains the actions that the agent has been observed performing. This task is similar to diagnosing the agent's plan, except that the recognizer might need to first identify the agent's goals in performing the actions and then infer a plan from the goals.

Helft and Konolige (1990) propose an abductive model of plan recognition. In their model, the goal $\mathcal{G}$ of a planning agent is an event that an agent wishes to bring about, such as `OCCURS(TravelToSpain,t)`[27]. A plan is a sequence of actions whose combined effect is to bring about $\mathcal{G}$. The agent's theory $\mathcal{T}_a$ includes an abstraction relation among events (*e.g.,* `TravelToSydney` $\supset$ `TravelToAustralia`) and a decomposition relation between actions and events (*e.g.,* `BookTicket` $\wedge$ `TakePlane` $\supset$ `Travel`).[28] The model also includes a theory $\mathcal{T}_w$ of the actual consequences of actions, which may differ from $\mathcal{T}_a$. The set of observations, $\mathcal{O}$, consists of end events, which are the consequences of some actions A as given by the decomposition axioms in $\mathcal{T}_w$; that is, $\mathcal{T}_w \cup A \vdash \mathcal{O}$. $\mathcal{G}$ will be a subset of the observations.

The model incorporates a proof-based approach to abduction. To explain $\mathcal{O}$, the system locates a list of actions, A_0, such that $\mathcal{T}_w \cup A_0 \vdash \mathcal{O}$. For example, if its planning theory were as given in Figure 2.7 and it observed the event `rename_file`, then the system would identify {`move`} and {`copy, delete_original`} as possible candidates. The proof involves assuming $\mathcal{T}_w \cup \neg\mathcal{O}$ and deducing some $\neg(\alpha_1 \wedge \alpha_2 \ldots \wedge \alpha_m)$. Then $\alpha_1 \wedge \alpha_2 \ldots \wedge \alpha_m$ (*e.g.,* `copy` $\wedge$ `delete_original`) is a possible candidate for A_0.

Since it is assumed that an agent executes actions only to serve $\mathcal{G}$, once A_0 has been identified, $\mathcal{G}$ can be deduced, according to the decomposition axioms of $\mathcal{T}_a$. To identify a

[27] Since their model does not include temporal constraints, Helft and Konolige abbreviate such expressions as `TravelToSpain`.

[28] Deductive theories such as that of Kautz (1986) would represent this decomposition as `Travel` $\supset$ `BookTicket` $\wedge$ `TakePlane`. Although we will not say more about the issue here, the choice of which form to use affects not only whether the axioms may be used abductively or deductively, it also constrains the kinds of incompleteness one can represent (Horton and Hirst 1993).

End Events

`rename_file, hide_file, backup_file`

Actions

`move, delete, change_permission, copy`

Plan Library

Decomposition

`move` $\supset$ `rename_file`
`rename_by_copy` $\wedge$ `change_permission` $\supset$ `hide_file`
`copy` $\wedge$ `delete_original` $\supset$ `rename_by_copy`
`copy` $\supset$ `backup_file`

Abstraction

`rename_by_copy` $\supset$ `rename_file`

Figure 2.7: An example planning theory

goal $\mathcal{G}$ and a plan A such that $\mathcal{T}_a \cup A \vdash \mathcal{G}$, the algorithm assumes $\mathcal{T}_a \cup \alpha_1 \wedge \alpha_2 \ldots \wedge \alpha_m$ and deduces a set of formulae[29] of the form $\beta_1 \wedge \beta_2 \ldots \wedge \beta_n \supset \gamma$. For each of these formulae, γ is candidate goal and $\beta_1 \wedge \beta_2 \ldots \wedge \beta_n \cup A_0$ is a candidate plan. For the example, if the system assumes `copy` $\wedge$ `delete_original`, it will deduce the set {`rename_file`, `change_permission` $\supset$ `hide_file`} and thus find `rename_file` and `hide_file` to be candidate goals. (A final step verifies that $\mathcal{T}_a \wedge \beta_1 \wedge \beta_2 \ldots \wedge \beta_n$ is consistent.)[30]

Evaluating an agent's plans

Plan evaluation is a special case of plan inference in which the actor's intended action and goal in performing that action are known to the observer.[31] The observer's task is to provide a *justification* for the actor's intended action in light of the given goal and to determine if the action is part of a valid or invalid plan to achieve it. The justification will include beliefs about the potential effects of actions that relate the intention and the goal and about properties of the situation that would enable the actions achieve the goal (or would prevent them from doing so). As in a diagnostic problem, the observer will assume that the action and the justification entail the goal (directly).

Appelt and Pollack (1992) describe an application of weighted abduction, a variant of proof-based abduction, to the problem of plan evaluation. In their model, a goal is an intention of an agent a to achieve some property P by performing action

[29]The algorithm weeds out old information (*i.e.*, the conclusions of $\mathcal{T}_a$ alone), and irrelevant information (*i.e.*, conclusions that use symbols outside some specified set).

[30]Observed events derivable from a plan and $\mathcal{T}_w$, but are not in $\mathcal{G}$, are *side effects*. In the example, the event `backup_file` is a side effect.

[31]This problem is relevant for task-oriented dialogues, because in these dialogues it is common for the speaker to tell the hearer what his/her goals are (Pollack 1986).

α (*i.e.,* $\texttt{Int}(a,\texttt{To}(\alpha,P))$). The background theory contains a model of the observer's prior beliefs about the actor's mental state and any general beliefs about the domain. Among these beliefs is the following axiom, which allows the observer to conclude that an actor's goal is to achieve P by performing α whenever the observer believes that the actor believes α accomplishes P, intends to do α, and she or he indeed accomplishes P:

$$(\forall a,\alpha)\ \texttt{Bel}(a,\texttt{To}(\alpha,P))^{0.9}\wedge\ \texttt{Int}(a,\alpha)^{0.5}\wedge\ \texttt{Ach}(a,P)^{0.5}\supset\ \texttt{Int}(a,\texttt{To}(\alpha,P))$$

After an utterance, if the observer recognizes the action α the actor intended and the stated goal of that action, the abduction will attribute to the actor similar beliefs as well as a plan to achieve P by doing α. That is, the observer will hypothesize that the actor's plan library includes $\texttt{To}(\alpha,P)$.

Chapter 3

The structured intentional approach

We now introduce a new model of dialogue that extends both intentional and social accounts of discourse. The model unifies theories of speech act production, interpretation, and the repair of misunderstandings. This unification is achieved by treating production as default reasoning, while using abduction to model interpretation and repair. In addition, the model avoids open-ended inference about goals by using linguistic intentions and social norms to derive expectations. Speakers' beliefs also play a filtering role. As a result, the model provides a constrained, yet principled, account of interpretation; it also links social accounts of expectation with other mental states.

In this chapter, we will discuss how the model addresses the following concerns:

- The need to control the inference from observed actions to expected replies. Extended inference about goals is not what people do most of the time. It is also a waste of resources.

- The need to account for nonmonotonicity in both the interpretation and production of utterances. This nonmonotonicity takes two forms. First, utterances are necessarily incomplete, so listeners must reason abductively to account for them. Second, expectations are defeasible. At any given moment, speakers may differ in their beliefs about the dialogue and hence can only *assume* that they understand each other. Speakers manage the nonmonotonicity by *negotiating* with each other to achieve understanding.

- The need to detect and correct misunderstandings. Misunderstandings occur when speakers differ in their beliefs about what has been said or why. Speakers rely on their expectations to decide whether they have understood each other. When listeners hear something that seems inconsistent, they can reinterpret an earlier utterance and respond to it anew. However, if they fail to identify a misunderstanding, the communication might mislead them into prematurely believing that their goals have been achieved.

- The need for an alternative to the notion of mutual belief. Typically, models rely on such beliefs without accounting for how speakers achieve them or for why speakers should believe that they have achieved them.

3.1 Using social conventions to guide interpretation and repair

In the intentional accounts discussed in Section 2.2.1, speakers use their beliefs, goals, and expectations to decide what to say; when they interpret an utterance, they identify goals that might account for it. For example, a speaker who wants someone to know that she lacks a pencil might say "I don't have a pencil." Similarly, a listener might interpret an utterance of this sentence as an attempt to convey the information. However, for any goal that would explain an utterance, the reasons for having that goal would also be potential interpretations of the utterance. For the above utterance, for example, intentional accounts might consider interpretations corresponding to an attempt to express a need for a pencil, a request to be given the pencil, an incomplete attempt to complete a test, and an incomplete attempt to get to California by car. Each of these explanations would create its own range of expected replies. If misconceptions are taken into account, the range of possibilities explodes even further. Yet, the intentional approach offers nothing but heuristics to limit the search for "deeper" reasons or expected replies.

By contrast, in the accounts provided by Conversation Analysis, discourse interaction and the resolution of misunderstanding are normal activities guided by social conventions. These conventions include general guidelines such as "one should answer a question", "one should offer support to a friend that is being criticized", and "if one disagrees with the understanding displayed by others, one should correct them". However, the result is very orderly, because at each turn a speaker is accountable for displaying both her own understanding and her acceptance of others' understanding. The meaning of an utterance is thus as much a product of the listeners' own intentions and accountability to conventions as of speakers' accountability and intentions. Although the notion of social accountability alleviates some of the problems inherent in the intentional approach, the paradigm fails to account for how various linguistic forms accomplish a speech act or for how listeners know whether an interpretation is compatible with what might be expected.[1] The intentional accounts help with both these points.

The model therefore represents a synthesis of the intentional and social approaches. Following intentional accounts, the model will provide a mapping from the features of an utterance (*e.g.,* words, prosody, syntax, and semantics) to a limited class of speech acts.[2] The model will also include a relation between speech acts and the *linguistic intentions* that they conventionally express (Grice 1957).[3] For example, a *suggestion*

[1] Conversation analysts attempt to describe discourse without such a theory, in the hope that they will be able to identify the features present in the dialogue accurately, without distorting or embellishing them.

[2] In general, this mapping would also depend on the context in a functional way.

[3] Although linguistic intentions are conventional, they are not the same as social norms. For example, while an invitation to visit around dinner time may create an *expectation* that dinner will be served, the invitation itself does not express this intention; a social norm might.

that P expresses "the belief there is a reason, but not a sufficient reason, to believe P, and the intention that the hearer believe there is a reason, but not a sufficient reason, to believe P" (Bach and Harnish 1980, p. 44). However, unlike previous intentional accounts, the role of such information will be limited to that of constraint; speakers expect each other to be consistent in the attitudes that they express. Speakers' strategic use of social norms provides discourse coherence. When speakers say exceptional things, they become responsible for providing a reasonable explanation.

The model follows existing speech taxonomies, especially those given by Bach and Harnish (1980), Allen (1983), and Hinkelman (1990). These taxonomies define the intentions expressed by many different speech acts. Many of these acts are specializations of other acts. I have selected to model only a few of the most general types, and found them sufficient to demonstrate the examples in this thesis.[4] Figure 3.1 describes the linguistic intentions associated with nine different kinds of asking, telling, and requesting.[5] For example, it would be inconsistent for a speaker simultaneously to intend to perform both an *askref(S,H,P)* and an *informref(S,H,P)* or both a *pretell(S,H,P)* and an *inform(S,H,not knowref(S,P))*.

Currently the social norms accounted for in the model are limited to the linguistic norms defined by adjacency pairs. Figure 3.2 lists these pairs.[6] Each of the acts selected has a single most-expected reply.

3.2 Characterizing interpretation, production, and repair

If the model is to combine social conventions with mental attitudes, it must say how they interact to create the kinds of understanding, misunderstanding, and repair that occur. While utterance production is a default process of choosing speech acts that meet communicative and interactional goals, it is also necessary that utterances be interpretable. To account for what was said, listeners will try to relate it to goals and expectations, and failing that, attribute it to misunderstanding. Thus, if speakers want others to find what

[4]It is not necessary that all speakers have the same taxonomy of speech acts. In support of this assumption, Kreckel (1981) found that although people within the same family had a similar definition of the speech act *warn*, there were significant differences between non-related individuals; for example, some saw it as a request to take evasive action, whereas others saw it playing a more advisory role. Such variation is possible because speakers seek evidence of understanding within their own model of the discourse. Under normal circumstances, they will not need to reason about another's interpretation. Coordination becomes an issue only if expectation fails.

Similarly, although many social conventions apply to any speaker and hearer, this need not always be the case. People within a family or other social group may develop rules that apply only to members of the group. For example, within a social group it may be known that when a particular member, J, expresses to someone the possibility that she or he perform some action, and that person is able to do it, J expects that person to do it. Similarly, certain situations, like "talking to a bank teller" activate a special set of social conventions.

[5]In the figure, I have used the symbol *intend* to name both the intention to achieve a situation in which a property holds and the intention to do action. This is a simplification. Defining a proper characterization of intention is an open problem that I will not address here (see Cohen et al 1990; Goldman and Lang 1993).

[6]Quantitative results by Jose (1988) provide evidence for these adjacency pairs. In addition, I have used pairs discovered by Conversation Analysis from real dialogues (Schegloff 1988).

Act type	Speech act name	Attitudes expressed
informative	*inform(S, H, P)*	*P* *intend(S, knowif(H, P))*
	informref(S, H, P)	*knowref(S, P)* *intend(S, knowref(H, P))*
	informif(S, H, P)	*knowif(S, P)* *intend(S, knowif(H, P))*
inquisitive	*askref(S, H, P)*	*not knowref(S, P)* *intend(S, knowref(S, P))* *intend(S, do(H, informref(H, S, P)))*
	askif(S, H, P)	*not knowif(S, P)* *intend(S, knowif(S, P))* *intend(S, do(H, informif(H, S, P)))*
requestive	*request(S, H, do(H, P))*	*intend(S, do(H, P))*
	pretell(S,H, P)	*knowref(S, P)* *knowsBetterRef(S, H, P)* *intend(S, do(S, informref(S, H, P)))* *intend(S, knowref(H, P))*
	testref(S, H, P)	*knowref(S, P)* *intend(S, do(H, informref(H, S, P)))*
	testif(S, H, P)	*knowif(S, P)* *intend(S, do(H, informif(H, S, P)))*

Figure 3.1: Linguistic intentions

First turn	Expected reply
informref	*accept*
informif	*accept*
askref	*informref*
askif	*informif*
request	*comply*
request	*refuse*
pretell	*askref*
testref	*informref*
testif	*informif*

Figure 3.2: Adjacency pairs

they say coherent, they must link their utterances to past and future ones using conventional strategies for communication. These strategies will, in turn, take any relevant beliefs and social conventions into account.

I divide the strategies for communication into five types: plan adoption, acceptance, challenge, repair, and closing.[7] In *plan adoption*, speakers simply choose an action that can be expected to achieve a desired illocutionary goal, given social norms and the discourse context. (The goal itself must originate within the speaker's non-linguistic planning mechanism.) With *acceptance*, speakers perform actions that have been elicited by a discourse partner. Because speakers are accountable for giving up an opportunity to repair or challenge an interpretation, this strategy also serves to accept the interpretations displayed by previous utterances. Actions that are used to *challenge* display non-acceptance of the presuppositions of previous action (*i.e.,* they address misconception or mishearing as opposed to misunderstanding) (Tsui 1991). Actions that are used to initiate *repair* display non-acceptance of an interpretation. Lastly, when speakers have no other goals to achieve nor expectations to address, they may attempt to *close* the conversation. Challenges and repairs require more inference than other conventional strategies; we will now consider them in greater detail.

A listener might deem a speaker's actions unacceptable if he disagrees with the attitudes that they presuppose of either participant. In addition to semantic sources, such attributions might derive from linguistic intentions or from more general presuppositions of need, authority, social status, etc. Consider Example 3.1 (from Tsui 1991). In this example, C *challenges* the presupposition that he heard and understood in T1. D then responds in T3 by repeating the challenged phrase and attempting to clarify its meaning.

Example 3.1 'Love Story'

T1 **D:** Why, sad novels can be fun to read. Tell me this, you know (who) Eric Segal was, *Love Story* was that—the book that was written in a bet.

T2 **C:** In a what?

T3 **D:** As a bet, as sort of, I bet you can't write a book that will become a best-selling novel.

Goals to challenge may arise when the condition of an adjacency-pair does not agree with a speaker's beliefs. They may also originate within the non-linguistic planning mechanism. Work by Webber and Mays (1983) and Moore (1989) on correcting misconceptions and by Carberry (1985, 1989) on steps in accepting a question discuss common types of challenges.

Repair typically involves recognizing a possible misunderstanding and then attempting to correct it.[8] Misunderstood speech acts are especially important to detect, because

[7]Currently, the model treats opening as a kind of plan adoption.

[8]Remler (1978) points out that not all repairs result from a perceived misunderstanding; sometimes

each turn creates expectations that are used to interpret the next one. These misunderstandings are also difficult to prevent, because they can result from many common sources, including intra-sentential ambiguity and mishearing. Fortunately, if there has been a misunderstanding and both speakers have been monitoring for inconsistencies in the discourse, it is likely that one of them will be able detect it. This is because eventually one of the speakers will likely say something that is inconsistent or incoherent with the other's view of the discourse.

In Section 2.3, we classified misunderstandings according to whether it is the same or different speaker who misunderstands that actually detects the misunderstanding. The former type, *self-misunderstandings*, can be detected when a hearer finds that an utterance violates his expectations, for example, because the linguistic intentions expressed by the speaker are inconsistent with those expressed earlier. The latter type, *other-misunderstandings*, typically occurs when a hearer recognizes that although a speaker's utterance appears unexpected, if one of his own acts had been interpreted differently, then her utterance would be expected. After a self-misunderstanding, if a hearer decides to reconstruct his own interpretation of an earlier turn, he will initiate a fourth-turn repair. After an other-misunderstanding, if a hearer attempts to change the speaker's interpretation, he will initiate a third-turn repair.

Interpretation is thus like abduction, because listeners must explain why, given their knowledge of rules for coherent interaction, a speaker might have said what she did. Listeners make assumptions about the nature of what the speaker is attempting to express and how it relates to what has been said before. It is thus similar to the abductive diagnosis and plan-recognition problems discussed in section 2.4.3, with utterances playing the role of "symptoms" and types of communicative strategies and goals playing the role of "faults" or "plans". Recognizing misunderstanding is also like abduction because listeners must explain why, given their knowledge of the symptoms of misunderstanding, a speaker might have said something that is apparently incoherent. They make assumptions attributing misunderstandings to themselves or the speaker. The similarity of this characterization to interpretation supports the claim that repair and interpretation both belong to the same set of core language abilities.

With this characterization, the unification of production, interpretation, and repair becomes natural, because of the similarity of these reasoning problems. To account for repairs, the model can be extended to include meta-plans for repair and axioms for identifying misunderstanding. Repair can be subsumed by interpretation and production.

---

people feign a misunderstanding or a lack of understanding in order to challenge a speaker's views, as in this example from Soskin (1963):

T1	**R:** The movie was pretty good, I mean, but I didn't see the stage show.
T2	**J:** It was what?
T3	**R:** I thought it was pretty good. I liked it.
T4	**J:** Oh, Roz, God!

3.3 Modeling expressed attitudes

It is important to keep separate the beliefs that speakers *act* as if they have during a course of a conversation from those they *actually* have. For example, a speaker might simplify or ignore certain facts that could interfere with the accomplishment of a primary goal (*cf.* Gutwin and McCalla 1992). Moreover, although participants normally continue to act in accordance with the beliefs they have expressed, they can recant in order to prevent or repair conversational breakdown. However, speakers tend to resist such changes. In fact, the need to recant may be a deciding factor in whether one chooses an explicit repair. Consider Example 3.2.[9]

Example 3.2 'Did you bring soap?'

T1	**Host:**	Did you bring any soap?
T2	**Guest:**	No.
T3	**Host:**	Well, the soap by the tub has lanolin in it. If you want I can get you another bar to use.
T4	**Guest:**	That's okay.
T5	**Host:**	I thought you said you didn't bring soap.
T6	**Guest:**	No, I didn't bring much. Not enough for you to use.
T7	**Host:**	Oh. Okay.

In this dialogue, Host is in the bathroom, speaking to Guest who is in the next room. Host knows that Guest is allergic to lanolin and could never use a soap that contained it. In the first turn, Guest presents himself as having no soap, but is forced to recant in T6. Host utters T1 as a *pretelling* (in this case, a preoffer). Thus, when Guest answers "No" in T2, Host interprets it as an *informif* expressing that Guest has not brought any soap and proceeds with the *offer* in T2. However, T4, a *reject-offer* by Guest, triggers a repair by Host, as it is inconsistent with Host's interpretation of T2. (T3 might also have triggered an overt repair of T2 by Guest, but the offer had already been made.) T5 questions the belief expressed in T2, leading Guest to recant.

Most discussions of belief and intention in the artificial intelligence literature do not distinguish between merely *expressing* belief and actually *believing*. Much of this discussion (see Cohen et al. 1990) has been concerned with defining idealized models of belief, belief change, and degrees of belief (*i.e.,* logics of possibility and conditional probability).[10] And most intentional accounts of discourse incorporate this work in some form to describe what happens when a speaker talks about a proposition. Some even

[9] This example was collected by the author.
[10] Thomason (1990) is an exception.

incorporate mutual belief, with all its added complexity. However, real belief involves notions of evidence, trustworthiness, and expertise, not accounted for in these models; it is not automatic. We must thus extend the intentional accounts to incorporate expressed, as well as actual, beliefs.

The model thus incorporates the notion of speakers being accountable for propositions they express, independent of both the truth values that the propositions might have in a particular world and whether or not participants believe them. Following Thomason (1990), we shall call the beliefs and intentions that speakers have expressed during a dialogue *suppositions*. In addition, we shall use the symbol *active* to name the attitude associated with participants being accountable for a supposition having been expressed.[11] For example, when a speaker performs a particular speech act, she activates the linguistic intentions associated with the act, along with a belief that the act has been done. Although these attitudes need not be sincere, active suppositions can affect the interpretation and production of utterances, because speakers choose to be consistent in the attitudes they express.

To represent activation, I am suggesting that discourse participants build a record of the suppositions expressed during a conversation. This record must also incorporate a history of repairs. The problem that repairs create for a model of discourse processing is that in order for speakers to negotiate understandings, they need to be able to reason about both the results of (possibly multiple) repairs and the differences between alternative interpretations under consideration. If interpretations were discarded after repair, then there would be no possibility of returning to an earlier interpretation if the other party disagreed with the repair. Moreover, different interpretations and even repairs themselves can become the object of discussion.

[11]Unlike beliefs, speakers do not normally distinguish their own suppositions from those of another (Stalnaker 1972; Thomason 1990).

Chapter 4

Abduction with Prioritized Theorist

In this chapter, we consider Prioritized Theorist (Poole, Goebel and Aleliunas 1987; Brewka 1989; van Arragon 1990), the framework that was used to develop the model presented in this thesis. (The axiomatization of the model is given in Chapter 6). Theorist typifies what is known as a "proof-based approach" to abduction (see Section 2.4.3), because it relies on a backward-chaining theorem prover to collect the assumptions that would be needed to prove a given set of observations and to verify their consistency. Theorist was selected because it supports both default and abductive reasoning, allowing one to specify preferences among defaults. Within Theorist one can represent both linguistic knowledge and the discourse context, and also model how speakers reason about their meta-plans and misunderstandings.

Theorist implements a full first-order clausal theorem prover in Prolog. It extends Prolog with a true negation symbol. It also generates the contrapositive forms of each clause. That is, each Theorist clause $\alpha \supset \beta$ is interpreted as $\{\beta \leftarrow \alpha, \neg\alpha \leftarrow \neg\beta\}$. A Prioritized Theorist reasoner can also assume any default d that the programmer has designated as a potential hypothesis, unless it can prove $\neg d$ from some fact or overriding hypothesis. This makes Prioritized Theorist nonmonotonic, because the addition of a new fact or overriding default may make less preferable ones underivable.

The reasoning algorithm uses model elimination (Loveland 1978; Stickel 1989; Umrigar and Pitchumani 1985) as its proof strategy. Like Prolog, it is a resolution-based procedure that chains backward from goals to subgoals, using rules of the form $goal \leftarrow subgoal_1 \wedge \cdots \wedge subgoal_n$ to reduce the goals to their subgoals. However, unlike Prolog, a model elimination procedure records each subgoal that occurs in the proof tree leading to the current goal and checks this list before searching its knowledge base for a relevant clause. If $\neg G$ appears on the list, then G is proven. As a result, unlike Prolog, Theorist can reason by cases.[1] As mentioned earlier, a Theorist reasoner can also make default

[1] For example, given the following theory, Theorist could prove c, whereas Prolog could not:
$a \vee b$
$c \leftarrow a$
$c \leftarrow b$

assumptions. Appendix F discusses in detail how Prioritized Theorist works.

A Theorist knowledge base consists of two sets of predicate calculus formulae, facts and defaults. Facts are propositions taken as true in the domain, whereas defaults correspond to the hypotheses of the domain (*i.e.,* formulae that may be assumed to be true when the facts alone are insufficient to explain some observation). For each formula in the knowledge base, Poole's compiler for Theorist, THCONSULT, will form all the contrapositives and build the Prolog predicates that implement the model elimination algorithm.

One can use a compiled Theorist knowledge base to locate assumptions that would need to be made to account for a given observation. To do this, one executes the command "EXPLAIN w.", where w is an arbitrary first-order formula. The top-level Theorist interpreter will then attempt a proof of w using the compiled knowledge base. The answer will indicate the success of the proof, any assumptions that were made to complete the proof, and bindings for any unbound variables.

In the rest of this chapter, we will discuss the syntax of Theorist, the explanation of clauses, and the role of priorities.

4.1 The syntax of Theorist

As we saw earlier, a Theorist knowledge base consists of two finite sets of statements, facts and defaults. Facts are given by the Theorist code "FACT w.", where w is a well-formed formula (wff). The wffs of Theorist mirror those of predicate calculus, including any legal combination of atomic formulae, joined by the standard connectives ($\vee, \wedge, \neg, \supset$, and $\equiv$). Variables are understood to be universally quantified in facts and defaults (but existentially quantified in an explanation, as defined below). For example, we could represent "Peter is a man" as

FACT $man(\mathbf{peter})$.

and "all bachelors are men" as:

FACT $bachelor(x) \supset man(x)$.

As in Prolog, variables are understood to be universally quantified (except during explanation, to be described below).[2]

A default can be given either by "DEFAULT (p, d)." or "DEFAULT $(p, d) : w$.", where p is a priority value, d is an atomic formula with only free variables as arguments, and w is a wff.[3] Thus, we could represent the default "Birds fly" as either a single default as in the following:

DEFAULT $(2, birdsFly(b)) : bird(b) \supset fly(b)$.

or, separately as a default and a fact:

DEFAULT $(2, birdsFly(b))$.

[2] More recently, the Artificial Intelligence Research Lab of the University of Alberta has implemented a version of Theorist that includes universal and existential quantifiers, but not priorities.

[3] Poole (1988) and van Arragon (1990) prove that one can restrict the syntax of both unprioritized and prioritized defaults to literals without any loss of generality. The construction involves replacing each nonliteral default $\alpha \in \Delta^P$ by a new default $name_\alpha$ of priority level P, and a fact $name_\alpha(X_1, \ldots, X_n) \supset \alpha$, where $X_1, \ldots, X_n$ are the variables of α and $name_\alpha(X_1, \ldots, X_n)$ does not appear in the original facts and defaults.

FACT $bird(b) \wedge birdsFly(b) \supset fly(b)$.
A priority value is an integer associated with a given default (and all ground instances of it), such that lower integers represent higher priorities. Formally, if $\mathcal{F}$ is the set of facts and Δ^p is the set of defaults with priority p, then an expression DEFAULT $(p, d) : w$ asserts that $d \in \Delta^p$ and $(d \Rightarrow w) \in \mathcal{F}$. That is, d specifies the assumable predicate; the wff specifies a fact that includes the assumption as a premise.

4.2 Theorist explanations

Explanation is a process akin to scientific theory formation—if a closed formula representing an observation is a logical consequence of the facts and the assumable defaults, then we say that Theorist can explain it:

DEFINITION 3: An *explanation* from the set of facts $\mathcal{F}$ and the sets of prioritized defaults $\Delta^1, \ldots, \Delta^n$ of a closed formula g is a set $\mathcal{F} \cup D^1 \cup \cdots \cup D^n$, where each D^i is a set of ground instances of elements of Δ^i, such that:

1. $\mathcal{F} \cup D^1 \cup \cdots \cup D^n$ is consistent
2. $\mathcal{F} \cup D^1 \cup \cdots \cup D^n \models g$
3. For all D^i such that $2 \leq i \leq n$, there is no $\mathcal{F} \cup D'^{\,1} \cup \cdots \cup D'^{\,i-1}$ that satisfies the priority constraints and is inconsistent with D^i.

Priority constraints require that no ground instance of $d \in \Delta^i$ can be in D^i if its negation is explainable with defaults usable from any Δ^j, $j < i$.

For example, suppose a Theorist knowledge base contains the following fact:
FACT $bachelor(x) \supset man(x)$.
Then, for any $\mathbf{a}$, Theorist could explain $man(\mathbf{a})$, given $bachelor(\mathbf{a})$, or $\neg bachelor(\mathbf{a})$ given $\neg man(\mathbf{a})$. Similarly, if the knowledge base contained the following default:
DEFAULT $(2, birdsFly(b)) : bird(b) \supset fly(b)$.
then, for any $\mathbf{b}$, Theorist could:

- Explain $birdsFly(\mathbf{b})$ by *assuming* it, unless it would be inconsistent to do so (*i.e.*, unless $bird(\mathbf{b})$ and $\neg fly(\mathbf{b})$).[4]
- Explain $fly(\mathbf{b})$, given $bird(\mathbf{b})$ (or an explanation of it) and assuming $birdsFly(\mathbf{b})$.
- Explain $bird(\mathbf{b}) \supset fly(\mathbf{b})$, assuming $birdsFly(\mathbf{b})$.
- Explain $\neg bird(\mathbf{b})$, given $\neg fly(\mathbf{b})$ (or an explanation of it) and assuming $birdsFly(\mathbf{b})$.
- Explain $\neg birdsFly(\mathbf{b})$, given (explanations of) $\neg fly(\mathbf{b})$ and $bird(\mathbf{b})$.

[4] It would also be inconsistent to assume $birdsFly(\mathbf{b})$ if there were a default of priority value 1 that could explain $\neg birdsFly(\mathbf{b})$.

Note that $bird(\mathbf{b})$ alone cannot be assumed, even if given $fly(\mathbf{b})$, because it was not defined to be a potential hypothesis.

Explanation corresponds directly to other forms of nonmonotonic reasoning—a goal G can be explained if and only if G is in a (prioritized) extension, that is, if G is in the set of logical consequences of a a set $\mathcal{F} \cup D^1 \cup \cdots \cup D^n$ such that for all $k, 1 \leq k \leq n$ $\mathcal{F} \cup D^1 \cup \cdots \cup D^k$ is consistent and maximal (with respect to set inclusion), where D^k is a subset of instances of Δ^k. (Brewka 1989; van Arragon 1990). If two defaults of different priority conflict, then only the stronger one is in the extension; the weaker one is deemed inconsistent. If two defaults of equal priority conflict, then there are multiple extensions, each of which is self-consistent, but inconsistent with each of the others. Thus, while Theorist might explain both G and $\neg G$ from a given theory, it could never explain $G \wedge \neg G$.

To explain a goal, Theorist may add a default to the list of proven premises if the default contains (unbound) variables, or if it can be shown to be consistent with the facts, any defaults used in the proof so far, and the priority constraints. The consistency of an assumption is shown if all attempts to prove its negation from facts, prior assumptions, and defaults of higher priority fail. That is, a default is assumable if Theorist cannot prove its negation.

Defaults with variables are a special case, because there may be constraints introduced later in the proof that would affect their consistency. To account for these constraints, the system must postpone consistency checking until it completes an explanation and replaces all remaining variables by new constants.

4.3 Priority constraints

Priorities enable one to state that one default is stronger than another, and thus can override it. Then, if two defaults conflict, the stronger one takes precedence. By using priorities, one can represent exceptions and preferences for certain explanations in the syntax of the language. For example, suppose that people with the flu typically have nausea, but people taking anti-nausea drugs typically do not experience nausea; and, suppose we know that a person Eric has the flu, but is on medication. One can represent this as follows:

DEFAULT $(2, \mathit{untreatedNausea}(x)) : flu(x) \supset nausea(x)$.
DEFAULT $(1, \mathit{treatedNausea}(x)) : drugged(x) \supset \neg nausea(x)$.
FACT $flu(\mathbf{eric})$.
FACT $drugged(\mathbf{eric})$.

Then, any attempt to explain $nausea(\mathbf{eric})$ would fail and attempts to explain $\neg nausea(\mathbf{eric})$ would suggest the single hypothesis: $\mathit{treatedNausea}(\mathbf{eric})$. (Without a difference in priority, both EXPLAIN $nausea(\mathbf{eric})$ and EXPLAIN $\neg nausea(\mathbf{eric})$ would succeed.)

The process of testing whether a default δ of priority m satisfies the priority constraints, given a set of accumulated assumptions D, involves the following steps:

1. Using D, try to explain $\neg\delta$ with defaults of priority $m-1$ (*i.e.,* the priority stronger than m). If $\neg\delta$ cannot be explained, then D is large enough and δ satisfies the priority constraints.

2. If $\neg\delta$ can be explained, δ has been *blocked*; however, if to explain $\neg\delta$ one had to make some new assumptions δ_p of priority p, try to *unblock* δ by explaining $\neg\delta_p$ using defaults of priority stronger than p. (If no new assumptions were required, then δ cannot be explained.)

3. If unblocking is successful, then add the new assumptions required to block δ_p to D to form D' and return to step 1 to verify that $\neg\delta$ cannot be explained with D'; if unblocking is unsuccessful, then δ cannot be explained.

This process terminates when either $\neg\delta$ cannot be explained (step 1), no new assumptions were needed to explain $\neg\delta$ (step 2), or δ_p could not be explained (step 3).

Chapter 5

The formal language

In this chapter, I introduce a first-order language with defaults in which I will formulate the model. This language will be used in Chapter 6 to give a precise account of the knowledge and inferences of discourse participants, as described in Chapter 3. Later I will also test the account by implementing participants' decision-making as theorem proving (see Chapter 7).

The axiomatization uses the notation of Prioritized Theorist (Brewka 1989; van Arragon 1990), a framework for default and abductive reasoning (see Chapter 4). This framework provides a language for representing facts and defaults, as well as a clausal theorem prover. The syntax for defaults in Prioritized Theorist requires that each default be given a priority level, but does not constrain the number of such levels. Clarity and efficiency require a minimal number. I use three, corresponding to *strong, weak,* and *very weak* defaults. Strong defaults can override other defaults, but can only be overridden by facts (*i.e.,* they are almost always true); weak defaults can override very weak defaults and can be overridden by either facts or strong defaults (*i.e.,* they are sometimes true); very weak defaults cannot override anything (*i.e.,* they are rarely true, except as a last resort). If axioms are written to specify preferences between competing defaults, the theorem prover will enforce the constraints automatically.

Similarly, the syntax for formulae permits arbitrary combinations of literals, joined by the connectives of predicate calculus. However, to describe conversation, I define a sublanguage with a fixed set of types and constraints on their use. These constraints clarify the intended meaning of each symbol and insure that its use is the same throughout the theory. Terms of the sublanguage come in six sorts: agents, turns, sequences of turns, actions, descriptions, and suppositions. I distinguish two important subtypes of actions: surface-level speech acts and discourse-level speech acts. I also distinguish two classes of suppositions: simple and compound. (Compound suppositions are formed from simple ones by the function symbol **and**.) For clarity, I use the following conventions for variables: s for agents, t for turns, ts for turn sequences, a for actions, d for descriptions, and p for suppositions, possibly with a subscript. I use a bold font for function names, and italic for predicates and variables. Below I will describe the representation of suppositions, speech acts, and turn sequences. Agents, turns, and descriptions are given by constants and will not be discussed in detail. (Appendix A gives the complete syntax.)

5.1 Suppositions

Suppositions are terms naming the propositions that are expressed when a speaker performs a speech act with associated linguistic intentions.[1] The following functional expressions are used to denote them:

- **do**(s, a) expresses that agent s has performed the action a;
- **mistake**(s, a_1, a_2) expresses that agent s has mistaken an act a_1 for act a_2;
- **intend**(s, p) expresses that agent s intends to achieve a situation described by the supposition p;
- **knowif**(s, p) expresses that the agent s knows whether the proposition named by supposition p is true;
- **knowref**(s, d) expresses that the agent s knows the referent of description d;
- **knowsBetterRef**(s_1, s_2, d) expresses that agent s_1 has "expert" knowledge about the referent of description d, so that if s_2 has a different belief about the referent, then s_2 is likely to be wrong;[2]
- **suspectThat**(s_1, p) expresses that agent s_1 suspects (but is uncertain) that the proposition p holds;
- **and**(p_1, p_2) expresses the conjunction of suppositions p_1 and p_2, where p_1 must be simple;
- **not** p expresses the negation of a simple supposition p.[3]

By themselves, suppositions have no truth value. Two suppositions are equivalent if and only if they are syntactically identical. We will make no special assumptions about the semantic relationship among different suppositions. (However, below I will define a notion of default compatibility.)

To capture the notion that speakers are normally consistent in the suppositions they choose to express, we need to know how different suppositions relate to each other. In particular, we need to know when two two simple suppositions are (in)compatible. However, because there is not a complete account of expressed intention and knowledge, the approach taken here will be to make compatibility a default and define axioms to exclude clearly incompatible cases.[4] It should be considered a place holder for an appropriate theory.

[1] For simplicity, I will also use them to give an explicit description of the contents of the participants' beliefs. This is sufficient because the examples do not require any reasoning about beliefs.

[2] This specialization is needed to capture the pragmatic force of *pretelling*, which says, in effect, "I'm going to tell you something that will surprise you. You might think you know, but you don't."

[3] The function **not** is distinct from boolean connective $\neg$. I use it to capture the supposition expressed by an agent who says something negative, *e.g.,* "I do not want to go."

[4] The danger in this approach is that it might miss some intuitively incompatible cases and hence some misunderstandings might not be detectable.

The most important property of compatibility is that suppositions should be consistent (*i.e.,* suppositions Q and $\mathbf{not}(Q)$ should be incompatible). Representing this property is straightforward:

DEFAULT $(2, compatible(p, q))$.

FACT $\neg compatible(p, \mathbf{not}(p))$.

FACT $\neg compatible(\mathbf{not}(p), p)$.

Compatibility of suppositions is complicated by expressions of intention. The particular type of intention expressed seems to be a future-directed one; that is, an intention by the speaker to change the discourse record. As a first approximation, I will require that intended changes be productive and that intended actions to not "obviously" lead to an inconsistency.[5] To capture this property, compatibility is defined asymmetrically, treating the first argument as "new" information and the second argument as "given". Then we can say that is incompatible for the content of a new intention to include the given information or for the linguistic intentions of a new intended action to be inconsistent with the given information. This is represented in Theorist as follows:[6]

FACT $\neg compatible(\mathbf{intend}(s_1, p), p)$.

FACT $lintention(a, p)$
$\wedge\ simplePart(\mathbf{not}(q), p)$
$\supset \neg compatible(\mathbf{intend}(s_1, \mathbf{do}(s_2, a)), q)$.

FACT $lintention(a, p)$
$\wedge\ simplePart(q, p)$
$\supset \neg compatible(\mathbf{intend}(s_1, \mathbf{do}(s_2, a)), \mathbf{not}(q))$.

The supposition of intention to **knowif** is also problematic, because it implicitly expresses disjunctive information—an intention to $\mathbf{knowif}(P)$ would be incompatible given either P or $\mathbf{not}(P)$. This is represented as follows:

FACT $\neg compatible(\mathbf{intend}(s_1, \mathbf{knowif}(s_2, p)), p)$.
FACT $\neg compatible(\mathbf{intend}(s_1, \mathbf{knowif}(s_2, p)), \mathbf{not}(p))$.
FACT $\neg compatible(\mathbf{intend}(s_1, \mathbf{knowif}(s_2, \mathbf{not}(p))), p)$.

[5] Another property to consider is ability—it would also be incompatible to intend an action if it were given that the intended actor is unable to do it.

[6] The *lintention* relation holds of an action and a supposition if performing the action expresses the linguistic intentions named by the supposition. (See Section 6.3.1.) The predicate *simplePart* holds of a pair of suppositions (P, Q) if and only if P is simple (*i.e.,* a supposition not formed from others using the function symbol **and**) and either: Q is simple and $P = Q$; or: $Q = \mathbf{and}(Q_1, \mathbf{and}(Q_2, \cdots, Q_n))$ and $P = Q_i$ for some $Q_i, 1 \leq i \leq n$.

5.2 Speech acts

For simplicity, I represent utterances as surface-level speech acts in the manner first used by Perrault and Allen (1980).[7] Following Cohen and Levesque (1985), I limit the surface language to the acts **surface-request**, **surface-inform**, **surface-informref**, and **surface-informif**. Example 5.1 shows the representation of the literal form of Example 2.6, the fourth-turn repair example. (I abbreviate "**m**" for "Mother", "**r**" for "Russ", and "**whoIsGoing**" for "who's going".)

Example 5.1 Theorist version of a fourth-turn repair

T1	**m:**	**surface-request(m, r, informif(r, m, knowref(r, whoIsGoing)))**
T2	**r:**	**surface-request(r, m, informref(m, r, whoIsGoing))**
T3	**m:**	**surface-inform(m, r, not knowref(m, whoIsGoing))**
T4	**r:**	**surface-informref(r, m, whoIsGoing)**

The theory also includes the discourse-level acts **inform, informif, informref, askref, askif, request, pretell, testref,** and **warn**, which I represent using a similar notation.[8] Although this set does not necessarily represent an exhaustive enumeration of the class, I suspect that a complete set need not be significantly larger. [9]

5.3 Turn sequences

A turn sequence represents the interpretations of the discourse that a participant has considered. They are characterized by the following three relations:

- A membership relation for turns, $turnOf(ts, t)$, holds if and only if t is a turn in the sequence ts;

[7] Other representation languages, such as one based on case semantics, would also be compatible with the approach and would permit greater flexibility. The cost of the increased flexibility would be increased difficulty in mapping surface descriptions onto speech acts; however, because less effort would be required in sentence processing, the total complexity of the problem need not increase. Using a more finely grained representation, one could reason about sentence type, particles, and prosody explicitly, instead of requiring the sentence processor to interpret this information (*cf.* Hinkelman 1989; Beun 1990).

[8] In the utterance language, a yes–no question is taken to be a **surface-request** to **informif** and a wh-question is taken to be a **surface-request** to **informref**. I then translate these request forms into the discourse-level actions **askif** and **askref** using the mapping relations given in Section 6.3.1. An alternative would be to identify them as **surface-askif** or **surface-askref** during sentence processing, as Hinkelman (1989) does.

[9] As with the supposition terms we considered earlier, speech act names that end with the suffix **-ref** take a description as an argument; speech act names that end with **-if** take a supposition. The act **inform**(s,p) asserts that the proposition is true. The act **informif**(s,p) asserts the truth value of the proposition named by p (*i.e.,* **informif** is equivalent to "**inform** $\vee$ **inform-not**").

- An ordering relation on turns, $succ(t_j, t_i, ts)$, holds if and only if $turnOf(ts, t_i)$, $turnOf(ts, t_j)$, t_j follows t_i in ts, and there is no t_k such that $turnOf(ts, t_k)$, $succ(t_k, t_i, ts)$, and $succ(t_j, t_k, ts)$;
- A focusing relation for turns, $focus(ts, t)$, holds if t is a distinguished turn upon which the sequence is focused.

We will call the turn sequence whose focus is the current turn the "discourse context". In order to consider previous states of the context, such as before a possible misunderstanding occurred, we also define a successor relation on turn sequences:

DEFINITION 4: A turn sequence $TS2$ is a *successor* to turn sequence $TS1$ if $TS2$ is identical to $TS1$ except that $TS2$ has an additional turn t that is not a turn of $TS1$ and t is the successor to the focused turn of $TS1$.

Chapter 6

The architecture of the model

In this chapter, we consider how interactions among social conventions and mental states can account for understanding, misunderstanding, and repair. In the architecture that I have formulated, producing an utterance is a default, deductive process of choosing both a speech act that satisfies an agent's communicative and interactional goals and an utterance that will be interpretable as this act in the current context. Utterance interpretation is the complementary (abductive) process of attributing communicative and interactional goals to the speaker by attributing to him or her a discourse-level form that provides a reasonable explanation for an observed utterance in the current context. Expectations deriving from social norms delimit the range of responses that can occur without additional explanation. The attitudes that speakers express provide additional constraints, because speakers are expected not to contradict themselves. I therefore attribute to each agent:

- A set $\mathcal{B}$ of prior assumptions about the beliefs and goals expressed by the speakers (including assumptions about misunderstanding).
- A set $\mathcal{M}$ of potential assumptions about misunderstandings and meta-planning decisions that agents can make to select among coherent alternatives.
- A theory $\mathcal{T}$ describing his or her linguistic knowledge, including principles of interaction and facts relating linguistic acts.

Definition 5: An *interpretation* of an utterance $\mathbf{u}$ to hearer $\mathbf{h}$ by speaker $\mathbf{s}$ in discourse context $\mathbf{ts}$ is a set M of instances of elements of $\mathcal{M}$, such that

1. $\mathcal{T} \cup \mathcal{B} \cup M$ is consistent
2. $\mathcal{T} \cup \mathcal{B} \cup M \models \mathit{utter}(\mathbf{s}, \mathbf{h}, \mathbf{u}, \mathbf{ts})$
3. $\mathcal{T} \cup \mathcal{B} \cup M$ satisfies the priority constraints.[1]

[1]That is, $\mathcal{T} \cup \mathcal{B} \cup M$ is not in conflict with any stronger defaults that might apply.

DEFINITION 6: It would be *coherent* for **s** to utter **u** in discourse context **ts** if **u** is a solution to the following default reasoning problem:

$$\mathcal{T} \cup \mathcal{B} \cup M^{meta} \vdash (\exists u)\ utter(\mathbf{s},\ \mathbf{h},\ u,\ \mathbf{ts})$$

where M^{meta} is a set of assumptions about meta-planning decisions in $\mathcal{M}$, such that

1. $\mathcal{T} \cup \mathcal{B} \cup M^{meta}$ is consistent
2. $\mathcal{T} \cup \mathcal{B} \cup M^{meta} \models utter(\mathbf{s}, \mathbf{h}, \mathbf{u}, \mathbf{ts})$
3. $\mathcal{T} \cup \mathcal{B} \cup M^{meta}$ satisfies the priority constraints.

In addition, acts of interpretation and generation update the set of beliefs and goals assumed to be expressed during the discourse. The current formalization focuses on the problems of identifying how an utterance relates to a context and whether it has been understood. The update of expressed beliefs is handled in the implementation, but outside the formal language.[2]

6.1 A model of active suppositions

The first component of the model is $\mathcal{B}$, the set of prior assumptions about the beliefs and goals expressed by the speakers (*i.e.,* of both participants' suppositions). I assume that an agent will maintain a record of these suppositions, indexed by the turns in which they were expressed. To represent the dialogue as a whole, including misunderstandings, I introduce the notion of activation of a supposition with respect to a turn sequence. Beliefs and intentions that participants express during a turn of a sequence TS become and remain active in all sequences that are successors to TS, unless they are explicitly refuted.

As mentioned above, suppositions are expressed as a result of acts of interpretation and generation. In the model, expression is treated as a primitive not axiomatized within the theory; however, the implementation captures the following definition:

DEFINITION 7: If, during turn T, a speech act A was performed by agent S and L is the supposition denoting the linguistic intentions associated with A, then

- if L is simple, we say that L was *expressed* during turn T.
- if L is complex, such that $L = \mathbf{and}(L_1, \mathbf{and}(L_2, \cdots, L_n)$ where each $L_i, 1 \leq i \leq n$, is simple, we say of each L_i that it was *expressed* during turn T.

Also, if A was performed coherently, then $\mathbf{do}(S, A)$ is expressed; if speaker S manifested a misunderstanding of speech acts $A_{intended}$ and $A_{observed}$, then $\mathbf{mistake}(S, A_{intended}, A_{observed})$ is expressed.

[2] A related concern is how an agent's beliefs might change after an utterance has been understood as an act of a particular type. Although we have nothing new to add here, Perrault (1990) shows how default logic might be used to address this problem.

Thus, the notion of expression corresponds to a speaker acting as if the denoted proposition were true. For example, an agent **mary** performing an **informref(mary, bob, theTime)** expresses the supposition **do(mary, informref(mary, bob, theTime))** and the following linguistic intention:

and(knowref(mary, theTime),
intend(mary, knowref(bob, theTime)))

Activation is defined in terms of expressed supposition:

DEFINITION 8: If, according to the interpretation of the conversation represented by turn sequence TS with focused turn T, the supposition P was expressed during turn T, we say that P becomes *active* with respect to that interpretation.

Expressed suppositions are represented by a set of facts of the form $expressed(P, T)$ or $expressedNot(P, T)$ where P is a simple, unnegated supposition and T is a turn.[3]

This is represented in Theorist by the following axioms:

FACT $expressed(p, t) \land focus(ts, t)$
$\supset active(p, ts).$

FACT $expressedNot(p, t) \land focus(ts, t)$
$\supset active(\mathbf{not}(p), ts).$

As mentioned above, if supposition P is active within a sequence TS, it will remain active in successors of TS until $\mathbf{not}(P)$ is expressed. This is represented by the following Theorist facts:[4]

FACT $expressed(p, t) \land focus(ts, t)$
$\supset \neg activationPersists(\mathbf{not}(p), t).$

FACT $expressedNot(p, t) \land focus(ts, t)$
$\supset \neg activationPersists(p, t).$

DEFAULT $(1, activationPersists(p, t)):$
$active(p, ts_i)$
$\land\ successorTS(ts_j, ts_i)$
$\land\ focus(ts_j, t)$
$\supset active(p, ts_j).$

In the absence of any other information, a supposition is inactive. This is represented by a weak default, so that it can be overridden by the persistence of a previously expressed supposition. (Persistence itself can only be overridden by the expression of a supposition, so it is represented by a maximum-strength default.)

[3]The intended meaning of $expressedNot(P,T)$ is that during turn T speakers have acted as if the supposition P were false. Although $expressed(\mathbf{not}(P), T)$ and $expressedNot(P,T)$ represent the same state of affairs, the latter expression avoids infinite recursion by Theorist.

[4]The default $activationPersists$ has been given priority 1 in order to make persistence overridable only by facts (*i.e.,* expressed suppositions).

6.2 Possible hypotheses

The second component of the model is $\mathcal{M}$, the set of potential assumptions about misunderstandings and meta-planning decisions. This is given by a set of Theorist defaults: *intentionalAct, acceptance, adoptPlan, makeFourthTurnRepair, makeThirdTurnRepair, reconstructTS, otherMisunderstanding, selfMisunderstanding*, and *done*.
The theorem prover may assume ground instances of any of these predicates if they are consistent with all facts and any defaults with higher priority. As mentioned in Chapter 5, each of these defaults will have one of three priority values, *strong*, *weak*, or *very weak*. The strongest level is reserved for attitudes about beliefs and suppositions. Assumptions about expectations (*i.e., expectedReply, acceptance, makeFourthTurnRepair*, and *makeFourthTurnRepair*) are given as weak defaults. Assumptions about unexpected actions or interpretations (*i.e., adoptPlan, challenge, done, selfMisunderstanding*, and *otherMisunderstanding*) are given as very weak defaults, so that axioms can be written to express a preference for expected analyses when there is an ambiguity. We will consider each of these predicates in greater detail when we discuss the third component of the model.

6.3 A speaker's theory of language

The third component of the model is $\mathcal{T}$, a speaker's theory of language and communicative interaction. This theory includes strategies for expressing beliefs and intentions, for displaying understanding, and for identifying when understanding has broken down. The strategies for displaying understanding suggest performing speech acts that have an identifiable, but defeasible, relationship to other speech acts in the discourse (or to the situation). Misunderstandings are recognized when an utterance is inconsistent or incoherent; strategies for repair suggest reanalyzing previous utterances or making the problem itself public.

6.3.1 Relations on linguistic knowledge

There are three important linguistic knowledge relations: *decomp*, *lintention*, and *lexpectation*. They are shown as circles in Figure 6.1; the boxes in the figure are the objects that they relate. The *decomp* relation links surface-level forms to the discourse-level forms that they might accomplish in different contexts. It corresponds to the *body* relation in STRIPS-based approaches.[5] The *lexpectation* relation (for "linguistic expectation") relates each act to the acts that might be expected to follow. It also provides a basis for several well-formed deviations from expected behavior. The *lintention* relation (for "linguistic intention") relates discourse acts to the linguistic intentions that they conventionally express. These intentions help constrain the range of expected acts.

The arguments to these relations can be suppositions, descriptions, and other actions; we shall now consider the representation of these relations in Theorist.

[5] Pollack (1986) calls this the "is-a-way-to" relation.

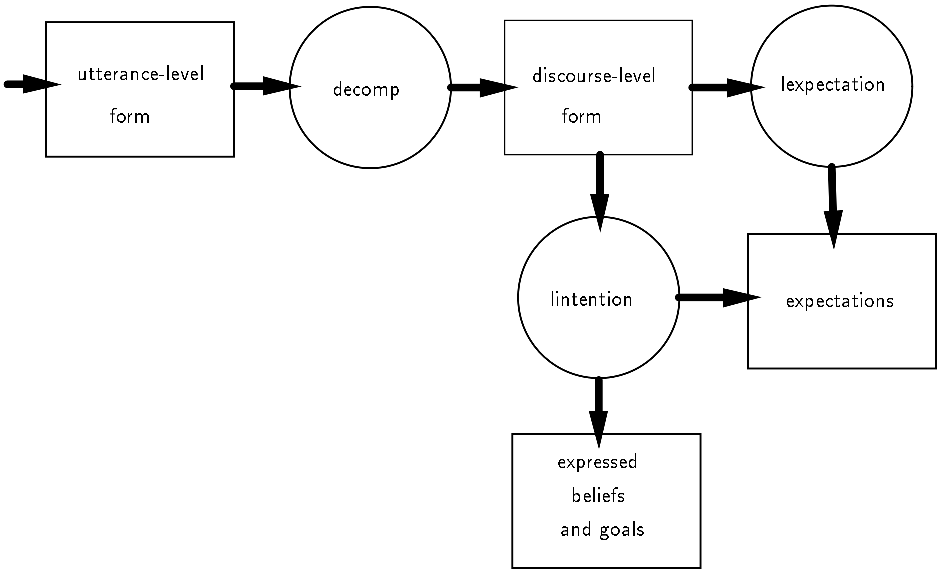

Figure 6.1: How the knowledge relations fit together

The *decomp* relation

The *decomp* relation is represented as a set of facts of the form:
FACT $decomp(A1, A2)$.
The proposition $decomp(A1, A2)$ holds if producing surface act $A1$ might, under some circumstances, accomplish discourse-level act $A2$. For example, the following axiom says that a **surface-inform** can accomplish an **inform**:
FACT $decomp(\mathbf{surface\text{-}inform}(s_1, s_2, p), \mathbf{inform}(s_1, s_2, p))$.
The same surface form might also accomplish several other discourse acts. For example, a speaker might use a surface-level yes–no question of the form "Do you know ...?" to perform either an **askif,** an **askref**, or a **pretell**.

FACT $decomp(\mathbf{surface\text{-}request}(s_1, s_2, \mathbf{informif}(s_2, s_1, p)),$
$\mathbf{askif}(s_1, s_2, p))$.

FACT $decomp(\mathbf{surface\text{-}request}(s_1, s_2, \mathbf{informif}(s_2, s_1, \mathbf{knowref}(s_2, d))),$
$\mathbf{askref}(s_1, s_2, d))$.

FACT $decomp(\mathbf{surface\text{-}request}(s_1, s_2, \mathbf{informif}(s_2, s_1, \mathbf{knowref}(s_2, d))),$
$\mathbf{pretell}(s_1, s_2, d))$.

For example suppose that Mary said "Do you know the time?" to Fred, which has the surface form:
surface-request(**m**, **f**, **informif**(**f**, **m**, **knowref**(**f**, **theTime**)))
Then, she might be performing any of the following discourse-level acts:
askif(**m**, **f**, **knowref**(**f**, **theTime**))
askref(**m**, **f**,**theTime**)
pretell(**m**, **f**, **theTime**)

This representation is neutral about whether a given surface form may perform different discourse acts at the same time.[6] Appendix B contains the elements of the relation for the complete set of speech acts.

The *lintention* relation

Associated with each speech act are the linguistic intentions that performers of the action express. I call this association the *lintention* relation. In Theorist, I represent it as a set of facts of the form: FACT $lintention(A, P)$, where $lintention(A, P)$ holds if a speaker performing act A expresses the proposition named by P. For example, when a speaker performs an **informref**, she expresses that she intends that the recipient believe that she knows the referent of the given description and that she intends the recipient to know it as well. This corresponds to the following Theorist fact:

$$
\begin{array}{l}
\text{FACT } lintention(\textbf{informref}(s_1, s_2, d), \\
\qquad\qquad \textbf{and}(\textbf{knowref}(s_1, d), \textbf{intend}(s_1, \textbf{knowref}(s_2, d)))).
\end{array}
$$

The linguistic intentions of **askref, askif,** and **pretell,** are analogous.

$$
\begin{array}{l}
\text{FACT } lintention(\textbf{askref}(s_1, s_2, d), \\
\qquad\qquad \textbf{and}(\textbf{not}\ \ \textbf{knowref}(s_1, d), \\
\qquad\qquad\quad \textbf{and}(\textbf{intend}(s_1, \textbf{knowref}(s_1, d)), \\
\qquad\qquad\qquad \textbf{intend}(s_1, \textbf{do}(s_2, \textbf{informref}(s_2, s_1, d)))))).
\end{array}
$$

$$
\begin{array}{l}
\text{FACT } lintention(\textbf{askif}(s_1, s_2, p), \\
\qquad\qquad \textbf{and}(\textbf{not}\ \textbf{knowif}(s_1, p), \\
\qquad\qquad\quad \textbf{and}(\textbf{intend}(s_1, \textbf{knowif}(s_1, p)), \\
\qquad\qquad\qquad \textbf{intend}(s_1, \textbf{do}(s_2, \textbf{informif}(s_2, s_1, p)))))).
\end{array}
$$

$$
\begin{array}{l}
\text{FACT } lintention(\textbf{pretell}(s_1, s_2, d), \\
\qquad\qquad \textbf{and}(\textbf{knowref}(s_1, d), \\
\qquad\qquad\quad \textbf{and}(\textbf{knowsBetterRef}(s_1, s_2, d), \\
\qquad\qquad\qquad \textbf{and}(\textbf{intend}(s_1, \textbf{knowref}(s_2, d)) \\
\qquad\qquad\qquad\quad \textbf{intend}(s_1, \textbf{do}(s_1, \textbf{informref}(s_1, s_2, d))))))).
\end{array}
$$

Although these intentions do not constrain what speakers really believe, they do constrain action, because speakers expect each other to be consistent in the attitudes they express. Otherwise, the dialogue will be incoherent. Thus, speakers must maintain some sort of model of the linguistic intentions expressed in a conversation. (See Section 6.1).

Appendix C includes the Theorist representations of the linguistic intentions for the complete set of speech acts.

---

[6] Work by Sanford and Roach (1987) attempts to account for speakers performing several acts at once, but they have not addressed the problem of deciding when two acts are incompatible.

The *lexpectation* relation

Linguistic expectations are represented with the *lexpectation* relation. This relation is represented as a set of facts of the form: FACT $lexpectation(A1, P, A2)$. The proposition $lexpectation(A1, P, A2)$ holds if, when condition P is believed, act $A2$ is the normal expected reply to act $A1$. The supposition P only specifies conditions that distinguish alternative expected replies. For example, the following facts encode the expectation relation between asking and telling:[7]

$$
\begin{aligned}
\text{FACT } lexpectation(&\mathbf{do}(s_1, \mathbf{askref}(s_1, s_2, d)),\\
&\mathbf{knowref}(s_2, d),\\
&\mathbf{do}(s_2, \mathbf{informref}(s_2, s_1, d))).
\end{aligned}
$$

"A speaker s_1 can expect that making an **askref** of d to s_2 will result in s_2 telling s_1 the referent of d, if s_2 knows it.

$$
\begin{aligned}
\text{FACT } lexpectation(&\mathbf{do}(s_1, \mathbf{askref}(s_1, s_2, d)),\\
&\mathbf{not\ knowref}(s_2, d),\\
&\mathbf{do}(s_2, \mathbf{inform}(s_2, s_1, \mathbf{not\ knowref}(s_2, d)))).
\end{aligned}
$$

"A speaker s_1 can expect that making an **askref** of d to s_2 will result in s_2 telling s_1 that s_2 does not know the referent of d, if s_2 does not know it."

The facts that capture the adjacency pairs *pretell–askref* and *askif–informif* are similar:

$$
\begin{aligned}
\text{FACT } lexpectation(&\mathbf{do}(s_1, \mathbf{pretell}(s_1, s_2, d)),\\
&\mathbf{knowsBetterRef}(s_1, s_2, d),\\
&\mathbf{do}(s_2, \mathbf{askref}(s_2, s_1, d))).
\end{aligned}
$$

"A speaker s_1 can expect that making a **pretelling** of d to s_2 will result in s_2 asking about d, if s_1 is considered to be more expert about d than s_2."

$$
\begin{aligned}
\text{FACT } lexpectation(&\mathbf{do}(s_1, \mathbf{askif}(s_1, s_2, d)),\\
&\mathbf{knowif}(s_2, d),\\
&\mathbf{do}(s_2, \mathbf{informif}(s_2, s_1, d))).
\end{aligned}
$$

"A speaker s_1 can expect that making an **askif** of p to s_2 will result in s_2 telling s_1 p if s_2 knows p."

[7] It is controversial whether the adjacency pair *askref* followed by *inform-not-knowref* is a valid one. By some accounts, such questions are taken to presuppose that the hearer does know the answer and hence a response to the contrary could be considered a *challenge* of the presupposition (Tsui 1991)(see Section 3.2).

These rules are useful for identifying the speech acts that speakers might use to achieve their goals in a given situation.

Appendix D includes the Theorist representations of the *lexpectation* rules for the complete set of speech acts.

6.3.2 Expectation

In the model, actual expectation originates from social norms captured by the *lexpectation* relation. The linguistic intentions that have been expressed during a turn sequence (along with speakers' beliefs) constrains these expectations, because speakers are not expected to express beliefs or intentions that are inconsistent. Expectation delimits the range of responses that can occur without requiring a deeper explanation.

I introduce the predicate $lintentionsOk(P, TS)$ to capture the notion of (in)compatibility between a supposition and a turn sequence. (It is based on the notion of compatibility of simple suppositions given in Section 5.1.) A supposition $P = \mathbf{and}(P_1, \mathbf{and}(P_2, \cdots, P_n))$ is incompatible with a turn sequence TS if some simple $P_i, 1 \leq i \leq n$, is incompatible with a (simple) supposition that is active in TS:

FACT $\quad simplePart(p_{part}, p_{whole})$
$\quad \wedge \neg compatible(p_{part}, p_{active})$
$\quad \wedge\ active(p_{active}, ts)$
$\quad \supset \neg lintentionsOk(p_{whole}, ts)$.

We can now capture the notion of "expectation" in the following definition:

DEFINITION 9: A speaker *S1* is *expected* to do action R in dialogue *TS* whenever there is an action A that is active in *TS* (because it was performed earlier), R as the normal expected reply to A, and the linguistic intentions of R are compatible with the active suppositions of *TS*.

The following Theorist axioms capture this definition:[8]

DEFAULT $(2, expectedReply(p_{do}, p_{condition}, \mathbf{do}(s_1, a_{reply}), ts))$:
$\quad active(p_{do}, ts)$
$\quad \wedge\ lexpectation(p_{do}, p_{condition}, \mathbf{do}(s_1, a_{reply}))$
$\quad \wedge\ believe(s_1, p_{condition})$
$\quad \supset expected(s_1, a_{reply}, ts)$.

FACT $\quad lintention(a, p_l)$
$\quad \wedge \neg lintentionsOk(p_l, ts)$
$\quad \supset \neg expectedReply(p_{do}, p_{condition}, \mathbf{do}(s, a), ts)$.

The model also requires a subjunctive form of expectation to reason about expectations that would arise as a result of future actions (*e.g.,* plan adoption) or that must be

[8] Recall that *lintentionsOk* depends on *active*. Although, like *expectedReply*, *active* is a default, *active* will take precedence over *expectedReply*, because it has been given a higher priority on the assumption that memory for suppositions is stronger than expectation.

considered when evaluating a potential repair. This type of expectation differs from the type defined above in that it depends on the real beliefs of the agent performing the first (rather than the second part) of an adjacency pair and it does *not* depend on the activity of any suppositions or actions.

FACT $$\begin{aligned} & lexpectation(\mathbf{do}(s_1, a_1), p, \mathbf{do}(s_2, a_2)) \\ & \wedge\ believe(s_1, p) \\ & \equiv wouldExpect(s_1, a_1, a_2). \end{aligned}$$

6.3.3 Beliefs and goals

In the model, beliefs must be given explicitly, as facts or defaults, using the predicate *believe*. The intended meaning of this predicate is that *believe* is true of an agent s_i and a supposition p only if the proposition named by p is among the (actual) beliefs of s_i. I assume no special properties of belief, as the examples that are considered here do not require any.[9]

The many complexities associated with representing intention have also been ignored in the present model. I use the predicate $hasGoal(s, \psi, ts)$, which corresponds (roughly) to Allen's (1984) predicate "IS-GOAL-OF" or to the "achievement goals" of Cohen and Levesque (1990a), to express that an agent s intends to achieve a turn sequence ts in which the supposition ψ is active. Currently, a speaker's own intentions must be provided in the initial background theory. The implementation does allow a speaker to explain by default the goals of another. A more complete account would include a formal theory linking agents' beliefs, intentions, and plans.

6.3.4 A characterization of interpretation and production

Utterance production involves reasoning from goals and expectations to appropriate linguistic actions. Conversely, to interpret an utterance, a speaker reasons backward from her observations to find goals and expectations that explain them. The most significant difference between these two processes (and one reason that interpretation is abductive) is that while generation considers only coherent actions, interpretation must also account for misunderstandings. The model uses the predicate *shouldTry* for discourse actions that are coherent and the predicate *try* for actions that are explainable.

The meta-plans encode strategies for selecting an appropriate act, that is, they characterize the predicate *shouldTry*. The antecedents of these axioms refer to expectations. In addition, in order to preserve discourse coherence, they require that either the linguistic intentions of suggested actions be compatible with the context or that there be some public acknowledgement of the discrepancy. (Only the former case is addressed presently.)

Along with the meta-plans, a speaker's linguistic theory includes diagnostic axioms that characterize two types of speech act misunderstandings, self-misunderstandings and

[9] It would have been possible to characterize actual belief using an appropriate set of axioms, such as those defining a weak S4 modal logic. However, current formalizations do not seem to account for the context-sensitivity of speakers' beliefs. Section 8.3 discusses some of these problems in greater detail.

other-misunderstandings. The first type become apparent to the participant who has misunderstood, while the second become apparent to the other participant. The antecedents of these axioms refer to ambiguities and inconsistencies with expressed linguistic intentions, as well as expectations. These diagnostic axioms, together with an axiom for intentional action that depends on *shouldTry*, characterize the predicate *try*. See Figure 6.2.

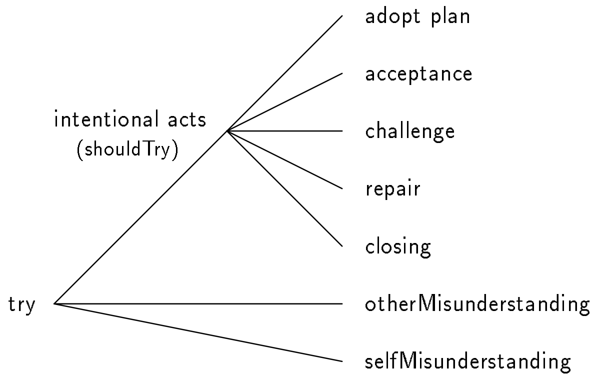

Figure 6.2: Summary of utterance processing relationships

The following axioms relate the predicates try and $shouldTry$ to the observation of utterances:

DEFAULT $(1, intentionalAct(s_1, s_2, a, ts))$:
$shouldTry(s_1, s_2, a, ts)$
$\supset try(s_1, s_2, a, ts)$.

DEFAULT $(3, pickForm(s_1, s_2, a_{surfaceForm}, a, ts))$:[10]
$decomp(a_{surfaceForm}, a)$
$\wedge\ try(s_1, s_2, a, ts)$
$\supset utter(s_1, s_2, a_{surfaceForm}, ts)$.

According to the characterization given above, an interpretation of an utterance **u** to hearer **h** by speaker **s** in discourse context **ts**, is a solution to the following abduction problem:

$\mathcal{T} \cup \mathcal{B} \cup M \vdash utter(\mathbf{s}, \mathbf{h}, \mathbf{u}, \mathbf{ts})$

where $\mathcal{T}$ is the linguistic theory, $\mathcal{B}$ is the prior assumptions about expressed beliefs and goals, M is a set of new assumptions about default meta-plans or misunderstandings. This corresponds to the Theorist explanation problem:

EXPLAIN $utter(\mathbf{s1}, \mathbf{s2}, \mathbf{u}, \mathbf{ts})$.

[10] Currently the model does not discriminate between equally acceptable alternatives. The predicate *pickForm* is an oracle that allows one to test different choices. Because it is represented as a very weak default, it can be overridden by user input, but will not influence other defaults. A user model could also be incorporated to rank the different interpretations (van Arragon 1990).

Similarly, any assignment to variable a_s that proves the following would be a coherent utterance to generate:

$$\mathcal{T} \cup \mathcal{B} \cup M \vdash (\exists a_s)\ \mathit{shouldTry}(\mathbf{s},\ \mathbf{h},\ a_d\ ,\ \mathbf{ts}) \wedge \mathit{decomp}(a_s, a_d)$$

This corresponds to the following problem in Theorist:

EXPLAIN $\mathit{shouldTry}(\mathbf{s1}, \mathbf{s2}, a_d, \mathbf{ts}) \wedge\ \mathit{decomp}(a_s, a_d)$.

Theorist identifies solutions to these problems by reasoning backward from the goal, making default assumptions as necessary.

Meta-plans

As mentioned above, meta-plans describe the range of coherent behavior. The following tables, 6.1–6.8, give each of these axioms in detail.

Table 6.1:

Name	Plan adoption
Purpose	Introducing a new goal
Axiom	DEFAULT $(3, adoptPlan(s_1, s_2, a_1, a_2, ts))$: $hasGoal(s_1, \mathbf{do}(s_2, a_2), ts)$ $\wedge\ wouldExpect(s_1, \mathbf{do}(s_1, a_1), \mathbf{do}(s_2, a_2), ts)$ $\supset shouldTry(s_1, s_2, a_1, ts)$. FACT $active(\mathbf{do}(s_2, a_2), ts)$ $\supset \neg adoptPlan(s_1, s_2, a_1, a_2, ts)$. FACT $lintention(a, p)$ $\wedge\ \neg lintentionsOk(p, ts)$ $\supset \neg adoptPlan(s, s_2, a, a_2, ts)$.
Summary	Speaker s_1 should do action a_1 in discourse ts when: 1. s_1 wants speaker s_2 to do action a_2; 2. s_1 would expect a_2 to follow an action a_1; 3. s_1 may adopt the plan of performing a_1 to trigger a_2 (*i.e.,* a_2 has not already been performed and the linguistic intentions of a_1 are compatible with ts).
Example	If it were true that $hasGoal$(**susan,do(jim,informif(jim,susan,hasFood(jim))),ts0**) $\wedge wouldExpect$(**do(susan,askif(susan,jim,hasFood(jim))),** **do(jim,informif(jim,susan,hasFood(jim))),ts0**), $\wedge lintentionsOk$(**askif(susan,jim,hasFood(jim)), ts0**) and it was assumed that $adoptPlan$(**susan,jim,askif(susan,jim,hasFood(jim)),** **informif(jim,susan,hasFood(jim)),ts0**), then $shouldTry$(**susan,jim,askif(susan,jim,hasFood(jim)), ts0**). For Susan to expect the **informif** requires her to have a *lexpectation* for an *askif–informif* adjacency pair and that its conditions match her real beliefs. Compatibility with the discourse (*lintentionsOk*) requires that the linguistic intentions of **askif** are compatible with the other suppositions that have been expressed.

Table 6.2:

Name	Acceptance
Purpose	Producing an expected reply
Axiom	DEFAULT $(2, acceptance(s_1, a_{reply}, ts)) :$ $expected(s_1, a_{reply}, ts)$ $\supset shouldTry(s_1, s_2, a_{reply}, ts).$ FACT $active(\mathbf{do}(s_1, a), ts)$ $\supset \neg acceptance(s_1, a, ts).$
Summary	Speaker s_1 should do action a_{reply} in discourse ts when: 1. s_1 expects a_{reply} to occur next and 2. s_1 may accept the interpretation corresponding to ts.

Table 6.3:

Name	Fourth-turn repair
Purpose	Recovering from one's own misunderstanding
Axiom	DEFAULT $(2, makeFourthTurnRepair(s_1, s_2, a_{reply}, ts)) :$ $active(\mathbf{mistake}(s_1, a_{intended}, a_{observed}), ts)$ $\wedge\ reconstruction(a_{observed}, a_{intended}, ts, ts_{reconstructed})$ $\wedge\ expected(s_1, a_{reply}, ts_{reconstructed})$ $\supset shouldTry(s_1, s_2, a_{reply}, ts).$ FACT $active(\mathbf{do}(s_1, a), ts)$ $\supset \neg makeFourthTurnRepair(s_1, s_2, a, ts, ts_{reconstructed}).$
Summary	Speaker s_1 should do action a_{reply} in discourse ts when: 1. s_1 has mistaken an instance of act $a_{intended}$ as an instance of act $a_{observed}$. 2. A reconstruction of the discourse is possible. 3. s_1 would expect to do a_{reply} in this reconstruction. 4. And, s may perform a fourth-turn repair.

Table 6.4:

Name	Third-turn repair
Purpose	Recovering from another speaker's misunderstanding
Axiom	DEFAULT $(2, makeThirdTurnRepair(s_1, s_2, a_{intended}, ts)):$ $active(\mathbf{mistake}(s_2, a_{intended}, a_{observed}), ts)$ $\wedge\ a = \mathbf{inform}(s_1, s_2, \mathbf{intend}(s_1, \mathbf{do}(s_1, a_{intended})))$ $\supset shouldTry(s_1, s_2, a, ts).$ FACT $wouldExpect(s_1, \mathbf{do}(s_1, a_{intended}), \mathbf{do}(s_2, a_{reply}))$ $\wedge\ active(\mathbf{do}(s_2,\ a_{reply}),\ ts)$ $\supset \neg makeThirdTurnRepair(s_1, s_2, a_{intended}, ts).$ FACT $\neg lintentionsOk(\mathbf{intend}(s_1, \mathbf{do}(s_1, a_{intended})), ts)$ $\supset \neg makeThirdTurnRepair(s_1, s_2, a_{intended}, ts)$
Summary	Speaker s_1 should tell speaker s_2 that she intended to perform $a_{intended}$ in discourse ts if: 1. s_2 has apparently mistaken an instance of act $a_{intended}$ for act $a_{observed}$; and 2. s_1 may perform a third-turn repair (*i.e.*, the act that s_1 would expect to follow her intended action has not already been done and it would be consistent for s_1 to tell s_2 that she intended to perform $a_{intended}$).

Table 6.5:

Name	Challenge
Purpose	Challenging an action
Axiom	DEFAULT $(3, challenge(s_1, a_1, a_2, ts))$: $active(\mathbf{do}(s_2, a_1), ts)$ $\wedge\ lexpectation(\mathbf{do}(s_2, a_1), p, \mathbf{do}(s_1, a_2))$ $\wedge\ p \neq \mathbf{true}$ $\wedge\ believe(s_1, \mathbf{not\ knowif}(s_2, \mathbf{not}\ p))$ $\supset shouldTry(s_1, s_2, \mathbf{inform}(s_1, s_2, \mathbf{not}\ p), ts)$. FACT $lexpectation(\mathbf{do}(s_2, a_1), p, \mathbf{do}(s_1, a_2))$ $\neg lintentionsOk(\mathbf{inform}(s_1, s_2, \mathbf{not}\ p), ts)$ $\supset \neg challenge(s_1, a_1, a_2, ts)$. FACT $lexpectation(\mathbf{do}(s_2, a_1), p, \mathbf{do}(s_1, a_2))$ $\wedge\ active(\mathbf{do}(s_1, a_2), ts)$ $\supset \neg challenge(s_1, a_1, a_2, ts)$.
Summary	Speaker s_1 should inform s_2 that p does not hold when: 1. Speaker s_2 has performed an action a_1; 2. action a_2 might be expected, according to the *lexpectation* relation; 3. s_1 believes that s_2 is unaware that the condition p for this expectation does not hold; and 4. s_1 may challenge a_1 (*i.e.,* the challenge is consistent with the discourse and othe expected reply has not already been performed).

Table 6.6:

Name	Closing
Purpose	Terminating a dialogue
Axiom	DEFAULT $(3, done(s_1, s_2, ts))$: $shouldTry(s_1, s_2, \mathbf{closing}(s_1, s_2), ts)$. FACT $\neg(shouldTry(s_1, s_2, a, ts)$ $\wedge\ shouldTry(s_1, s_2, \mathbf{closing}(s_1, s_2), ts)$ $\wedge\ a \neq \mathbf{closing}(s_1, s_2))$.
Summary	A speaker may end the conversation by performing a closing, unless there is some other act she should do.

6.3.5 Strategies for resolving incoherence

To account for utterances that fail to cohere with the situation or the discourse, the theory includes two diagnostic axioms that characterize how the understanding of speech acts might fail. The first axiom handles the case in which an observed inconsistency of s_1 performing a_{new} might manifest s_2's misinterpretation of an earlier act by s_1. Such mistakes are possible when the surface form of the earlier act might be used to accomplish either $a_{observed}$ or $a_{intended}$. (See Table 6.7.) Similarly, the observed inconsistency of s_1 performing a_{new} might manifest s_1's misinterpretation of an earlier act $a_{intended}$ by s_1. Such mistakes are possible when the surface form that s_1 observed might have accomplished either $a_{observed}$ or $a_{intended}$. (See Table 6.8.)

These defaults have a lower priority than repairs or acceptance because speakers consider misunderstandings only when no coherent interpretation is possible. The preference for coherent interpretations is especially important when there is more than one discourse-level act for which the utterance is a possible decomposition.

Table 6.7:

Name	Self-misunderstanding
Purpose	Detecting one's own misunderstanding
Axiom	DEFAULT $(3, selfMisunderstanding(s_1, s_2, p_{mistake}, a_{new}, ts)):$ $active(\mathbf{do}(s_1, a_{observed}), ts)$ $\wedge\ lintention(a_{new}, p_l)$ $\wedge\ lintention(a_{observed}, p_{l_2})$ $\wedge\ inconsistentLI(p_l, p_{l_2})$ $\wedge\ ambiguous(a_{observed}, a_{intended})$ $\wedge\ p_{mistake} = \mathbf{mistake}(s_2, a_{intended}, a_{observed})$ $\supset try(s_1, s_2, a_{new}, ts).$ FACT $selfMisunderstanding(s_1, s_2, p_{mistake}, a_1, ts)$ $\supset \neg shouldTry(s_1, s_2, a_1, ts).$ FACT $selfMisunderstanding(s_1, s_2, p_{mistake}, a_1, ts)$ $\wedge\ ambiguous(a_1, a_2)$ $\supset \neg shouldTry(s_1, s_2, a_2, ts).$
Summary	Speaker s_1 might be attempting action a_{new} in discourse ts if: 1. s_1 has performed action $a_{observed}$; 2. But, the linguistic intentions of a_{new} are inconsistent with the linguistic intentions of $a_{observed}$; 3. $a_{observed}$ and action $a_{intended}$ can be performed using a similar surface-level speech act; and 4. s_2 may have mistaken $a_{intended}$ for $a_{observed}$.

Table 6.8:

Name	Other-misunderstanding
Purpose	Detecting another's misunderstanding
Axiom	DEFAULT $(3, otherMisunderstanding(s_1, s_2, p_{mistake}, a_{new}, ts)):$ $active(\mathbf{do}(s_2, a_{intended}), ts)$ $\wedge\ ambiguous(a_{intended}, a_{similar})$ $\wedge\ wouldExpect(s_1, \mathbf{do}(s_2, a_{similar}), \mathbf{do}(s_1, a_{new}))$ $\wedge\ p_{mistake} = \mathbf{mistake}(s_1, a_{intended}, a_{similar})$ $\supset try(s_1, s_2, a_{new}, ts)$. FACT $otherMisunderstanding(s_1, s_2, p_{mistake}, a_1, ts)$ $\wedge\ ambiguous(a_1, a_2)$ $\supset \neg shouldTry(s_1, s_2, a_2, ts)$.
Summary	Speaker s_1 might be attempting action a_{new} in discourse ts if: 1. Earlier, speaker s_2 performed act $a_{intended}$; 2. Actions $a_{intended}$ and $a_{similar}$ can be performed using a similar surface form; 3. If s_2 had performed $a_{similar}$, then a_{new} would be expected; 4. s_1 may have mistaken $a_{intended}$ for $a_{similar}$.

Chapter 7

Applications of the model

In this chapter, we examine how the model accounts for conversational exchanges involving third-turn and fourth-turn repairs. We focus on three examples, the first two involving misunderstanding that leads to repair, the third involving an expectation failure that might have led to misunderstanding and repair, but instead leads to a clarification subdialogue. For each example, we identify the key assumptions made during each turn of the dialogue. We also examine in detail the system's output for the first example and the reasoning that underlies it. (Appendix E includes the system's input and output for the second and third examples.)

7.1 Fourth-turn repair: The meeting example

This example (previously Example 2.6) is from Terasaki (1976). In this example, Russ recognizes that he has misunderstood Mother's question in T1 as a pretelling and generates a fourth-turn repair.

T1	**Mother:**	Do you know who's going to that meeting?
T2	**Russ:**	Who?
T3	**Mother:**	I don't know.
T4	**Russ:**	Oh. Probably Mrs. McOwen and probably Mrs. Cadry and some of the teachers.

We represent this dialogue as the following sequence of speech acts.

T1	**m:**	**surface-request(m, r, informif(r, m, knowref(r,whoIsGoing)))**
T2	**r:**	**surface-request(r, m, informref(m, r, whoIsGoing))**
T3	**m:**	**surface-inform(m, r, not knowref(m, whoIsGoing))**
T4	**r:**	**surface-informref(r, m, whoIsGoing)**

We will consider this conversation first from Russ's perspective and then from Mother's.

7.1.1 Russ's perspective

We assume that Russ believes he knows who is going to the meeting (but believes that Mother's knowledge about the meeting would be more accurate than his own.) For simplicity, we shall represent these beliefs as facts.[1]
FACT *believe*(**r, knowref(r, whoIsGoing)**).
FACT *believe*(**r, knowsBetterRef(m,r,whoIsGoing)**).

We also assume that Russ believes that he knows whether (or not) he knows.
FACT *believe*(**r, knowif(r,knowref(r,whoIsGoing))**).
Lastly, we assume that he has *lexpectation* rules for **pretell**, **askref**, and **askif** as in Section 6.3.1 and Appendix D.[2] Figures 7.1–7.4 show the output of the system for each of the four turns of this dialogue, from Russ's perspective.

Turn 1: Russ decides that Mother is pretelling

According to the model, after Russ hears Mother's **surface-request**, *"Do you know who is going to that meeting?"*, he interprets it by attempting to construct a plausible explanation of it. This requires tentatively choosing a discourse-level act on the basis of the decomposition relation and then attempting to abduce either that it is an intentional display of understanding or that it is a symptom of misunderstanding. Theorist is called to explain the utterance and returns with a list of assumptions that were made to complete the explanation. (The portion of the output from the update describes Russ's interpretation of this explanation.)

In this simulation, T1 was explained as an intentional pretelling. The explanation contains the meta-planning assumption that Mother was pretelling as part of a plan to get Russ to ask a question. The reasoner also attributed to her the linguistic intentions of pretelling.

As a first step, decomposition information would have been used to identify the above utterance as either an **askif**, **askref**, or a **pretelling**:

decomp(**surface-request(m, r, informif(r, m, knowref(r, whoIsGoing)))**,
  pretell(m, r, whoIsGoing))

decomp(**surface-request(m, r, informif(r, m, knowref(r, whoIsGoing)))**,
  askref(m, r, whoIsGoing))

decomp(**surface-request(m, r, informif(r, m, knowref(r, whoIsGoing)))**,
  askif(m, r, whoIsGoing))

[1] As mentioned in Section 6.3.3, I only distinguish believing or not believing that a proposition is true. A more complex account, such as in Walker (1991), might use priorities to express different degrees of belief.

[2] He might also have rules for **warn** and **testref**.

```
| ?- startDialogue2.

      >>>surface-request(m,r,informif(r,m,knowref(r,whoIsGoing)))

***Interpreting Utterance***

Explaining
  utter(m,r,surface-request(m,r,informif(r,m,knowref(r,whoIsGoing))),ts(0))

Is formula
  pickForm(m,r,surface-request(m,r,informif(r,m,knowref(r,whoIsGoing))),
           pretell(m,r,whoIsGoing),ts(0)) ok (y/n)?y.

Explanation:
  intentionalAct(m,r,pretell(m,r,whoIsGoing),ts(0))
  adoptPlan(m,r,pretell(m,r,whoIsGoing),askref(r,m,whoIsGoing),ts(0))
  lintend(m,(knowref(m,whoIsGoing)
             and knowsBetterRef(m,r,whoIsGoing)
             and intend(m,do(m,informref(m,r,whoIsGoing)))
             and intend(m,knowref(r,whoIsGoing))),ts(0))
  credulousBA(m,knowsBetterRef(m,r,whoIsGoing))
  credulousI(m,ts(0))
  pickForm(m,r,surface-request(m,r,informif(r,m,knowref(r,whoIsGoing))),
           pretell(m,r,whoIsGoing),ts(0))

***Updating Discourse Model***

Interpretation: pretell(m, r, whoIsGoing) (turn number 1)
  expressed(do(m, pretell(m, r, whoIsGoing)), 1)

Linguistic Intentions of pretell(m,r,whoIsGoing):
  knowref(m,whoIsGoing)
  knowsBetterRef(m,r,whoIsGoing)
  intend(m,do(m,informref(m,r,whoIsGoing)))
  intend(m,knowref(r,whoIsGoing))

Suppositions Added:
  expressed(knowref(m, whoIsGoing), 1)
  expressed(knowsBetterRef(m, r, whoIsGoing), 1)
  expressed(intend(m, do(m, informref(m, r, whoIsGoing))), 1)
  expressed(intend(m, knowref(r, whoIsGoing)), 1)

Agent m adopted plan to achieve:  askref(r,m,whoIsGoing)
```

Figure 7.1: The output for turn 1 from Russ's perspective

In this case, the possibility that Mother is attempting a pretelling was considered. (The system uses an oracle, represented by the default *pickForm* to simulate this choice.[3]) It is important to note that this is just one of the possible explanations available to Russ. Nothing in his beliefs rules out abducing explanations from either the **askif** or the **askref** interpretation.

After this initial choice, the reasoner must explain

try(**m, r, pretell(m, r, whoIsGoing), ts(0)**)

as either an intentional action (by explaining *shouldTry*(**m, r, pretell(m, r, whoIsGoing), ts(0)**)), as self-misunderstanding, or as other-misunderstanding. In this case, an intentional explanation was found to be sufficient because T1 could be explained as plan adoption. Russ could explain T1 as plan adoption because there is a plausible goal act that a pretelling would help Mother to achieve and it is consistent to assume that achieving this act was, in fact, her goal.[4] In particular, **askref** is a plausible goal for Mother from Russ's perspective because:

- Russ has a *lexpectation* rule that says that a pretelling normally creates an expectation for the listener to ask the speaker to tell:

  FACT *lexpectation*(**do(m, pretell(m, r, whoIsGoing)),**
  knowsBetterRef(m, r, whoIsGoing),
  do(r, askref(r, m, whoIsGoing))).

  By his credulousness about Mother's beliefs (*credulousBA*), Russ may assume that Mother believes **knowsBetterRef(m, r, whoIsGoing)**. (This would not be possible if he previously believed that she disbelieved this condition.) Thus, Russ could infer
  wouldExpect(r, pretell(m, r, whoIsGoing), askref(r, m, whoIsGoing))

- By his credulousness about Mother's goals (*credulousI*), Russ may assume that Mother wants him to perform the expected **askref**.

Lastly, the linguistic intentions of pretelling are consistent with those that have been expressed, so it is consistent to assume that Mother is intending to use it as part of a plan.

Turn 2: Russ decides to respond with an askref

In this turn, Russ produces a **surface-request**. This utterance is appropriate, independent of whether or not Russ wants to know the information, because it displays acceptance of Mother's pretelling. It displays acceptance, because a **surface-request** is one way to perform an **askref**, an act that is expected according to Russ's

[3]This oracle permits the user to rule out certain choices to test different interpretations. However, once a default is declared inconsistent it cannot be reconsidered. Thus, although misunderstandings are less preferable than a coherent interpretations based on an alternative decomposition, the system can only consider alternatives that have not been ruled out by the user.

[4]Because Russ does not have any expectations with regard to what Mother is doing, her utterance could not be explained as acceptance or challenge.

```
Explaining shouldTry(r,m,A,ts(1)),intentionalAct(r,m,A,ts(1)),decomp(A2,A)

Answer: shouldTry(r,m,askref(r,m,whoIsGoing),ts(1)),
        intentionalAct(r,m,askref(r,m,whoIsGoing),ts(1)),
        decomp(surface-request(r,m,informref(m,r,whoIsGoing)),
               askref(r,m,whoIsGoing))

Explanation:
  intentionalAct(r,m,askref(r,m,whoIsGoing),ts(1))
  acceptance(r,askref(r,m,whoIsGoing),ts(1))
  lintend(r,(not knowref(r,whoIsGoing)
             and intend(r,knowref(r,whoIsGoing))
             and intend(r,do(m,informref(m,r,whoIsGoing)))),ts(1))
  expectedReply(do(m,pretell(m,r,whoIsGoing)),
                knowsBetterRef(m,r,whoIsGoing),
                do(r,askref(r,m,whoIsGoing)),ts(1))

***Updating Discourse Model***

Interpretation: askref(r,m,whoIsGoing) (turn number 2)
  expressed(do(r,askref(r,m,whoIsGoing)),2)

Linguistic Intentions of askref(r,m,whoIsGoing):
  not knowref(r,whoIsGoing)
  and intend(r,knowref(r,whoIsGoing))
  and intend(r,do(m,informref(m,r,whoIsGoing)))

Suppositions Added:
  expressedNot(knowref(r,whoIsGoing),2)
  expressed(intend(r,knowref(r,whoIsGoing)),2)
  expressed(intend(r,do(m,informref(m,r,whoIsGoing))),2)

Agent r performed expected act: askref(r,m,whoIsGoing)

***Generating Utterance***

<<<surface-request(r, m, informref(m, r, whoIsGoing))
```

Figure 7.2: The output for turn 2 from Russ's perspective

model of the discourse after the first turn.[5] Theorist finds (by default deduction of $(\exists u)\ shouldTry(\mathbf{r}, \mathbf{m}, u, \mathbf{ts(1)}))$ that the discourse-level act **askref** would be coherent in the current context:

- According to the discourse model, it is true that *active*(**do(m pretell(m, r, whoIsGoing)), ts(1)**).
- There is an *lexpectation* rule corresponding to the adjacency pair *pretell–askref.*
- Russ believes the conditions of this rule: **knowsBetterRef(m, r, whoIsGoing)**.
- The linguistic intentions of **askref** are compatible with those already expressed.

Turn 3: Russ decides his interpretation of T1 was wrong

Mother replies with a **surface-inform**. This is interpreted as a discourse-level **inform-not-knowref**. This act signals a misunderstanding, because the linguistic intentions associated with it are incompatible with those previously assumed. For T3 to have demonstrated acceptance, the linguistic intentions of **inform(m, r, not(knowref(m, whoIsGoing)))** would need to have been compatible with this interpretation of the discourse. However, **not(knowref(m, whoIsGoing))** is among these intentions, while *active*(**knowref(m, whoIsGoing),ts(2)**). As a result, T3 cannot be attributed to any expected act, and must be attributed to a misunderstanding by Russ or Mother.

Figure 7.3 shows that Theorist attributed T3 to a misunderstanding by Russ, in particular, to his incorrectly interpreting one of Mother's utterances as a **pretelling**, rather than as an **askref.** This explanation succeeded because each of the conditions of the rule for self-misunderstanding were explainable. We sketch the proof below.

1. According to the context, *expressed*(**do(m, pretell(m, r, whoIsGoing)), 0**). And, it is consistent to assume that the activation of this supposition persists:
   activationPersists(**do(m, pretell(m, r, whoIsGoing)), 2**)

   Thus, it holds that *active*(**do(m, pretell(m, r, whoIsGoing)), ts(2)**).

2. The acts **pretell** and **askref** both have a surface form:
   surface-request(m, r, informif(r, m, knowref(r, whoIsGoing)))
   So, *ambiguous*(**pretell(m, r, whoIsGoing), askref(m, r, whoIsGoing)**).

3. The linguistic intentions of the pretelling are:

[5] If, for some reason, Russ did *not* want to know the information, he might decide not to produce an **askref**. However, he would then be accountable for justifying his action as well as for displaying his acceptance of Mother's displayed understanding (*e.g.,* by including an explicit rejection of her offer); otherwise she might think that one of them has misunderstood.

```
    >>>surface-inform(m, r, not knowref(m, whoIsGoing))

***Interpreting Utterance***

Explaining utter(m,r,inform(m,r,not knowref(m,whoIsGoing)),ts(2))

Is formula
  pickForm(m,r,surface-inform(m,r,not knowref(m,whoIsGoing)),
           inform(m,r,not knowref(m,whoIsGoing)),ts(2)) ok (y/n)?y.

Explanation:
  selfMisunderstanding(m,r,mistake(r,askref(m,r,whoIsGoing),
                                     pretell(m,r,whoIsGoing)),
                       inform(m,r,not knowref(m,whoIsGoing)),ts(2))
  activationPersists(do(m,pretell(m,r,whoIsGoing)),2)
  lintend(m,(not knowref(m,whoIsGoing)
             and intend(m,knowif(r,not knowref(m,whoIsGoing)))),ts(2))
  pickForm(m,r,surface-inform(m,r,not knowref(m,whoIsGoing)),
           inform(m,r,not knowref(m,whoIsGoing)),ts(2))

***Updating Discourse Model***

Interpretation:
  inform(m, r, not knowref(m, whoIsGoing)) (turn number 3)
  expressed(do(m, inform(m, r, not knowref(m, whoIsGoing))), 3)

Linguistic Intentions of inform(m,r,not knowref(m,whoIsGoing)):
  not knowref(m,whoIsGoing)
  intend(m,knowif(r,not knowref(m,whoIsGoing)))

Suppositions Added:
  expressed(mistake(r, askref(m, r, whoIsGoing),
                       pretell(m, r, whoIsGoing)),3)
  expressedNot(knowref(m, whoIsGoing), 3)
  expressed(intend(m, knowif(r, not knowref(m, whoIsGoing))), 3)

Agent r misunderstood act do(m, askref(m, r, whoIsGoing))
as do(m, pretell(m, r, whoIsGoing))
```

Figure 7.3: The output for turn 3 from Russ's perspective

and(knowref(m, whoIsGoing),
and(knowsBetterRef(m, r, whoIsGoing),
and(intend(m, do(m, informref(m, r, whoIsGoing))),
intend(m, knowref(r, whoIsGoing)))))

The linguistic intentions of inform-not-knowref are:
and(not(knowref(m, whoIsGoing)),
intend(m, knowif(r, not(knowref(m, whoIsGoing))))).

But these intentions are inconsistent, because the simple conjuncts **knowref(m, whoIsGoing)** and **not(knowref(m, whoIsGoing))** are incompatible. As a result, *inconsistentLI* holds for these linguistic intentions.

4. Because there is no coherent interpretation, it is consistent to assume that a misunderstanding occurred:

   selfMisunderstanding(m,r,
   mistake(r,askref(m, r, whoIsGoing),
   pretell(m, r, whoIsGoing)),
   inform(m, r, not(knowref(m, whoIsGoing))),
   ts(2)).

5. By explanations 1–4 and the rule for self-misunderstanding, *try*(**m, r, inform(m, r,not(knowref(m, whoIsGoing))),ts(2)**) is explained.

6. The rest of the explanation follows from 5, the *decomp* relation, and the utterance explanation rule.

As a result of this interpretation, the supposition **not(knowref(m, whoIsGoing))** is added to the discourse model as the fact *expressedNot*(**not(knowref(m, whoIsGoing))**). This addition terminates the activation of **knowref(m, whoIsGoing)** that had persisted since the first turn. (At the same time, if Russ had revised his beliefs on the basis of the first turn, probably he would now revise them again; however, the present theory does not account for any changes.)

Turn 4: Russ produces an informref

Russ performs a fourth-turn repair by producing a **surface-informref**. When Theorist is called to find a coherent discourse-level act, it finds **informref** to be a solution. This act would be coherent because, according to the updated context, Russ had misunderstood a prior utterance by Mother and the expectations invoked by this utterance have yet to be addressed.

An **informref** by Russ is expected in the reconstructed dialogue because:

- There is an *lexpectation* rule corresponding to the adjacency pair *askref–informref*;
- Russ believes the conditions of this rule; and
- The linguistic intentions of **informref** are compatible with the reconstruction.

```
Explaining shouldTry(r,m,A,ts(3)),intentionalAct(r,m,A,ts(3)),decomp(As,A)

***Reconstructing Turn Number 1***

Suppositions Added:
  expressed(do(m, askref(m, r, whoIsGoing)), alt(1))
  expressedNot(knowref(m, whoIsGoing), alt(1))
  expressed(intend(m, knowref(m, whoIsGoing)), alt(1))
  expressed(intend(m, do(r, informref(r, m, whoIsGoing))), alt(1))

**********************************

Answer: shouldTry(r,m,informref(r,m,whoIsGoing),ts(3)),
        intentionalAct(r,m,informref(r,m,whoIsGoing),ts(3)),
        decomp(surface-informref(r,m,whoIsGoing),
               informref(r,m,whoIsGoing)

Explanation:
  intentionalAct(r,m,informref(r,m,whoIsGoing),ts(3))
  makeFourthTurnRepair(r,m,informref(r,m,whoIsGoing),ts(3),ts(1))
  lintend(r,(knowref(r,whoIsGoing
             and intend(r,knowref(m,whoIsGoing))),ts(1))
  reconstructTS(1, do(m,pretell(m,r,whoIsGoing)),
                do(m,askref(m,r,whoIsGoing)), ts(3),ts(alt(1)))

***Updating Discourse Model***

Interpretation: informref(r,m,whoIsGoing) (turn number 4)
  expressed(do(r,informref(r,m,whoIsGoing)),4)

Linguistic Intentions of informref(r,m,whoIsGoing):
  knowref(r,whoIsGoing)and intend(r,knowref(m,whoIsGoing))

Suppositions Added:
  expressed(knowref(r,whoIsGoing),4)
  expressed(intend(r,knowref(m,whoIsGoing)),4)

r performed fourth turn repair

***Generating Utterance***

<<<surface-informref(r,m,whoIsGoing)
```

Figure 7.4: The output for turn 4 from Russ's perspective

7.1.2 Mother's perspective

We now consider the 'meeting' example from Mother's perspective. First, we assume that Mother wants Russ to tell her if he knows, and that she believes that she does not already know and that he knows whether he knows.
FACT *hasGoal*(**m, do(r, informif(r, m, knowref(r, whoIsGoing))),ts(0)).**
FACT *believe*(**m, not(knowref(m, whoIsGoing))).**
FACT *believe*(**m, knowif(r, knowref(r, whoIsGoing))).**
We also assume that she has *lexpectation* rules for **pretell**, **askref**, and **askif**. Figures 7.5–7.8 show the output of the system for each of the four turns of this dialogue from Mother's perspective.

Turn 1: Mother performs an askif

As stated earlier, Mother begins the dialogue by performing a **surface-request**.[6] Theorist suggests performing an **askif** as part of a plan to get Russ to produce an **informif**. Plan adoption is useful here because the conditions of this meta-plan are explainable:

1. According to our initial assumptions, Mother wants Russ to perform an **informif-knowref**.

2. According to the *lexpectation* relation, this act would be expected to follow an **askif**, because Mother believes the conditions **knowif(r, knowref(r, whoIsGoing))**.

3. The linguistic intentions of **askif** are (trivially) compatible with the discourse.

4. The predicate *shouldTry*(**m, r, askif(m, r, knowif(r, whoIsGoing)), ts(0)**) is explained by 1–3 and the meta-plan for plan adoption.

In fact, plan adoption would have to have been part of any solution, because we assumed that no utterances or misunderstandings were active in the context; thus, any attempt to infer a solution as acceptance, challenge, or repair would fail.

Turn 2: Mother decides that Russ is doing an askref

In this turn, Russ produces an **surface-request** that Theorist attributes to an intentional **askref**. The explanation includes the meta-planning assumption that he was doing so as part of a plan to get her to produce an **informref**. (T2 might also have been explained by abducing that Russ incorrectly believes that she knows who is going and misunderstood T1 as an attempted pretelling.)[7] Plan adoption provides a plausible explanation here because:

[6] Real dialogues normally begin with an opening exchange to establish the topic of conversation. The 'meeting' example is actually a fragment of a longer conversation.

[7] If such a misunderstanding had been detected, it might have led her to produce a third-turn repair ("No, I'm asking you") in the next turn.

```
| ?- startDialogue1.

Explaining shouldTry(m,r,A,ts(0)),intentionalAct(m,r,A,ts(0)),decomp(As,A)

Answer: shouldTry(m,r,askif(m,r,knowref(r,whoIsGoing)),ts(0)),
        intentionalAct(m,r,askif(m,r,knowref(r,whoIsGoing)),ts(0)),
        decomp(surface-request(m,r,informif(r,m,knowref(r,whoIsGoing))),
               askif(m,r,knowref(r,whoIsGoing)))

Explanation:
  intentionalAct(m,r,askif(m,r,knowref(r,whoIsGoing)),ts(0))
  adoptPlan(m,r,
            askif(m,r,knowref(r,whoIsGoing)),
            informif(r,m,knowref(r,whoIsGoing)),ts(0))
  lintend(m,(not knowif(m,knowref(r,whoIsGoing))
             and intend(m,knowif(m,knowref(r,whoIsGoing)))
             and intend(m,do(r,informif(r,m,knowref(r,whoIsGoing))))),
          ts(0))

***Updating Discourse Model***

Interpretation: askif(m,r,knowref(r,whoIsGoing)) (turn number 1)
  expressed(do(m,askif(m,r,knowref(r,whoIsGoing))),1)

Linguistic Intentions of askif(m,r,knowref(r,whoIsGoing)):
  not knowif(m,knowref(r,whoIsGoing))
  and intend(m,knowif(m,knowref(r,whoIsGoing)))
  and intend(m,do(r,informif(r,m,knowref(r,whoIsGoing))))

Suppositions Added:
  expressedNot(knowif(m,knowref(r,whoIsGoing)),1)
  expressed(intend(m,knowif(m,knowref(r,whoIsGoing))),1)
  expressed(intend(m,do(r,informif(r,m,knowref(r,whoIsGoing)))),1)

Agent m adopted plan to achieve: informif(r,m,knowref(r,whoIsGoing))

***Generating Utterance***

<<<request(m,r,informif(r,m,knowref(r,whoIsGoing)))
```

Figure 7.5: The output for turn 1 from Mother's perspective

```
    >>>surface-request(r,m,informref(m,r,whoIsGoing))

***Interpreting Utterance***

Explaining utter(r,m,request(r,m,informref(m,r,whoIsGoing)),ts(1))

Is formula pickForm(r,m,surface-request(r,m,informref(m,r,whoIsGoing)),
                    askref(r,m,whoIsGoing),ts(1)) ok (y/n)?y.

Explanation:
  intentionalAct(r,m,askref(r,m,whoIsGoing),ts(1))
  adoptPlan(r,m,askref(r,m,whoIsGoing),informref(m,r,whoIsGoing),ts(1))
  lintend(r,(not knowref(r,whoIsGoing)
             and intend(r,knowref(r,whoIsGoing))
             and intend(r,do(m,informref(m,r,whoIsGoing)))),ts(1))
  credulousBA(r,knowref(m,whoIsGoing))
  credulousI(r,ts(1))
  pickForm(r,m,surface-request(r,m,informref(m,r,whoIsGoing)),
           askref(r,m,whoIsGoing),ts(1))

***Updating Discourse Model***

Interpretation: askref(r,m,whoIsGoing) (turn number 2)
  expressed(do(r,askref(r,m,whoIsGoing)),2)

Linguistic Intentions of askref(r,m,whoIsGoing):
  not knowref(r,whoIsGoing)
  and intend(r,knowref(r,whoIsGoing))
  and intend(r,do(m,informref(m,r,whoIsGoing)))

Suppositions Added:
  expressedNot(knowref(r,whoIsGoing),2)
  expressed(intend(r,knowref(r,whoIsGoing)),2)
  expressed(intend(r,do(m,informref(m,r,whoIsGoing))),2)

Agent r adopted plan to achieve:  informref(m,r,whoIsGoing)
```

Figure 7.6: The output for turn 2 from Mother's perspective

1. Mother has a *lexpectation* rule that says that an askref normally creates an expectation for the listener to tell the speaker the answer:

   FACT *lexpectation*(**do(r, askref(r, m, whoIsGoing)),**
   knowref(m, whoIsGoing),
   do(m, informref(m, r, whoIsGoing))).

   This, along with her credulousness about Russ's beliefs (*credulousBA*), explains **wouldExpect(r, askref(r, m, whoIsGoing), informref(m, r, whoIsGoing))**

2. Mother's credulousness about Russ's goals (*credulousI*), explains her belief that he wants her to perform the expected **informref.**

3. The linguistic intentions of **askref** are consistent with those that have been expressed, so it is consistent to assume that Russ is intending to use it as part of a plan. (They are consistent with the context because T1 expresses only that Mother does not know *whether* Russ knows and not that she does not herself know.)

4. Thus, by 1–3 and the meta-plan for plan adoption, *shouldTry*(**r, m, askref(r, m, whoIsGoing), ts(0)**) is explainable.

Once this meta-planning explanation is found, *try*(**r, m, askref(r, m, whoIsGoing), ts(0)**) is explainable by assuming the default *intentionalAct*. The rest follows from the *decomp* relation and the utterance rule.

Turn 3: Mother decides to produce an inform

In this turn, Mother produces a **surface-inform**. Theorist suggests that an **inform** would display acceptance of Russ's **askref**, because it is expected according to her representation of the context:[8]

1. From her interpretation of T2, it holds that
   active(**do(r askref(r, m, whoIsGoing)), ts(2)**).

2. There is an *lexpectation* rule corresponding to the adjacency pair
   askref–inform-not-knowref.

3. Mother believes the conditions of this rule
   (*i.e., believe*(**m, not(knowref(m, whoIsGoing))**)).

4. The linguistic intentions of the **inform** are compatible with those already expressed.

It then follows that:
shouldTry(**m, r, inform(m, r, not knowref(m, whoIsGoing)), ts(2)**).
And, by default (*intentionalAct*):
try(**m, r, inform(m, r, not knowref(m, whoIsGoing)), ts(2)**).

[8]This expectation depends on an assumption to treat *askref–inform-not-knowref* as a valid adjacency pair. Without this assumption, Mother's reply in T3 would still be recognized as coherent, but as a *challenge* rather than an *acceptance*.

```
Explaining shouldTry(m,r,A,ts(2)),intentionalAct(m,r,A,ts(2)),decomp(As,A)

Answer: shouldTry(m,r,inform(m,r,not knowref(m,whoIsGoing)),ts(2)),
        intentionalAct(m,r,inform(m,r,not knowref(m,whoIsGoing)),ts(2)),
        decomp(surface-inform(m,r,not knowref(m,whoIsGoing)),
               inform(m,r,not knowref(m,whoIsGoing)))

Explanation:
  intentionalAct(m,r,inform(m,r,not knowref(m,whoIsGoing)),ts(2))
  acceptance(m,inform(m,r,not knowref(m,whoIsGoing)),ts(2))
  lintend(m,(not knowref(m,whoIsGoing)
             and intend(m,knowif(r,not knowref(m,whoIsGoing)))),ts(2))
  expectedReply(do(r,askref(r,m,whoIsGoing)),
                not knowref(m,whoIsGoing),
                do(m,inform(m,r,not knowref(m,whoIsGoing))),ts(2))

***Updating Discourse Model***

Interpretation: inform(m,r,not knowref(m,whoIsGoing)) (turn number 3)
  expressed(do(m,inform(m,r,not knowref(m,whoIsGoing))),3)

Linguistic Intentions of inform(m,r,not knowref(m,whoIsGoing)):
  not knowref(m,whoIsGoing)
  and intend(m,knowif(r,not knowref(m,whoIsGoing)))

Suppositions Added:
  expressedNot(knowref(m,whoIsGoing),3)
  expressed(intend(m,knowif(r,not knowref(m,whoIsGoing))),3)

Agent m performed expected act: inform(m,r,not knowref(m,whoIsGoing))

***Generating Utterance***

<<<inform(m,r,not knowref(m,whoIsGoing))
```

Figure 7.7: The output for turn 3 from Mother's perspective

```
      >>>surface-informref(r,m,whoIsGoing)

***Interpreting Utterance***

Explaining utter(r,m,informref(r,m,whoIsGoing),ts(3))

Is formula pickForm(r,m,surface-informref(r,m,whoIsGoing),
                    informref(r,m,whoIsGoing),ts(3)) ok (y/n)?n.

Is formula pickForm(r,m,surface-informref(r,m,whoIsGoing),
                    informif(r,m,knowref(r,whoIsGoing)),ts(3)) ok (y/n)?y.

Explanation:
  activationPersists(do(m,askif(m,r,knowref(r,whoIsGoing))),3)
  activationPersists(do(m,askif(m,r,knowref(r,whoIsGoing))),2)
  intentionalAct(r,m,informif(r,m,knowref(r,whoIsGoing)),ts(3))
  acceptance(r,informif(r,m,knowref(r,whoIsGoing)),ts(3))
  lintend(r,(knowif(r,knowref(r,whoIsGoing))
             and intend(r,knowif(m,knowref(r,whoIsGoing)))),ts(3))
  expectedReply(do(m,askif(m,r,knowref(r,whoIsGoing))),
                knowif(r,knowref(r,whoIsGoing)),
                do(r,informif(r,m,knowref(r,whoIsGoing))),ts(3))
  credulousBA(r,knowif(r,knowref(r,whoIsGoing)))
  pickForm(r,m,surface-informref(r,m,whoIsGoing),
           informif(r,m,knowref(r,whoIsGoing)),ts(3))

***Updating Discourse Model***

Interpretation: informif(r,m,knowref(r,whoIsGoing)) (turn number 4)
  expressed(do(r,informif(r,m,knowref(r,whoIsGoing))),4)

Linguistic Intentions of informif(r,m,knowref(r,whoIsGoing)):
  knowif(r,knowref(r,whoIsGoing))
  and intend(r,knowif(m,knowref(r,whoIsGoing)))

Suppositions Added:
  expressed(knowif(r,knowref(r,whoIsGoing)),4)
  expressed(intend(r,knowif(m,knowref(r,whoIsGoing))),4)

Agent r performed expected act:  informif(r,m,knowref(r,whoIsGoing))
```

Figure 7.8: The output for turn 4 from Mother's perspective

Turn 4: Mother's question is accepted

In this turn, Russ produces an **surface-informref**. This utterance is attributed to an intentional **informif** by making the meta-planning assumption that Russ is displaying acceptance of her earlier **askif**. (The explanation is similar to T3, but requires additional assumptions about the persistence of her interpretation of T1.)

The interpretation of this turn was simplified by the fact that I directed the system to consider that the **surface-informref** was an **informif**, making it fit Mother's unsatisfied expectation. If I had allowed Theorist to pursue the **informref** interpretation, it would have treated Russ's utterance as a signal that Mother had misunderstood T2 (because T2 expressed **not(knowref(r, whoIsGoing))** and an **informref** expresses **knowref(r, whoIsGoing)**).[9][10] In particular, it would have considered it an indication that she had mistaken a **testref** by Russ as an **askref**. Since coherent interpretations are preferred to misunderstandings when there is an alternative, the system would have first attempted to verify that it did not violate priority constraints by attempting to find a higher priority, coherent interpretation. Although it would have found **informif** to be a candidate (because it is expected), unless Mother has a belief:
believe(**r, knowif(r, knowref(r, whoIsGoing))**),
this interpretation would have been blocked by her own credulousness (which would allow her to assume Russ had the opposite belief).[11] If the coherent interpretation were not blocked, the system would backtrack to the top level, ask the user whether it would be okay to consider a different discourse act, and then attempt to explain it.

7.1.3 Summary

From Russ's perspective, the utterances had the following discourse-level interpretations at the time each was produced:

T1	**m:**	**pretell(m, r, whoIsGoing)**
T2	**r:**	**askref(r, m, whoIsGoing)**
T3	**m:**	**inform(m, r, not knowref(m, whoIsGoing))**
T4	**r:**	**informref(r, m, whoIsGoing)**

The inconsistency of T1 and T3 leads Russ to decide that his interpretation T1 as a **pretelling** is incorrect. T4 is a repair of this mistake that is a response to his reinterpretation of T1 as an **askref**.

[9] A more expressive language for suppositions would be needed to capture the differences in (un)certainty expressed in T2 and T4.

[10] A related limitation of the model is that currently it does not provide mechanisms for a speaker to retract a previous utterance, except during the repair of a misunderstanding. However, extending the model would be straightforward.

[11] One could argue that the notion of priority constraints does not match our intuitions for this example, because with equal priority it could be assumed that Russ did have the requisite belief and hence the coherent interpretation should not be overruled.

From Mother's perspective, the utterances in this example had the following discourse-level interpretations at the time each was produced:

T1	**m:**	**askif(m, r, knowref(r,whoIsGoing))**
T2	**r:**	**askref(r, m, whoIsGoing)**
T3	**m:**	**inform(m, r, not(knowref(m, whoIsGoing)))**
T4	**r:**	**informif(r, m, knowref(r,whoIsGoing))**

According to this interpretation, T1 is an attempt to find out whether Russ knows who is going.Although from the perspective of an analyst, other readings of this turn are possible, this one is supported by Mother's treatment of T2 as a legitimate request for an answer, rather a mistake warranting explicit repair. In T3, Mother produces a rejection of Russ's request, an expectable response in this situation. As a result, T2 and T3 form a side-sequence and T4 resumes the top-level sequence initiated by T1. T4 is interpreted as an **informif**, because it is the expected reply to her **askif** in T1.

7.2 Third-turn repair: The 'dinner for Alfred' example

The next example, from McLaughlin (1984, page 219), involves two agents, A and B, discussing the time of a dinner:

Example 7.1 'Dinner for Alfred'

T1	**A:**	When is the dinner for Alfred?
T2	**B:**	Is it at seven-thirty?
T3	**A:**	No, *I'm* asking *you.*
T4	**B:**	Oh. I don't know.

In this example, there is a misunderstanding about the discourse role of T1. Agent A produces a question in T1 apparently intending to express a desire for Agent B to tell A the time of the dinner, but B interprets T1 as a *test.* That is, B interprets T1 as an attempt by A to verify that he knows when the dinner is (*e.g.,* to make sure that he will not be late). In T2, B produces one of the possible responses to a test: a guess about the answer, phrased as a yes-no question. As T2 is inconsistent with A's interpretation of T1, A produces a third-turn repair at T3, telling B to interpret T1 as a request. In T4, B responds to this *re*interpretation of T1 by rejecting the request.

This conversation has the following representation in the utterance-level language:

T1	**a:**	**surface-request(a, b, informref(b, a, whenIsDinner))**
T2	**b:**	**surface-request(b, a, informif(a, b, dinnerAtSevenThirty)**
T3	**a:**	**surface-inform(a, b, intend(a, do(a, askref(a, b, whenIsDinner))))**
T4	**b:**	**surface-inform(b, a, not(knowref(b, whenIsDinner)))**

We shall now consider the key assumptions that each participant makes in the derivation of this conversation. The basic mechanisms for interpretation and repair are identical to the previous example, so we shall not repeat those details here. Instead, we shall focus on the meta-planning and diagnostic assumptions that each agent makes, allowing us to compare the two views side-by-side.

We will assume that each agent had the following initial beliefs and goals:

A's perspective

DEFAULT $(1, credulousBA(P))$:
believe(**b**, P).

FACT *believeA*(**a, not(knowref(a, whenIsDinner))**).

FACT *believe*(**a, knowref(b, whenIsDinner)**).

FACT *believe*(**b, not(knowif(a, dinnerAtSevenThirty))**).

FACT *hasGoal*(**a, do(b, informref(b, a, whenIsDinner, ts0)**).

B's perspective

DEFAULT $(1, credulousBA(P))$:
believe(**a**, P)).

FACT *believeA*(**b, knowref(a, whenIsDinner)**).

FACT *believe*(**b, not(knowref(b, whenIsDinner))**).

FACT *believe*(**b, suspectThat(b, dinnerAtSevenThirty)**).

The key assumptions that each agent makes to explain each turn are the following:

A's perspective

T1 generation
lintend(**a,and(
not(knowref(a,whenIsDinner)),and(
intend(a,knowref(a,whenIsDinner)),
intend(a, do(b, informref(b, a,
whenIsDinner))))),
ts0)**

adoptPlan(**a, b,
askref(a, b, whenIsDinner),
informref(b, a, whenIsDinner), ts0)**

B's perspective

T1 interpretation
lintend(**a,and(
knowref(a, whenIsDinner),
intend(a, do(b, informref(b, a,
whenIsDinner)))),
ts0)**

adoptPlan(**a, b,
testref(a, b, whenIsDinner),
askif(b, a, dinnerAtSevenThirty),
ts0)**

T2 interpretation
lintend(**b,and(**
not(knowif(b,dinnerAtSevenThirty)),
and(
intend(b, knowif(b, dinnerAtSeven-Thirty)),
intend(b, do(a,informif(a,b,
dinnerAtSevenThirty))))), ts1)

otherMisunderstanding(**b,a,mistake(b,**
askref(a, b, whenIsDinner),
testref(a, b, whenIsDinner)),
askif(b,a,dinnerAtSevenThirty),
ts1)

T3 generation
lintend(**a, and(**
intend(a, do(a,askref(a,b,
whenIsDinner))),
intend(a, knowif(b, intend(a,
do(a,askref(a,b,whenIsDinner)))))),
ts2)

makeThirdTurnRepair(**a, b,**
inform(a, b, intend(a,
do(a, askref(a, b, whenIsDinner)))),
ts2)

T4 interpretation
lintend(**b,**
and(not(knowref(b,whenIsDinner)),
intend(b,knowif(a,
not(knowref(b,whenIsDinner))))),ts3)

acceptance(**b,**
inform(b, a, not(knowref(b,
whenIsDinner))), ts3)

T2 generation
lintend(**b, and(**
not(knowif(b,dinnerAtSevenThirty)),
and(
intend(b, knowif(b, dinnerAtSeven-Thirty)),
intend(b, do(a,informif(a,b,
dinnerAtSevenThirty))))), ts1)

acceptance(**b,**
askif(b,a,dinnerAtSevenThirty),
ts1)

T3 interpretation
lintend(**a,**
and(intend(a,do(a,askref(a,b,
whenIsDinner))),
intend(a,knowif(b,intend(a,
do(a,askref(a,b,
whenIsDinner)))))), ts2)

selfMisunderstanding(**a, b,**
mistake(b,
askref(a, b, whenIsDinner),
testref(a, b, whenIsDinner)),
inform(a, b, intend(a,
do(a, askref(a, b, whenIsDinner)))),
ts2)

T4 generation
lintend(**b,**
and(not(knowref(b,whenIsDinner)),
intend(b,knowif(a,
not(knowref(b,whenIsDinner))))),
ts3)

makeFourthTurnRepair(**b, a,**
inform(b, a, not(knowref(b,
whenIsDinner))), ts3,alts1)

These utterances had the following discourse-level interpretations by each agent at the time each utterance was produced:

A's perspective	**B's perspective**
T1 askref(a, b, whenIsDinner)	**T1 testref(a, b, whenIsDinner)**
T2 askif(b, a, dinnerAtSevenThirty	**T2 askif(b, a, dinnerAtSevenThirty**
T3 inform(a, b, intend(a, do(a, askref(a, b, whenIsDinner))))	**T3 inform(a, b, intend(a, do(a, askref(a, b, whenIsDinner))))**

T4 inform(b, a, not(knowref(b, whenIsDinner)))	**T4 inform(b, a, not(knowref(b, whenIsDinner)))**

Thus, the model accounts for A's ability to recognize B's misunderstanding and generate a third-turn repair. From A's perspective, B's utterance in T2 displays misunderstanding because it would not be consistent for B to express that B intends A to produce an **informif** given that A has supposed that A does not know the time of the dinner. From B's perspective, A's utterance in T3 makes him aware of this inconsistency, revealing to B his own misunderstanding of T1. As a result, B performs a *fourth-turn* repair in T4. From B's perspective there is no third-turn repair, because prior to T3 B had not supposed himself to have misunderstood T1. Finally, A interprets T4 as an acceptance of T1.

7.3 Expectation failure without repair: The 'Susan wants nails' example

This example, from Brennan (1990, page 1), involves two office mates, Susan and Bridget, trying to ascertain whether Bridget has some nails suitable for hanging pictures:

Example 7.2 'Susan wants nails'

T1	**Susan:**	You don't have any *nails*, do you?
T2	**Bridget:**	Fingernails?
T3	**Susan:**	No, nails to nail into the wall. (*pause*) When I get bored here I'm going to go put up those pictures.
T4	**Bridget:**	No.

In this example, Susan produces T1 intending and expecting that Bridget will tell her if she has some nails; however, Bridget violates this expectation by instead producing a question of her own in T2. Although Susan did not expect this response, unlike the previous examples this does not trigger a repair, because Susan is able to produce an answer in T3 that is consistent with Bridget's model of the discourse. Thus, for both agents, T2 and T3 form a side-sequence (*i.e.,* a clarification subdialogue). T4 then resumes the top-level sequence serving as an answer to the question posed in T1.

This example, in addition to demonstrating that expectation failure need not trigger a repair, highlights some important limitations of the current model. Here, we will consider only one of them in detail, briefly mention the others, but postpone a full discussion until after the example. The first limitation is that the type of understanding failure in this example involves a problem of reference, rather than a problem of relevance as in the previous examples, but the model includes no special representation of attempts to refer. To handle the example, we introduce a new supposition, **typeWanted**, where **typeWanted**(d_1, d_2) expresses that the referent of description d_2 is the desired referent of d_1 (which subsumes d_2); a less *ad hoc* approach would be to represent reference as an action in its own right (see Heeman and Hirst 1992). Another problem with the model is that it lacks mechanisms for updating the actual beliefs of an agent on the basis of the discourse model; hence we will have to attribute some prior beliefs to the agents that they would not normally form until later in the dialogue. We will also permit any agent to assume that any activated supposition is believed. Lastly, the model presumes a one-to-one correspondence between turns and speech acts; so even though it would be more reasonable to represent T3 as two separate acts, we represent only one.

Given the limitations mentioned above, this exchange has the following representation in the utterance-level language:

T1	s:	**surface-request(s,b,informif(b,s,have(b,nails)))**
T2	b:	**surface-request(b,s,informif(s,b,typeWanted(nails,fingerNails)))**
T3	s:	**surface-inform(s,b,not typeWanted(nails,fingerNails))**
T4	b:	**surface-informif(b,s,have(b,nails))**

We will assume that each agent had the following initial beliefs and goals:

Susan's perspective

DEFAULT $(1, credulousBA(P))$:
$believe(\mathbf{b}, P)$).

FACT *believeA*(**s, knowif(b,**
have(b, nails))).

FACT *believe*(**s,**
not typeWanted(nails,
fingerNails)).

FACT *believe*(**s,**
typeWanted(nails, pictureNails)).

FACT *believe*(**s, knowif(s,**
typeWanted(nails, fingerNails))).

FACT *hasGoal*(**s, do(b, informif(b, s,**
have(b, nails))), ts0).

Bridget's perspective

DEFAULT $(1, credulousBA(P))$:
$believe(\mathbf{s}, P)$).

FACT *believeA*(**b,**
not have(b, pictureNails)).

FACT *believe*(**b, knowif(s,**
typeWanted(nails,
fingerNails))).

FACT *hasGoal*(**b, do(s, informif(s, b,**
typeWanted(nails,
fingerNails))),
ts1)

As mentioned above, although we would not expect Susan and Bridget to have many of these beliefs about nails and types of nails until later in the conversation, we must assume them at the start because the current model does not account for any updates to beliefs.

Below are the key inferences that Susan and Bridget make; note that they are again identical:

Susan's perspective	Bridget's perspective
T1 generation *adoptPlan*(**s, b, askif(s, b, have(b,nails)), informif(b, s, have(b, nails)), ts0)**	**T1** interpretation *adoptPlan*(**s, b, askif(s, b, have(b,nails)), informif(b, s, have(b, nails)), ts0)**
T2 interpretation *adoptPlan*(**b, s, askif(b, s, typeWanted(nails, fingerNails)), informif(s, b, typeWanted(nails, fingerNails)), ts1)**	**T2** generation *adoptPlan*(**b, s, askif(b, s, typeWanted(nails, fingerNails)), informif(s, b, typeWanted(nails, fingerNails)), ts1)**
T3 generation *acceptance*(**s, informif(s, b, typeWanted(nails, fingerNails)), ts2)**	**T3** interpretation *acceptance*(**s, informif(s, b, typeWanted(nails, fingerNails)), ts2)**
T4 interpretation *acceptance*(**s, informif(s, b, have(b, nails)), ts3)**	**T4** generation *acceptance*(**s, informif(s, b, have(b, nails)), ts3)**

These utterances had the following discourse-level interpretations by each agent at the time each utterance was produced:

Susan's perspective	Bridget's perspective
T1 askif(s, b, have(b, nails))	**T1 askif(s, b, have(b, nails))**
T2 askif(b, s, typeWanted(nails, fingerNails))	**T2 askif(b, s, typeWanted(nails, fingerNails))**
T3 informif(s, b, typeWanted(nails, fingerNails))	**T3 informif(s, b, typeWanted(nails, fingerNails))**
T4 informif(b, s, have(b, nails))	**T4 informif(b, s, have(b, nails))**

Note that these interpretations are identical. This is just an assumption; however, the dialogue gives no overt reason for supposing they would be different.

Thus, aside from the limitations mentioned, the model accounts for Susan's ability to tolerate a failure of her expectations without requiring a repair. It also accounts for both agents' ability to return to the top-level sequence when the subdialogue is complete. Consider now the kinds of belief change that would be required to handle this example more accurately: For Bridget to produce T4 while satisfying the planning rule conditions, she must believe that she knows whether she has nails; however, this was not true at the start of the dialogue, because the referent of "nails" was not known to Bridget at T1. On the basis of what is expressed in T3, Bridget apparently revises her actual beliefs about her own knowledge. Currently, the model allows an agent to assume that any activated supposition is believed, and that supposing **not typeWanted(nails,fingerNails)** is enough to justify a belief about **knowif(b, have(b,nails))**. Both assumptions are problematic.

The first assumption leads to the problem with Perrault's (1987) default logic approach—agents can convince themselves of propositions just by uttering them. The only solution would be to incorporate some psychological model (like a user model) to control belief revision. The second assumption ignores the other types of nails (*e.g.,* toenails). This problem is an artifact of the model's inability to allow more than one speech act per turn. To prevent it, the representation of T3 would need to express both the supposition of **not typeWanted(nails, fingerNails)** and **typeWanted(nails, pictureNails)** as in the following:

T1	s:	**askif(s,b,have(b,nails))**
T2	b:	**askif(b,s,typeWanted(nails,fingerNails))**
T3	s:	**informif(s,b,typeWanted(nails,fingerNails)) and inform(s,b,typeWanted(nails,pictureNails))**
T4	b:	**informif(b,s,have(b,nails))**

This representation is actually closer to what Susan actually said, but for the model to handle this representation, it would have to account for a conjunction of acts being used to satisfy an expectation (and possibly also for T2 being both a yes–no question (*i.e.,* **askif**) and a wh-question (*i.e.,* **askref**) at the same time).

7.4 Discussion

7.4.1 The 'Meeting' example

In this example, we saw how the model is able to simulate the production and interpretation of a fourth-turn repair, given only a few (private) prior beliefs and goals. When the system takes the role of Russ, it implicitly displays its decision to interpret Mother's question in T1 as a **pretelling** by producing what it believes to be the expected reply, an **askref**. After T3, it attributes a misunderstanding to itself, but does not attempt to identify the source of the misunderstanding. (Any alternative interpretation would require it to assume that Mother intends to express both that she knows who is going and that she does not know who is going.) After T4, it reinterprets T1 as an **askref** and then displays this repaired understanding in T4 by producing the relevant next response, an **inform**. When the system takes the role of Mother, it displays its decision to interpret Russ's T2 as an **askref** by producing a relevant **inform** in T3. After Russ's T4, it attributes his **surface-inform** to his having understood her turn T1, because the act accomplishes the **informif** that T1 had been expecting. From both perspectives, although Mother's expectations have been satisfied at the end of the conversation, each agent has reached this conclusion having made very different inferences and assumptions at each step. Moreover, their interpretations of T1 still differ.

7.4.2 The 'Dinner for Alfred' example

In this example, Speaker A asks a question that Speaker B apparently mistakes for a test, causing A to produce an explicit third-turn repair. When the system simulates the role of B, the agent that produces the turn that triggers the repair, it implicitly displays its decision to interpret A's question in T1 as a **test** by producing what it believes to be an expected reply: an **askif**. Then, on the basis of the repair produced in T2, it decides in T3 that it has misinterpreted T1. Lastly, it displays its decision to reinterpret A's question in T1 as a **askref** by producing what it believes to be an expected reply: an **inform**. In doing so, B produces a fourth-turn repair. When the system simulates the role of A, the repair-generating agent, it first explicitly displays its decision to attribute B's question in T2 to a misunderstanding of A's T1 by producing a third-turn repair. In its next turn, it attributes B's **inform** in T4 to B's having understood A's turn T1, because, it is an expectable reply to an **askref**. At the end of this conversation, the models of both agents suppose that A's expectations have been satisfied. Again, the assumptions that each agent makes to explain the dialogue differ, but in a complementary way.

7.4.3 The 'Susan wants nails' example

In this example, there is an expectation failure, but no misunderstanding or repair. The model's account of the example demonstrates that it can distinguish different types of failure, and not always attribute them to misunderstanding. When the model is used to simulate either agent, each turn implicitly displays understanding of the previous one, and not just when it is the expected reply, but also when it has a coherent explanation in terms of goals or expectations raised earlier in the exchange. At the end of a simulation, the models of both agents suppose that their expectations have been satisfied; moreover, the assumptions and inferences that each agent makes to explain the dialogue are the same.

7.4.4 Summary and limitations

Together, the examples in this chapter illustrate how the model accounts for conversational exchanges involving third and fourth turn repairs and also expectation failures that might have led to repair, but did not. In the simulations, speakers formed expectations on the basis of what they heard, monitored for differences in understanding, reinterpreted utterances, and generated repairs. Speakers used the discourse context and conventions for interaction to select speech acts that were appropriate and, during interpretation, they used this same information abductively to identify a goal, expectation, or misunderstanding that might have led the other agent to produce it. The examples also highlight some of the limitations of the present model: the need to incorporate belief revision; the need to relax the requirement that there be a one-to-one correspondence between speech acts and turns; the need to treat referring as an action (Heeman and Hirst 1992); the need to incorporate additional types of repair (*e.g.,* repetitions to repair mishearing); and the need to allow for strategic uses of repair.

Chapter 8

Possible extensions to the model

In this chapter, we consider some important issues that have not been not addressed by this research. I will suggest possible approaches to these problems, identifying where existing work could be incorporated and where new research would be required. Section 8.1 discusses the relationship between discourse reasoning and sentence processing. Section 8.2 discusses user modeling. Section 8.3 considers the problem of representing real beliefs. Section 8.4 discusses belief revision.

8.1 Sentence processing

This research has made the simplifying assumption that ambiguities that arise at the discourse-processing level can be resolved independently of the ambiguities that arise at the sentence-processing level (including lexical, syntactic, and semantic ambiguities). Unfortunately, ambiguities that originate during sentence processing might not become apparent until they create an inconsistency at the discourse level. Moreover, sometimes resolving the trouble requires making a choice on the basis of information provided at a different linguistic level than where it originated (Véronis 1991). Failures of reference (see Sections 2.3 and 7.3) are an example of this, as we can see in the following exchange from Schegloff (1987):

Example 8.1 'Which ones are closed?'

T1 **A:** Which ones are closed and which ones are open?

T2 **Z:** Most of them. (pointing to a map) This, this, this, this

T3 **A:** I don't mean on the shelters, I mean on the roads.

T4 **Z:** Oh!

In this example, Z displays an understanding of T1 as a request by attempting to comply, but, from A's perspective, the inconsistency of Z's pointing to symbols of the wrong sort, while apparently attempting to be helpful, displays Z's misunderstanding of the referent of the pronoun "ones". Reference failures can also involve definite noun phrases. Here is an example from my own experience (just after Graeme began his sabbatical), in which a misunderstanding arose over determining the referent of "office number":

Example 8.2 'Office number'

T1	**Susan:**	Do you know Graeme's office number?
T2	**Jan:**	No. (starts looking through files) What do you want it for anyway?
T3	**Susan:**	I want to phone him.
T4	**Jan:**	Oh. I thought you meant the room number. I didn't think you meant phone number because I knew he sent it to everyone so you'd have it.

Similar kinds of misunderstandings can arise from lexical ambiguities, as shown by Example 8.3 in Clark and Schaefer (1989) (originally from the London-Lund corpus).

Example 8.3 'Who evaluates it?'

T1	**A:**	Who evaluates the property?
T2	**B:**	Uh, whoever you ask, the surveyor for the building society.
T3	**A:**	No, I meant who decides what price it'll go on the market.

What this evidence suggests is that there needs to be some form of interaction between discourse and sentence processing. For example, the repair mechanism might need to be able to generate alternative interpretations all the way down to the word-sense level, not just alternative discourse-level forms. An agent might also attempt to prevent such misunderstandings by combining information from the discourse-level with other sources of information available during word and sentence processing (McRoy 1992; McRoy and Hirst 1990).

The decisions about what information to use and when to use it brings us past the limits of computer science and into the realms of psychology and engineering. If we peek over the fence a bit, what we find is that whether or not it is psychologically plausible (or computationally worthwhile) for an agent to invest this sort of effort to avoid misunderstanding depends on the task the agent is performing, the medium of communication, and the relative costs associated with avoiding and repairing failures (Clark and Brennan 1991). Agents apparently try to somehow minimize their collective effort (Clark and Wilkes-Gibbs 1986). They might also apply a fixed strategy of cautiousness or risk-taking, depending on the task (Carletta 1991). Agents also balance goals of understanding with social goals (such as not stirring up trouble) and hence sometimes choose not to repair an apparent misunderstanding explicitly. The following multiparty conversation, recorded during a Ph.D. oral examination by Grimshaw (1982), involves a misunderstanding of this sort.[1]

[1]The analysis of this conversation is Grimshaw's, who participated in the discourse and later discussed it with the participants.

Example 8.4 'Is the dissertation representative?'

T1	**P:**	To what extent are these, the three theories that she selected, truly representative of theories in this area.
T2	**A:**	They are indeed. That's it. That's it.
T3	**S:**	Yeah. Yeah.
T4	**A:**	No more theories.
T5	**P:**	Oh. They are *the* theories?
T6	**A:**	Ah, well, ...
T7	**P:**	A sample of one? In each, ok.
T8	**A:**	Yuh. No, well, except for the sense that ...
T9	**P:**	So nobody, nobody would attack her on that ground then.
T10	**A:**	If she
T11	**B:**	No.
T12	**A:**	Oh, no. I don't think so. I think the only thing that would be, substantially different would be, a real social structuralist, who would say you don't have to worry about cognitions, what you have to do is find the location of these people in the social structure and then you'll find out how they're going to behave without having to get into their heads at all. And, that hasn't been tested.
T13	**B:**	Except in very gross kinds of ways with macro data, which is generally ...
T14	**C:**	... not been very satisfactory. Yeah.
T15	**P:**	Mm,hm? Mm, hm.
T16	**S:**	Right. Spilerman's stuff.

The source of the misunderstanding is in T1, in which there is an ambiguity in the meaning of the word "representative" as either 'sample' or 'exhaustive'. Agent P produces T1 intending to ask whether the theories discussed were just samples drawn from a wider class (which might be construed as a complaint); however, responses by Agents A and S (in T2–T4) indicate that they understood her to be asking whether the theories discussed cover all the salient ones. Their enthusiastic, affirmative reply thus displays a potential misunderstanding to P. In T5 and T7, P tries (unsuccessfully) to clarify her intention, but opts not to perform an explicit repair; in T9 and T15 she abandons this effort and accepts the other interpretation. As this example indicates, a complete solution to the problem of repair must consider the general policies and limitations of an agent, perhaps using them as a filter on possible goals and actions. The question of how to implement such solutions is addressed, in part, by work on user modeling, the topic of the next section.

8.2 User modeling

Currently, the theory does not distinguish between different actions that are consistent with an agent's expectations. When the agent's beliefs allow for a number of possible interpretations, an oracle is used to pick one of them to pursue. What is missing is that, in addition to the expectations that arise during the discourse itself, prior beliefs and expectations about what the participants are likely to do (or what the others might expect of them), knowledge about stereotypic actions, and social conventions also help guide the decisions that agents make. No simple strategy, such as a static preference ordering on interpretations, will suffice because it is clear that agents form different interpretations depending on who is talking and under what circumstances they are doing so. A more reasonable strategy would be for agents to make default assumptions dynamically, on the basis of what they know or observe. For example, Speaker1 might assume that if Speaker2 is (or is likely to be) an expert about some object X, perhaps because S2 performed some action requiring expertise in the past, then it is unlikely that S2 will express **not knowif**$(S2, \Psi(X))$ for some property Ψ. This sort of reasoning is one of the goals of *user modeling.*

There is an extensive literature on user modeling (see Kobsa and Wahlster 1988, and references therein), which we cannot consider exhaustively here. The following are the key issues addressed by this work:

1. Contents: A user model might include descriptions of the plans and goals of an agent, the capabilities of the agent (*e.g.,* whether she or he is naive or expert), the limitations of the agent with respect to learning, memory, and other cognitive activities, potential biases of the agent, and how particular terms or concepts affect his or her attitudes.

2. Specialization: A user model might be generic with respect to a broad class of users, specific to a given individual, or some combination of these. Generic models include information assumed to be common to all users (such as a fixed hierarchy of plans and goals). Individual models represent information about only a single user. Models that combine the two approaches match the user to stereotypical clusters of characteristics that tend to co-occur and then tune them for the individual.

3. Plasticity: A user model might include information that is maintained dynamically through the interaction (*e.g.,* plans and goals) or is acquired beforehand (*e.g.,* age, sex, occupation).

4. Function: A user model may be used for the storage of information or for simulation; for example, a system might simulate an agent's reasoning in order to predict his or her actions.

Van Arragon (1990) describes how a nested version of prioritized Theorist (*i.e.,* a version that supports metalevel reasoning) might be used as a framework for user modeling. In the proposed framework, a user is represented as an object-level reasoning procedure comprising facts and defaults about the user's abilities (including their ability to perform inference). In addition, some of the user's abilities can be inferred on the basis of observations of the user's behavior. On the basis of such a model, the system would then be able to perform metalevel reasoning to simulate the user's inference procedure and thereby infer the agent's beliefs, goals, and other abilities. For example, on the basis of the following model, a system could deduce that user u is capable of using the ls command and the $login$ command:

$\mathcal{F}_s(loggedIn(u))$

"The system has it as a fact that the user is logged in."

$\mathcal{F}_s(\forall A1\ loggedIn(A1) \supset capable(A1, login))$

"The system has it as a fact that if an agent is logged in, then the agent is capable of using the $login$ command."

$\Delta_s(loggedIn(A1) \supset capable(A1, ls))$

"The system has it as a default that agents who are logged in are capable of using the ls command."

8.3 Modeling real (versus expressed) belief

The model developed in this thesis requires that the real beliefs of an agent being modeled be given as an explicit set of facts. No special properties are attributed to these beliefs; for example, they do not necessarily include tautologies, nor beliefs about beliefs. No position is taken on the source of the agent's beliefs nor the truth of the beliefs. Although specifying a theory of beliefs has been one of the goals of Knowledge Representation, current work has been unsatisfactory for use in modeling discourse. In this section, I will try to explain why this is so, and suggest directions for a more appropriate account.

Typical KR accounts of belief focus on beliefs as a set of statements in a (modal) logic, divorced from the events of agency that generate them. Levesque (1984) points out that models of belief that have the property of logical omniscience are best thought of as models of what is *implicit* in an agent's beliefs, rather than what the agent believes directly. For the task of modeling communication this distinction is very important, because it is the information that communicators *actively* hold to be true that influences how they reason about utterances.[2] But formal attempts to characterize what an agent actively believes have all been problematic in some regard. However, underlying these

[2]This point is an empirical one, supported by the distinction between semantic and episodic mem-

difficulties is an issue of far greater significance: in particular, for the purposes of modeling discourse, the focus on formal dependencies among beliefs and knowledge misses the primary target. In fact, the right target for this application is the mental acts from which agents' beliefs originate.

The suggestion that I make here is that the beliefs that agents use to interpret language *need not exist* until *after* the agents begin to assimilate an utterance and the situation in which it was produced. During such activities, agents access their prior assumptions and experiences, but do so indirectly, through intentional acts of introspection. Moreover, they will not attempt to access every aspect of their prior experience, nor will every such aspect be accessible. We would like to call 'beliefs' all and only those effects that result from an act of introspection. Discourse modeling will derive very little benefit from specifications of the logical properties of belief that do not consider the mental actions that give rise to it. The rest of this section will serve to justify these views.

8.3.1 A bit of background

As we have seen throughout this thesis, to discriminate among alternative interpretations or to design utterances that another agent will be able to interpret, discourse participants must reason about the beliefs of one another and of themselves. For example, if a parent, S1, standing beside a working grandfather clock, says to her child, S2, after S2 walks in the door, "Do you know what time it is?", S2 can rule out the (sincere) interpretations corresponding to a question about S2's knowledge and a request for the time if S2 actively believes that they both know the time and that S1 believes S2 knows the time; a warning or a complaint would be more likely, under the circumstances.

So-called syntactic approaches to belief, including explicit enumerations of an agent's beliefs, are problematic, because they attempt to capture everything that an agent believes for all possible situations and yet are sensitive to the form in which a belief is expressed; for example, $Bel(X, A \wedge B)$ and $Bel(X, B \wedge A)$ are not necessarily treated as equivalent. For such a comprehensive theory, one would like to represent dependencies among an agent's beliefs, so that not all beliefs (which might be infinite in number) would need explicit representation (Levesque 1984).

Proposed semantic accounts of belief provide greater expressive power; however, they too are problematic, because they attribute to an agent too many (irrelevant) beliefs. Consider, for example, the axioms that define the ever-popular "Weak S5" modal logic of belief:

$$Bel(X, P) \wedge Bel(X, P \supset Q) \supset Bel(X, Q)$$
$$Bel(X, P) \supset Bel(X, Bel(X, P))$$
$$\neg Bel(X, P) \supset Bel(X, \neg Bel(X, P))$$
$$Bel(X, P) \supset \neg Bel(X, \neg P)$$

ory (Tulving 1972). Semantic memory holds time-independent facts such as the rules of language (*e.g.,* the grammar and lexicon) and the meanings of concepts. By contrast, memories of time-dependent, contextual, and pragmatic information are episodic. This information is accessed by actively searching the memory, guided by cues from the retrieval context, and evaluating the results with respect to that context.

Under these axioms, agents are logically omniscient and have perfect introspection. Typically, one also needs to add to these axioms a rule of necessitation, $P \vdash Bel(X, P)$, attributing to agents even more beliefs, most of which they would never use or even be aware they had.

Logical omniscience is a particularly implausible quality for belief, because it attributes to agents the following powers (Fagin and Halpern, 1988):

- They believe all the implications of their beliefs.
- They believe all tautologies.
- If they believe α, they also believe all formulae that are logically equivalent to α.
- If, for any α, they believe both α and $\neg\alpha$, then they believe *every* sentence in the language.

Unfortunately, proposed solutions to this problem within the semantic approach, such as those of Levesque (1984) and Fagin and Halpern (1985), introduce their own problems.

Levesque's theory of explicit belief avoids logical omniscience by defining explicitness using relevance logic and making special use of inconsistency. In this logic, situations provide truth values for some of the atomic sentences, ignoring information that is irrelevant to the agent's beliefs. A formula α is believed if it is true in all the situations that could be the actual one. Beliefs are not closed under logical implication, because a situation may make both α and $\neg\alpha$ true. For example, the following is satisfiable:

$Bel(X, \alpha) \wedge Bel(X, \alpha \supset \beta) \wedge Bel(X, \neg\alpha)$

which prevents the inference of $Bel(X, \beta)$ from $Bel(X, \alpha) \wedge Bel(X, \alpha \supset \beta)$. Unfortunately, allowing inconsistent situations also causes the logic to have some undesirable properties; for example, formulae such as the following are valid:

$Bel(X, \alpha) \wedge Bel(X, \alpha \supset \beta) \supset Bel(X, \beta \vee (\alpha \wedge \neg\alpha))$

In addition, it does not seem appropriate to say that an agent actively believes such inconsistencies; communication seems to depend on the following intuitions about human reasoning (Harman 1986, p. 11, emphasis added):

- The fact that one's view logically implies P can be a *reason* to accept P.
- Logical inconsistency is to be avoided.

Fagin and Halpern's (1985) theory of belief better preserves these intuitions, while still addressing the problem of logical omniscience. Their approach is to filter irrelevant beliefs with a syntactic awareness function. A formula α is believed in a world if the awareness set for that world includes α and α is true in all worlds accessible from that world. Although awareness can be defined so that it prevents logical omniscience, being syntactic, the account suffers from the same problem of over-sensitivity to the form of the formulae that plagued previous syntactic accounts of belief (Konolige 1986b). There is also a difficulty associated with defining an appropriate awareness function for an application; for discourse modeling, we would need one that is task-dependent.

A third set of approaches to belief that we shall consider are the deductive ones. According to these accounts, beliefs are those conclusions that can be derived if given a theory, a set of inference rules, and a control strategy. Limited reasoning is captured either through a logically incomplete set of inference rules (Konolige 1986a) or through a limited control strategy (van Arragon 1990). For example, Konolige prevents logical omniscience by defining a deduction rule that is similar to *modus ponens*, but places a bound on the number of steps in a derivation. Van Arragon's approach is nearly the same, except that it relies on metareasoning and represents such limitations declaratively by means of metalanguage axioms such as the following:

$$
\begin{array}{l}
(\forall G,\ B, G1, B1, G2, B2) \\
\qquad \mathcal{F}_{A1}(\mathit{infer}_{A2}(B1 \supset G1) \wedge \mathit{infer}_{A2}(B2 \supset G2) \\
\qquad\qquad \wedge (G \neq G1 \wedge G \neq G2 \wedge G1 \neq G2) \\
\qquad\qquad \supset \neg \mathit{infer}_{A2}(B \supset G))
\end{array}
$$

The axiom specifies that it is a fact of agent $A1$ (*e.g.,* the system) that the control strategy (linear resolution[3]) characterizing the agent $A2$ can chain on facts to a depth of two. (With this approach, one can introduce a similar metalevel predicate to model awareness for a given agent as well.) Deductive approaches are syntactic because a decision to represent a belief set as either $\{p \supset q, q \supset r\}$ or $\{p \supset q, q \supset r, p \supset r\}$ will determine whether or not the agent believes r for a given bound. Moreover, because these approaches focus on the depth of inference, rather than the breadth, the potential for a large number of irrelevant beliefs is only reduced, not eliminated.

8.3.2 An alternative view

From the perspective of discourse modeling, the greatest steps forward in modeling beliefs have been the introduction of the notion of awareness and the idea that there might be a resource constraint on inference. What all these approaches (syntactic, semantic, and deductive) still ignore, however, is the agent's own involvement in the process of introspection. Introspection is an act of will. Awareness is a property of an agent with respect to an object such as a statement in a knowledge base; this property might or might not hold at any given time. To capture these intuitions, I propose an alternative view that I will call the Effect Theory of Belief.

According to the Effect Theory, an agent has no persistent model of beliefs and hence no irrelevant beliefs in it. Instead, a belief is modeled as an effect (*i.e.,* an event or an end state) of an intentional act of introspection (Haas 1992; Sadek 1992). Moreover, such effects must be sensitive to the goals and perspective of the actor.[4] Previous formal accounts of belief are inadequate for discourse modeling, precisely because they treat introspection as primitive. The parameters of an introspective act, along with the preconditions, constraints, and actions that compose it, determine the relevant domain

[3] *infer* is a metalevel default corresponding to one step of resolution.

[4] There is empirical support for this claim. In one set of experiments by Anderson and Pichert (Anderson and Pichert 1978) subjects were read a story after being told information that placed it within a particular perspective (*e.g.,* a burglar), and then asked to report all that they could recall about it. Later, subjects who were given information placing the story in a different perspective (*e.g.,* a home buyer) were able to recall previously unreported information. However, similar subjects who were not given a new perspective, could recall no new information.

and range; inference chaining is achieved only by a sequence of acts, although common sequences might be compiled (Cohen and Levesque 1990b).

The specification of introspective actions might be represented with any of a number of formalisms for reasoning about action, such as the Event Calculus (Kowalski and Sergot 1986) or the Situation Calculus (McCarthy and Hayes 1969). Following accounts of other actions, for each act of introspection the theory would specify the steps of its possible decompositions into more primitive mental operations, any constraints on their application, and the (conditional) effects of the act.

Possible decompositions (strategies) for introspection would include these:

- Construct a logic problem, that is, retrieve a fixed set of facts (including axioms) and use them to construct a derivation of some goal concept.
- Recall an abstract description of an event (such as a frame) and see if it satisfies a given property.
- Apply inheritance reasoning.
- Free-associate, that is, activate a set of concepts within an associative network (*e.g.,* abstraction and decomposition relations) and apply marker-passing and path-evaluation techniques (Hendler 1988).
- Retrieve an image, diagram, or memory sequence and either check it directly or use it to construct an abstract description and then check that (Myers and Konolige 1992).
- Retrieve a set of objects and apply inductive reasoning.

The appropriate syntactic form for expressing such decompositions (*cf.* Horton and Hirst, 1992) is a topic of current research; a solution depends on our having a better understanding of other mental acts such as analysis, synthesis, disregard, deliberation, and choice.

In an Effect Theory framework, the general method for examining one's beliefs would be to select a decomposition of introspection (*e.g.,* through planning) and apply each step. As with other types of action, the effects of an introspective act will depend on the type of the act and its actual parameters. For a given sequence of acts, it is possible that inconsistent beliefs might result (although probably not often); however, this would not cause the reasoning system to collapse. Rather, it might be used to trigger a subgoal to perform another act (possibly another introspection) to resolve the inconsistency.

Representing belief within an appropriate theory of action would accomplish a number of important goals that have not been achieved concurrently in the past:

- The problem of logical omniscience would be seen for what it truly is, a limitation on introspection.
- The beliefs of an agent need not be dependent upon the logical form in which the information is stored (the advantage of the semantic approach).

- The beliefs of an agent would be sensitive to the syntactic features of how the information is queried—that is, what an agent believes in a given situation might depend on what questions the agent poses and the mechanisms selected for answering them.
- The beliefs of an agent may be inconsistent without triggering a logical collapse.
- With a similar treatment of other mental acts such as 'attend' and 'decide', there would be a natural link between belief and belief change.

Language use is a collaborative activity—no one agent is expected to do all the work and an agent might choose to do more or less depending on how much work this would impose on others. The amount of work any agent might expend on introspection should be guided both by social conventions for reasonable behavior and by the desires of the agent. An action-based account of belief, such as the Effect Theory, can capture both these normative and goal-directed aspects.

8.4 Belief revision

Related to the problem of determining what an agent believes is the problem of specifying how his or her beliefs might change. Over the course of a dialogue, discourse participants must make a number of default assumptions about one another's beliefs. In the current account, these assumptions remain defaults and have no causal effect on other beliefs and goals; the beliefs of an agent remain unchanged throughout the course of a dialogue. This simplification has been possible because the conversations we have modeled have all been quite short. To handle longer exchanges, the model would need to be able to account for how agents assimilate new beliefs and how these new beliefs subsequently influence the agents' goals. It would also need to specify how agents estimate the credibility of the evidence provided by the discourse and the situation of its production (Clark and Marshall 1981; Walker 1991). (User modeling, discussed above, attempts to address some of these concerns.)

Although ultimately an action-theoretic model of belief revision would be most appropriate for the theory of explicit belief discussed above, it is useful to consider the related work within existing frameworks. One line of research on the revision problem involves attempts to provide better default models of actual belief. Perrault (1987) (discussed in Section 1), for example, allows information about the content of the discourse to be used as a basis for default belief (*i.e.,* $\mathbf{DO}_{x,t}(Utter\ P) \Rightarrow \mathbf{B}_{x,t}P$); in the proposed model, we do this by allowing a default belief for each supposition. As stated, this simple revision strategy is problematic because a speaker could convince him- or herself of something simply by uttering it; the solution is to give such default assumptions lower priority than default assumptions about the persistence of the speaker's prior beliefs (Appelt and Konolige 1988).

Another line of research involves specifying how best to maintain the consistency of a set of logical statements representing an agent's beliefs. The foundational theory of belief revision assumes that any belief is either self-evident or has a non-circular, finite sequence of justifications made up of other beliefs (Harman 1986). Foundational revision involves adding justified (or self-evident) beliefs and removing any beliefs whose justifications are unsatisfactory. This approach thus requires one to maintain the complete sequence of justifications for each belief. Although in general this approach would be too memory-consuming to be what resource-limited human reasoners actually do (Gärdenfors 1988), for problems of limited size, programs known as truth maintenance systems can be used.

One less resource-intensive alternative to foundational revision is what Harman (1986) calls coherence-theoretic belief revision. This approach assumes that beliefs are justified simply by being a part of a coherent theory, not necessarily the result of any deductive chain of justifications. Consistency of one's beliefs is maintained by making a minimal set of changes; beliefs that lose their justification, but are not explicitly contradicted, are unaffected. This approach does not require keeping unneeded justifications and permits some latitude in choosing a revision, since in general there is no unique minimal change. Various sets of selection criteria have been defined by means of axioms or postulates that a belief-revision operator must satisfy (Dalal 1988; Gärdenfors 1988; Nebel 1989; Rao and Foo 1989). (Nonmonotonic formalisms such as circumscription (Winslett 1989) and probabilistic logic (Neufeld 1989) have also been used to specify minimal revisions.)

Chapter 9

Conclusion

9.1 Goals and achievements

In human dialogues, both the producer and the recipient of an utterance have a say in determining its interpretation. Moreover, they may both change their minds in the face of new information. Dialogue participants are able to *negotiate* the meaning of utterances because when a hearer responds to what he decides the speaker wants or expects regarding an utterance, he also provides evidence of that decision and hence constraints on what the speaker may do next. If the speaker disagrees with a displayed interpretation, she can challenge it directly or decide to respond in such a way that the hearer must infer a misunderstanding. People are guided in their interpretations by a number of interacting goals. Some of these goals are task-driven; for example, one might want to get someone to do something. Some of them are more general; for example, one might want to figure out why someone said something or what would be a reasonable response (or simply how she or he can keep the conversation going). Thus, meaning is not just something that disourse participants find or encode, it is something they *achieve* through their interactions with each other.

The goal of this work has been to construct a model of communicative interaction that would be able to support the negotiation of meaning. We have considered the information sources and reasoning processes that agents need in order to identify goals and expectations that might explain each other's utterances. Whereas previous models of dialogue tend to represent discourse meaning from some global perspective, make use of either purely structural or purely intentional information, and give minimal attention to repair, in the new model:

- Each agent has his or her own model of the discourse.
- Agents rely on both structural and intentional information in the discourse.
- Agents distinguish between intended actions and misunderstandings.
- Agents interpret utterances on the basis of expectations derived from previous utterances as well as expectations for future actions that are predicted by the utterance under interpretation.

- Agents are able to detect and repair their own misunderstandings as well as those of others.

According to this model, different agents sometimes form different interpretations of the same surface level utterance, because of differences in their actual beliefs or differences in the assumptions they made to explain earlier parts of the dialogue. Moreover, even when agents assign the same discourse-level speech act to a given utterance, they might have different reasons for doing so, because of differences in their prior beliefs and assumptions. Despite the significance of prior beliefs, however, when misunderstandings manifest themselves as actions that are inconsistent or incoherent, agents can identify and repair them without reasoning about the source of misunderstanding (*e.g.,* without actually identifying the differences in each other's plans or beliefs). Thus, it has not been necessary to represent mutual beliefs in order to account for the negotiation of understanding by discourse agents to handle this class of misunderstanding (*cf.* Thomason 1990).

9.2 Directions for future work

This research provides some steps toward a unified account of interpretation, generation, and repair. There are a number of important directions in which it could be expanded:

1. Other types of speech acts: At present, the model is limited to considering actions that correspond to exactly one turn, but speech acts can span more than a single turn and turns can accomplish more than a single act. Although I believe that the same general approach will apply, the level of ambiguity will increase.

2. Other types of misunderstandings and possible interactions among them: For example, misunderstandings also arise because of ambiguous references and discourse structure; currently we consider only misunderstandings that result from speech act ambiguities.

3. Other forms of repair: The repair types that have been modeled have been limited to acceptances of the reinterpretation or challenges that name the intended act. Repairs might also take the form of a challenge to a presupposition of the misunderstood turn, as in the following constructed example:

Example 9.1 'Money'

T1	**A:**	Do you have any idea how much money he makes?
T2	**B:**	How much?
T3	**A:**	No, I'm asking you.
T4	**B:**	You know that I can't tell you.

---

5. Other means of selecting interpretations: So far, adjacency pairs have been treated as the primary source of expectations and inconsistency as the sole filter on these expectations. The complete role of actual beliefs and intentions, involving issues of sincerity, cooperativeness, and commitment, has yet to be addressed. However, these are important research areas in their own right; our model can provide an empirical testbed for the development of these ideas.

6. The reintegration of discourse-level reasoning and domain planning: Although it is important to keep the discourse and domain levels distinct in order to best exploit the structure of the discourse level, it is still the planning level that determines whether an agent would prefer to accept, challenge, or pursue some new goal.

7. An explanation-based account of indirect speech acts: Although domain-level reasoning will always be necessary to identify extreme cases of indirectness, many can be interpreted by simply exploiting the constraints of the discourse-level. Here, it will be important to distinguish indirect utterances that introduce an adjacency pair from indirect replies. It appears that questions and requests are more conventional than their replies, but even the interpretation of the most difficult cases can benefit from constraints provided by the discourse level.

8. The integration of discourse-level reasoning and sentence processing: At present, we assume that syntactic and semantic processing can be done first, without input from the discourse level. This is reasonable as a first step, but ultimately the model should be able to use expectations from the discourse level to help guide the resolution of ambiguities at the sentence level. Conversely, the model should have more access to surface cues in the input, including prosody and lexical discourse markers, to guide discourse processing.

9.3 Contributions and applications

The primary contribution of this work is that it treats misunderstanding and repair as intrinsic to conversants' core language abilities, accountable with the same processing mechanisms that underlie normal speech. In particular, it formulates both interpretation and repair as explanation problems and models them as abduction. To support the repair of misunderstandings, the model incorporates a representation of the discourse that captures the agent's interpretation of the conversation both before and after a repair. This representation is independent of the actual beliefs of the participants—a dynamic mental artifact that is the *object* of belief and repair. This type of discourse record allows agents to refer to alternative interpretations or to the repair process itself, potentially enabling them to recover from rejected repairs. By addressing the problems of repair, this work should facilitate efforts to build natural language interfaces that can better recover from their own mistakes as well as those of their users.

References

Ahuja, Sanjiev B. and Reggia, James A. (1986). The parsimonious covering model for inexact abductive reasoning in diagnostic systems. In *Recent Developments in the Theory and Applications of Fuzzy Sets. Proceedings of NAFIPS '86 - 1986 Conference of the North American Fuzzy Information Processing Society.* pages 1–20.

Allen, James and Perrault, Raymond (1979). Plans, inference, and indirect speech acts. In *17th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference.* pages 85–87.

Allen, James F. (1979). *A Plan-Based Approach to Speech Act Recognition.* PhD thesis, Department of Computer Science, University of Toronto, Toronto, Canada. Published as University of Toronto, Department of Computer Science Technical Report No. 131.

Allen, James F. (1983). Recognizing intentions from natural language utterances. In Brady, Michael, Berwick, Robert C., and Allen, James F., editors, *Computational Models of Discourse.* The MIT Press, pages 107–166.

Allen, James F. (1984). Towards a general theory of action and time. *Artificial Intelligence*, 23:123–154.

Allen, James F. (1991). Planning as temporal reasoning. In *Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning*, Cambridge, MA. pages 3–14.

Anderson, R. C. and Pichert, J. W. (1978). Recall of previously unrecallable information following a shift in perspective. *Journal of Verbal Learning and Verbal Behavior*, 17:1–12.

Appelt, Douglas and Konolige, Kurt (1988). A practical nonmonotonic theory for reasoning about speech acts. In *26th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference.* pages 170–178.

Appelt, Douglas E. and Pollack, Martha E. (1992). Weighted abduction for plan ascription. *User Modelling and User Adapted Interaction*, 2(1–2):1–25.

Austin, J. L. (1962). *How to Do Things with Words.* Oxford University Press, London, England. Reprinted 1975.

Bach, Kent (1990). Communicative intentions, plan recognition, and pragmatics: Comments on Thomason and on Litman and Allen. In Cohen, Philip R., Morgan, Jerry, and Pollack, Martha, editors, *Intentions in Communication.* The MIT Press, pages 389–400.

Bach, Kent and Harnish, Robert M. (1979). *Linguistic Communication and Speech Acts.* The MIT Press.

Beach, Wayne A. (1983). Background understandings and the situated accomplishment of conversational telling-expansions. In Craig, Robert T. and Tracy, Karen, editors, *Conversational Coherence: Form, Structure, and Strategy.* Sage Publications, Beverly Hills, CA.

Beun, Robbert-Jan (1990). Speech acts and mental states. In *Proceedings of the Fifth Rocky Mountain Conference on Artificial Intelligence, Pragmatics in Artificial Intelligence*, Las Cruces, New Mexico. pages 75–80.

Bilmes, Jack (1988). The concept of preference in conversational analysis. *Lang. Soc.*, 17:161–181.

Brennan, Susan Elise (1990). *Seeking and Providing Evidence for Mutual Understanding.* PhD thesis, Department of Psychology, Stanford University, Stanford, CA.

Brewka, Gerhard (1989). Preferred subtheories: An extended logical framework for default reasoning. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, Detroit, MI. pages 1043–1048.

Brown, John Seely and Van Lehn, Kurt (1980). Repair theory: A generative theory of bugs in procedural skills. *Cognitive Science*, 4(4):379–426.

Button, Graham (1990). Going up a blind alley: Conflating conversation analysis and computational modelling. In Luff, Paul, Gilbert, Nigel, and Frolich, David, editors, *Computers and Conversation.* Academic Press, pages 67–90.

Calistri-Yeh, Randall J. (1991). Utilizing user models to handle ambiguity and misconceptions in robust plan recognition. *User Modelling and User Adapted Interaction*, 1(4):289–322.

Carberry, Sandra (1985). *Pragmatics Modeling in Information Systems Interfaces.* PhD thesis, University of Delaware, Newark, Delaware.

Carberry, Sandra (1987). Pragmatic modeling: Toward a robust natural language interface. *Computational Intelligence*, 3(3):117–136.

Carberry, Sandra (1988). Modeling the user's plans and goals. *Computational Linguistics*, 14(3):23–37.

Carletta, Jean (1991). Risk-taking and recovery in task-oriented dialogue. In *1991 AAAI Fall Symposium on Discourse Structure in Natural Language Understanding and Generation*, Pacific Grove, Monterey, CA. pages 7–12.

Cawsey, Alison J. (1991). A belief revision model of repair sequences in dialogue. In Costa, Ernesto, editor, *New Directions in Intelligent Tutoring Systems.* Springer Verlag.

Chapman, David (1992). Computer rules, conversational rules. *Computational Linguistics*, 18(4):531–536.

Charniak, Eugene and Goldman, Robert (1988). A logic for semantic interpretation. In *26th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, Buffalo, NY. pages 87–94.

Clark, Herbert. H. and Brennan, Susan E. (1991). Grounding in communication. In Resnick, Lauren B., Levine, John, and Teasley, Stephanie D., editors, *Perspectives on Socially Shared Cognition*. American Psychological Association.

Clark, Herbert. H. and Marshall, Catherine R. (1981). Definite reference and mutual knowledge. In Joshi, Aravind K., Webber, Bonnie L., and Sag, Ivan, editors, *Elements of Discourse Understanding*. Cambridge University Press, Cambridge, MA, pages 10–61.

Clark, Herbert H. and Schaefer, Edward F. (1989). Contributing to discourse. *Cognitive Science*, 13(2):259–294.

Clark, Herbert H. and Wilkes-Gibbs, Deanna (1986). Referring as a collaborative process. *Cognition*, 22:1–39. (Reprinted in Cohen et al., 1990).

Cohen, Philip R. (1978). *On Knowing What to Say: Planning Speech Acts*. PhD thesis, Department of Computer Science, University of Toronto, Toronto, Canada. Published as University of Toronto, Department of Computer Science Technical Report No. 118.

Cohen, Philip R. and Levesque, Hector J. (1985). Speech acts and rationality. In *23th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*. pages 49–60.

Cohen, Philip R. and Levesque, Hector J. (1990a). Persistence, intention, and commitment. In Cohen, Philip R., Morgan, Jerry, and Pollack, Martha, editors, *Intentions in Communication*. The MIT Press, pages 33–69.

Cohen, Philip R. and Levesque, Hector J. (1990b). Rational interaction as the basis for communication. In Cohen, Philip R., Morgan, Jerry, and Pollack, Martha, editors, *Intentions in Communication*. The MIT Press, pages 221–255.

Cohen, Philip R., Morgan, Jerry, and Pollack, Martha, editors (1990). *Intentions in Communication*. The MIT Press.

Coulthard, Malcolm and Brazil, David (1984). Exchange structure. In Coulthard, Malcolm and Montgomery, Martin, editors, *Studies in Discourse Analysis*. Routledge & Kegam Paul, London, pages 82–106.

Craig, Robert T. and Tracy, Karen, editors (1983). *Conversational Coherence: Form, Structure, and Strategy*. Sage Publications, Beverly Hills, CA.

Dalal, Mukesh (1988). Investigations into a theory of knowledge base revision. In *Proceedings, National Conference on Artificial Intelligence (AAAI-88)*, Saint Paul, MN. pages 475–479.

Eiselt, Kurt Paul (1989). *Inference Processing and Error Recovery in Sentence Understanding.* PhD thesis, Department of Information and Computer Scienc, University of California, Irvine. Published as Technical Report 89-24.

Eller, Rhonda and Carberry, Sandra (1992). A meta-rule approach to flexible plan recognition in dialogue. *User Modelling and User Adapted Interaction*, 2(1–2):27–53.

Eshghi, Kave (1988). Abductive planning with event calculus. In *Proceedings of the Fifth International Conference on Logic Programming.* pages 562–579.

Fagin, Ronald and Halpern, Joseph Y. (1985). Belief, awareness, and limited reasoning: Preliminary report. In *Proceedings of the 9th International Joint Conference on Artificial Intelligence.* pages 491–501.

Fagin, Ronald and Halpern, Joseph Y. (1988). Belief, awareness, and limited reasoning. *Artificial Intelligence*, 34(1):39–76.

Fikes, R. E. and Nilsson, Nils J. (1971). STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2:189–208.

Fox, Barbara (1987). Interactional reconstruction in real-time language processing. *Cognitive Science*, 11:365–387.

Gärdenfors, Peter (1988). *Knowledge in Flux.* The MIT Press, Cambridge, MA.

Garfinkel, Harold (1967). *Studies in Ethnomethodology.* Prentice Hall, Englewood Cliffs, NJ. (Reprinted: Cambridge, England: Polity Press, in association with Basil Blackwell, 1984.).

Genesereth, Michael R. and Nilsson, Nils J. (1987). *Logical Foundations of Artificial Intelligence.* Morgan Kaufmann, Palo Alto.

Goldman, Robert P. and Lang, R. Raymond (1993). Intentions in time. Technical Report TUTR 93-101, Tulane University.

Goodman, Bradley (1985). Repairing reference identification failures by relaxation. In *The 23rd Annual Meeting of the Association for Computational Linguistics: Proceedings of the Conference*, Chicago. pages 204–217.

Grice, H. P. (1957). Meaning. *The Philosophical Review*, 66:377–388.

Grice, H. P. (1975). Logic and conversation. In Cole, P. and Morgan, J. L., editors, *Syntax and Semantics 3: Speech Acts.* Academic Press, New York, pages 41–58.

Grimshaw, Allen D. (1982). Comprehensive discourse analysis: An instance of professional peer interaction. *Language and Society*, 11:15–47.

Gumperz, John J. (1982). *Discourse Strategies.* Cambridge University Press, New York, New York.

Gutwin, Carl and McCalla, Gordon (1992). Would I lie to you? Modelling context and pedagogic misrepresentation in tutorial dialogue. In *30th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, Newark, DE. pages 152–158.

Haas, Andrew R. (1992). A reactive planner that uses explanation closure. In *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning*, Cambridge, MA. pages 93–103.

Harman, Gilbert (1986). *Change in View.* The MIT Press, Cambridge, MA.

Heeman, Peter and Hirst, Graeme (1992). Collaborating on referring expressions. Technical Report 435, Department of Computer Science, University of Rochester.

Helft, Nicolas and Konolige, Kurt (1990). Plan recognition as abduction and relevance. Manuscript.

Hendler, James A. (1988). *Integrating Marker-Passing and Problem-Solving.* Lawrence Erlbaum Associates, Hillsdale, NJ.

Heritage, John (1984). *Garfinkel and Ethnomethodology.* Polity Press, Cambridge, England.

Hinkelman, Elizabeth Ann (1989). *Linguistic and Pragmatic Constraints on Utterance Interpretation.* PhD thesis, Department of Computer Science, University of Rochester, Rochester, New York. Published as University of Rochester Computer Science Technical Report 288, May 1990.

Hirst, Graeme (1991). Does conversation analysis have a role in computational linguistics? *Computational Linguistics*, 17(2):211–228.

Hobbs, Jerry R., Stickel, Mark, Martin, Paul, and Edwards, Douglas (1988). Interpretation as abduction. In *26th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference.* pages 95–103.

Horton, Diane and Hirst, Graeme (1993). Knowledge about planning: On the meaning and representation of plan decomposition. In *AAAI Spring Symposium on Reasoning about Mental States: Formal Theories and Applications*, Stanford, CA. pages 74–78.

Humphrey-Jones, Claire (1987). The structure of misunderstandings. In Reilly, Ronan, editor, *Communication Failure in Dialogue and Discourse: Detection and Repair Processes.* North Holland, Amsterdam, pages 25–33.

Jefferson, Gail (1972). Side sequences. In Sudnow, David, editor, *Studies in Social Interaction.* The Free Press, New York, pages 294–338.

Jose, Paul E. (1988). Sequentiality of speech acts in conversational structure. *Journal of Psycholinguistic Research*, 17(1):65–88.

Josephson, John R. (1987). A framework for situation assessment: Using best-explanation reasoning to infer plans from behavior. In *Proceedings, Knowledge-based Systems Workshop.* pages 76–85.

Kautz, Henry and Allen, James (1986). Generalized plan recognition. In *Proceedings, National Conference on Artificial Intelligence (AAAI-86).*

Kautz, Henry A. (1987). A formal theory of plan recognition. Technical Report TR215, Department of Computer Science, University of Rochester, Rochester, NY.

Kobsa, Alfred and Wahlster, Wolfgang, editors (1988). *Computational Linguistics,* 14(3). Special Issue on User Modeling.

Konolige, Kurt (1986a). *A Deduction Model of Belief.* Morgan Kaufmann, Pitman (London). Pitman Research Notes in Artificial Intelligence.

Konolige, Kurt (1986b). What awareness isn't: A sentential view of implicit and explicit belief. In Halpern, Joseph Y., editor, *Proceedings of the First Conference on Theoretical Aspects of Reasoning about Knowledge*, Monterey, CA. pages 241–250.

Kowalski, Robert and Sergot, M. (1986). A logic-based calculus of events. *New Generation Computing*, 4:67–95.

Kreckel, M. (1981). *Communicative Acts and Shared Knowledge in Discourse.* Academic Press, London.

Levesque, Hector J. (1984). A logic of implicit and explicit belief. In *Proceedings, National Conference on Artificial Intelligence (AAAI-84)*, Austin, TX. pages 198–202.

Levesque, Hector J. (1989). All I know: A study in autoepistemic logic. Technical report, Department of Computer Science, University of Toronto. Knowledge Representation and Reasoning Technical Report KRR-TR-89-3.

Levinson, Stephen C. (1983). *Pragmatics.* Cambridge University Press, Cambridge, MA.

Litman, Diane J. (1986). Linguistic coherence: A plan-based alternative. In *24th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, New York. pages 215–223.

Loveland, D. W. (1978). *Automated Theorem Proving: A Logical Basis.* North-Holland, Amsterdam, The Netherlands.

Mann, William C., Moore, James A., and Levin, James A. (1977). A comprehension model for human dialogue. In *Proceedings of the Fifth International Joint Conference on Artificial Intelligence*, Cambridge. pages 77–87.

McCarthy, John (1986). Applications of circumscription to formalizing commonsense reasoning. *Artificial Intelligence*, 28:89–116. Reprinted in Ginsberg(1987).

McCarthy, John and Hayes, Patrick (1969). Some philosophical problems from the standpoint of artificial intelligence. In *Machine Intelligence 4*. pages 463–502. Reprinted in *Readings in Artificial Intelligence*, Bonnie Webber and Nils Nilsson, editors, 1981, pp. 431–450.

McCoy, Kathleen F. (1985). The role of perspective in responding to property misconceptions. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, volume 2. pages 791–793.

McCoy, Kathleen F. (1986). The ROMPER system: responding to object-related misconceptions using perspective. In *24th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*. pages 97–105.

McCoy, Kathleen F. (1988). Reasoning on a highlighted user model to respond to misconceptions. *Computational Linguistics*, 14(3):52–63.

McDermott, Drew and Doyle, Jon (1980). Nonmonotonic logic I. *Artificial Intelligence*, 25:41–72.

McLaughlin, Margaret L. (1984). *Conversation: How Talk is Organized.* Sage Publications, Beverly Hills.

McRoy, Susan W. (1992). Using multiple knowledge sources for word sense discrimination. *Computational Linguistics*, 18(1):1–30.

McRoy, Susan Weber and Hirst, Graeme (1990). Race-based syntactic attachment. *Cognitive Science*, 14(3):313–354.

Moore, Johanna D. (1989). Responding to "Huh?": Answering vaguely articulated follow-up questions. In *Conference on Human Factors in Computing Systems (CHI'89)*, Austin, TX. ACM Press / Addison-Wesley, pages 91–96. Also published as a special issue of *SIGCHI Bulletin* (unnumbered).

Moore, Robert C. (1984). Possible-world semantics for autoepistemic logic. In *Nonmonotonic Reasoning Workshop, sponsored by AAAI*, New Platz, NM. pages 396–401.

Moore, Robert C. (1985). Semantical considerations on nonmonotonic logic. *Artificial Intelligence*, 25:75–94.

Myers, Karen L. and Konolige, Kurt (1992). Reasoning with analogical representations. In *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning*, Cambridge, MA. pages 189–125.

Nebel, Bernhard (1989). A knowledge level analysis of belief revision. In *Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning.* pages 301–311.

Neufeld, Eric (1989). Defaults and probabilities: Extensions and coherence. In *Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning.* pages 312–323.

Perrault, C. Raymond (1987). An application of default logic to speech act theory. Technical report, Center for the Study of Language and Information. Technical Report CSLI-87-90.

Perrault, C. Raymond (1990). An application of default logic to speech act theory. In Cohen, Philip R., Morgan, Jerry, and Pollack, Martha, editors, *Intentions in Communication*. The MIT Press, pages 161–186. An earlier version of this paper was published as Technical Report CSLI-87-90 by the Center for the Study of Language and Information.

Perrault, C. Raymond and Allen, James F. (1980). A plan-based analysis of indirect speech acts. *Computational Linguistics*, 6:167–183.

Pollack, Martha E. (1986). Inferring domain plans in question-answering. Technical Report TR 403, Artificial Intelligence Center, SRI International, Menlo Park, CA.

Poole, David (1986). Default reasoning and diagnosis as theory formation. Technical Report CS-86-08, Department of Computer Science, University of Waterloo, Waterloo, Ontario.

Poole, David (1988). A logical framework for default reasoning. *Artificial Intelligence*, 36(1):27–47.

Poole, David (1989). Normality and faults in logic-based diagnosis. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence*. pages 1304–1310.

Poole, David (1991). A theorist to Prolog compiler. Department of Computer Science, University of British Columbia, Vancouver. Manuscript.

Poole, David, Goebel, Randy, and Aleliunas, Romas (1987). Theorist: A logical reasoning system for defaults and diagnosis. In Cercone, Nick and McCalla, Gordon, editors, *The Knowledge Frontier: Essays in the Representation of Knowledge*. Springer-Verlag, New York, pages 331–352. Also published as Research Report CS-86-06, Faculty of Mathematics, University of Waterloo, February, 1986.

Pople, Harry E. (1973). A mechanization of abductive logic. In *Proceedings of the 4th International Joint Conference on Artificial Intelligence*. pages 147–152.

Rao, Anand S. and Foo, Norman Y. (1989). Formal theories of belief revision. In *Proceedings of the First Annual Conference on Principles of Knowledge Representation and Reasoning*. pages 369–380.

Raudaskoski, Pirrko (1990). Repair work in human-computer interaction. In Luff, Paul, Gilbert, Nigel, and Frolich, David, editors, *Computers and Conversation*. Academic Press, pages 151–172.

Reiter, Raymond (1980). A logic for default reasoning. *Artificial Intelligence*, 13:81–132. Reprinted in Ginsberg (1987).

Reiter, Raymond and Criscuolo, Giovanni (1983). Some representational issues in default reasoning. *Journal of Comp. Math. Appl.*, pages 1–13. (Special issue on computational linguistics).

Remler, Jane E. (1978). Some repairs on the notion of repairs in the interest of relevance. In Farkas, Donka, Jacobsen, Wesley M., and Todrys, Karol W., editors, *Papers from the Fourteenth Regional Meeting of the Chicago Linguistic Society*, Chicago, IL. Chicago Linguistic Society, pages 391–502.

Sacerdoti, Earl D. (1977). *A Structure for Plans and Behavior.* American Elsevier, New York.

Sadek, M. D. (1992). A study in the logic of intention. In *Proceedings of the Third International Conference on Principles of Knowledge Representation and Reasoning*, Cambridge, MA. pages 462–473.

Sanford, David L. and Roach, John W. (1987). Representing and using metacommunication to control speaker's relationships in natural-language dialogue. *International Journal of Man–Machine Studies*, 26:301–319.

Schegloff, Emanuel A. (1987). Some sources of misunderstanding in talk-in-interaction. *Linguistics*, 25:201–218.

Schegloff, Emanuel A. (1988). Presequences and indirection: Applying speech act theory to ordinary conversation. *Journal of Pragmatics*, 12:55–62.

Schegloff, Emanuel A. (1992). Repair after next turn: The last structurally provided defense of intersubjectivity in conversation. *American Journal of Sociology*, 97(5):1295–1345.

Schegloff, Emanuel A., Jefferson, Gail, and Sacks, Harvey (1977). The preference for self-correction in the organization of repair in conversation. *Language*, 53:361–382.

Schegloff, Emanuel A. and Sacks, Harvey (1973). Opening up closings. *Semiotica*, 7:289–327.

Schiffrin, Deborah (1987). *Discourse Markers.* Cambridge University Press, Cambridge, England.

Schmidt, C. F., Sridharan, N. S., and Goodson, J. L. (1978). The plan recognition problem: An intersection of psychology and artificial intelligence. *Artificial Intelligence*, 11:45–83.

Searle, John (1969). *Speech Acts.* Cambridge University Press, Cambridge, England.

Shelley, Cameron (1992). Speech acts and pragmatics in sentence generation. Technical Report CS-92-49, University of Waterloo Faculty of Mathematics, Waterloo, Ontario, Canada. M.Math. thesis.

Shimanoff, S. B. (1980). *Communication Rules: Theory and Research.* Sage, Beverly Hills, CA.

Shortliffe, E. H. (1976). *Computer-based Medical Consultations: MYCIN.* Elsevier, New York.

Sidner, Candice L. and Israel, David J. (1981). Recognizing intended meaning and speaker's plans. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence*, Vancouver, British Columbia. pages 203–208.

Soskin, William F. (1963). *Verbal Interactions in a Young Married Couple.* University of Kansas Publications, Lawrence, Kansas.

Stalnaker, Robert C. (1972). Pragmatics. In *Semantics of Natural Language.* D. Reidel Publishing Company, Dordrecht, pages 380–397.

Stickel, M. E. (1989). A Prolog technology theorem prover. *Journal of Automated Reasoning*, 4:353–360.

Terasaki, A. (1976). Pre-announcement sequences in conversation. Social Science Working Paper 99, School of Social Science, University of California, Irvine.

Thomason, Richmond H. (1990). Propagating epistemic coordination through mutual defaults I. In Parikh, Rohit, editor, *Proceedings, Third Conference on Theoretical Aspects of Reasoning about Knowledge (TARK 1990)*, Pacific Grove, CA. pages 29–39.

Tsui, Amy B. M. (1991). Sequencing rules and coherence in discourse. *Journal of Pragmatics*, 15:111–129.

Tulving, Endel (1972). Episodic and semantic memory. In Tulving, Endel and Donaldson, Wayne, editors, *Organization and Memory.* Academic Press, New York.

Umrigar, Zerksis D. and Pitchumani, Vijay (1985). An experiment in programming with full first-order logic. In *Symposium of Logic Programming*, Boston, MA. IEEE Computer Society Press.

van Arragon, Paul (1990). *Nested Default Reasoning for User Modeling.* PhD thesis, Department of Computer Science, University of Waterloo, Waterloo, Ontario. Published by the department as Research Report CS-90-25.

Véronis, Jean (1991). Error in natural language dialogue between man and machine. *International Journal of Man–Machine Studies*, 35:187–217.

Walker, Marilyn A. (1991). Redundancy in collaborative dialogue. In *1991 AAAI Fall Symposium on Discourse Structure in Natural Language Understanding and Generation*, Pacific Grove, Monterey, CA. pages 124–129.

Webber, Bonnie Lynn and Mays, Eric (1983). Varieties of user misconceptions: Detection and correction. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, Karlsruhe. pages 650–652.

Winslett, Marianne (1989). Some updates are circumscription. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, Detroit, MI. pages 859–863.

Zukerman, Ingrid (1991). Avoiding mis-communications in concept explanations. In *Proceedings of the 13th Annual Conference of the Cognitive Science Society*, Chicago, IL. pages 406–411.

Appendix A

The syntax of the language

The language of the model is a sorted-first order language $\mathcal{L}$, comprising a denumerable set of predicates, variables, constants, and functions, along with the boolean connectives $\vee, \wedge, \neg, \supset$, and $\equiv$, and the predicate $=$. The terms of $\mathcal{L}$ come in six sorts: agents, turns, sequences of turns, actions, descriptions, and suppositions. I distinguish two important subtypes of actions: surface-level speech acts and discourse-level speech acts, and two classes of suppositions: simple and compound. The vocabulary of $\mathcal{L}$ includes the following variables, functions, and predicates (I use bold font for function names and italic for predicates):

1. **Variables** An infinite number of variables of each sort; we use s for agents, t for turns, ts for turn sequences, a for actions, d for descriptions, and p for suppositions, possibly with a subscript.

2. **Function Symbols** An infinite number of function symbols of every sort and arity. I also define a number of special ones:

   alt, which takes a turn and a turn sequence and returns a (new) turn.

   The following functions all return a supposition:

   do, which takes an agent and an action as arguments.
   mistake, which takes an agent and two actions as arguments.
   intend, which takes an agent and an supposition as arguments.
   knowif, which takes an agent and an supposition as arguments.
   knowref, which takes an agent and a description as arguments.
   knowsBetterRef, which takes two agents and a description as arguments.
   suspectRef, which takes an agent, a description, and a supposition as arguments.
   not, which takes a simple supposition as an argument and returns a simple supposition of the opposite polarity.[1]

[1] The function **not** is distinct from boolean connective $\neg$ (which, as we shall see, is represented in Theorist as `not`). I use it to capture the supposition expressed by an agent who says something negative, *e.g.,* "I do not want to go."

and, which takes a simple supposition and a (possibly compound) supposition as arguments and returns a compound supposition.

The only functions that return a surface-level speech act are **surface-request, surface-inform, surface-informif**, and **surface-informref**, which all take two agents as their first two arguments, and either a discourse-level act, supposition, or description as their third argument. Similarly, all functions that return a discourse-level speech act take two agents, and possibly an action, supposition, or description as arguments.

3. **Predicate symbols** The following predicates have been defined; below, we group them according to the types of arguments they take:

- Predicates representing various relations among speech acts and suppositions. These include: *decomp, ambiguous, lintention, inconsistentLI, lexpectation, compatible*, and *wouldExpect.*
- The predicate *believeA*, which takes an agent and a supposition as arguments.
- Predicates that associate suppositions with turns, where, for clarity, we distinguish the fact symbols, *expressed* and *expressedNot*, and the default predicate symbols *skepticalAboutE* and *skepticalAboutEN.*
- Predicates that take among their arguments exactly one of type turn sequence, where again we can distinguish the fact symbols:

  utter, try, expected, shouldTry, hasGoal, lintentionsOk, conditionsOk, active, turnOf, root, focus, and *succ*

  and the defaults:

  pickForm, expectedReply, intentionalAct, acceptance, adoptPlan, otherMisunderstanding, selfMisunderstanding, makeFourthTurnRepair, makeThirdTurnRepair, done, lintend, skepticalAboutA, credulousI, and *activationPersists.*
- Predicates that relate two turn sequences, including the fact symbols *refocus, alternativeTS, reconstruction,* and *successorTS*, and the default symbol *reconstructTS.*

Appendix B

The decomposition relation

```
fact decomp(surface-request(X,Y,do(Y,P)), request(X,Y,do(Y,P))).
fact decomp(surface-greeting(X,Y), greeting(X,Y)).

fact decomp(surface-inform(X,Y,P), inform(X,Y,P)).

fact believeA(X,P)
     and lintend(X, P, TS)
     and nullTS(TS)
     => decomp(surface-inform(X,Y,P), informif(X,Y,P)).

fact believeA(X,not P)
     and lintend(X, not P, TS)
     and nullTS(TS)
     => decomp(surface-inform(X,Y, not P), informif(X,Y,P)).

fact decomp(surface-informref(X,Y,P), informif(X,Y,knowref(X,P))).
fact decomp(surface-informref(X,Y,P), informref(X,Y,P)).
fact decomp(surface-informref(X,Y,P), inform(X,Y,P)).

fact decomp(surface-informif(X,Y,P), informif(X,Y,P)).

% 1. Do you know who's going? (Jane, that's who.)
fact decomp(surface-request(X, Y, informif(Y, X, knowref(Y, P))),
            pretell(X, Y, P)).

% 2. Who's going (a plain old request to tell)
fact decomp(surface-request(X, Y, informref(Y, X, P)),
            askref(X, Y, P)).

% 3. I want to know who's going.  (this might also be treated as indirect)
%fact decomp(surface-inform(X,Y,desireref(X,P)), askref(X,Y,P)).
```

```
% 4. Do you know who's going? (then tell me)
fact decomp(surface-request(X, Y, informif(Y, X, knowref(Y, P))),
            askref(X, Y, P)).

% 5. Is Jane going? or Do you know who's going (Y/N)?
fact decomp(surface-request(X, Y, informif(Y, X, P)),
            askif(X, Y, P)).

% 6. Is Jane going? or Do you know whether (if) Jane is going?
fact decomp(surface-request(X, Y, informif(Y, X, knowif(Y, P))),
            askif(X, Y, P)).

% 7. Who's going?
fact decomp(surface-request(X, Y, informref(Y, X, D)),
            testref(X, Y, D)).

% Some defined acts.
fact (decomp(A,inform(X,Y,P)) or decomp(A,pretell(X,Y,P)))
     and believeA(H,danger(H,P))
     => decomp(A,warn(X,Y,P)).

fact decomp(exit, closing(X,Y)).
```

Appendix C

Linguistic intentions

```
%INFORM
fact lintention(inform(S, H, P), (P, intend(S, knowif(H,P)))).

%INFORMREF
fact lintention(informref(S, H, P), (knowref(S, P),
                                     intend(S, knowref(H, P)))).

%INFORMIF
fact lintention(informif(S, H, P), (knowif(S, P),
                                    intend(S, knowif(H, P)))).

%ASKREF
fact lintention(askref(S, H, P), (not knowref(S,P),
                                  intend(S, knowref(S,P)),
                                  intend(S, do(H,informref(H,S,P))))).

%ASKIF
fact lintention(askif(S, H, P), (not knowif(S, P),
                                 intend(S, knowif(S,P)),
                                 intend(S, do(H,informif(H,S,P))))).

%PRETELL
fact lintention(pretell(S,H,P), (knowref(S,P),
                                 knowsBetterRef(S,H,P),
                                 intend(S,do(S,informref(S,H,P))),
                                 intend(S,knowref(H,P)))).

%TESTREF
fact lintention(testref(S, H, P), (knowref(S, P),
                                   intend(S, do(H,informref(H, S, P))))).

%WARN
```

```
fact lintention(warn(S, H, P), (danger(P), intend(S, avoid(H,P)))).

%REQUEST
fact lintention(request(S, H, do(H,P)), intend(S, do(H,P))).
```

Appendix D

Linguistic expectations

```
fact lexpectation(do(S,informref(S,H,D)),
                  not knowsBetterRef(H,S,D),
                  accept(H,D)).

fact lexpectation(do(S,askif(S,H,P)),
                  knowif(H,P),
                  do(H,informif(H,S,P))).

fact lexpectation(do(S,askref(S,H,D)),
                  knowref(H,D),
                  do(H,informref(H,S,D))).

fact lexpectation(do(S,pretell(S,H,D)),
                  knowsBetterRef(S,H,D),
                  do(H,askref(H,S,D))).

fact lexpectation(do(S,testref(S,H,D)),
                  knowref(H,D),
                  do(H,informref(H,S,D))).

fact lexpectation(do(S,testref(S,H,D)),
                  suspectRef(H,D,P),
                  do(H,askif(H,S,P))).

fact lexpectation(do(S,request(S,H,do(H,P))),
                  cando(H,I,do(H,P)),
                  do(H,P)).
```

Appendix E

More output from the program

E.1 The ‘dinner for Alfred’ example

This appendix gives an annotated version of the output for Example 7.1 for one set of background assumptions. Text in typewriter font is the input and output from two runs of the program; one with the system playing speaker A and one with the system playing speaker B (indented one-half inch). Text in normal or italic font is commentary by the author.

To test the example, I used a version of Theorist implemented in Quintus Prolog Release 3.1.1 (Sun-4, SunOS 4.1). The system as speaker A took 16 cpu seconds; as speaker B, it took 22 seconds.

For this test, I assumed that A (`a`) wants B (`b`) to tell her the time of the dinner for Alfred, that she believes that she does not already know, that he knows when it is, and that he believes that she does not know, of any given time (including seven-thirty), whether it is the time of the dinner.

```
fact hasGoal(a, do(b, informref(b, a, whenIsDinner,ts(0)).
fact believe(a, not knowref(a, whenIsDinner)).
fact believe(a, knowref(b,  whenIsDinner)).
fact believe(b, not knowif(a, dinnerAtSevenThirty)).
```

I also assumed that speaker `b`, believes that he does not know the time of the dinner, but suspects it is at seven thirty and believes that `a` does know when it is.

```
fact believe(b, not knowref(b, whenIsDinner)).
fact believe(b, suspectRef(b, whenIsDinner, dinnerAtSevenThirty)).
fact believe(b, knowref(a, whenIsDinner)).
```

First, speaker a generates a request to try to satisfy her goal:
When is the dinner for Alfred?

```
| ?- startDialogue1.

Explaining shouldTry(a,b,A,ts(0)),intentionalAct(a,b,A,ts(0)),decomp(U,A)

Answer:
  shouldTry(a,b,askref(a,b,whenIsDinner),ts(0)),
  intentionalAct(a,b,askref(a,b,whenIsDinner),ts(0)),
  decomp(surface-request(a,b,informref(b,a,whenIsDinner)),
         askref(a,b,whenIsDinner))

took 0.567 sec.

Explanation:
  intentionalAct(a,b,askref(a,b,whenIsDinner),ts(0))
  adoptPlan(a,b,askref(a,b,whenIsDinner),
            informref(b,a,whenIsDinner),ts(0))
  lintend(a,(not knowref(a,whenIsDinner)
             and intend(a,knowref(a,whenIsDinner))
             and intend(a,do(b,informref(b,a,whenIsDinner)))),ts(0))

***Updating Discourse Model***

Interpretation: askref(a,b,whenIsDinner) (turn number 1)
  expressed(do(a,askref(a,b,whenIsDinner)),1)

Linguistic Intentions of askref(a,b,whenIsDinner):
  not knowref(a,whenIsDinner)
  and intend(a,knowref(a,whenIsDinner))
  and intend(a,do(b,informref(b,a,whenIsDinner)))

Suppositions Added:
  expressedNot(knowref(a,whenIsDinner),1)
  expressed(intend(a,knowref(a,whenIsDinner)),1)
  expressed(intend(a,do(b,informref(b,a,whenIsDinner))),1)

Agent a adopted plan to achieve: informref(b,a,whenIsDinner)

***Generating Utterance***

<<<surface-request(a,b,informref(b,a,whenIsDinner))
```

Speaker `b` interprets this utterance as a `test`:

```
| ?- startDialogue2.

      >>>surface-request(a,b,informref(b,a,whenIsDinner))

***Interpreting Utterance***

Explaining utter(a,b,request(a,b,informref(b,a,whenIsDinner)),ts(0))

Is formula pickForm(a,b,surface-request(a,b,informref(b,a,whenIsDinner)),
                    askref(a,b,whenIsDinner),ts(0)) ok (y/n)?n.

Is formula pickForm(a,b,request(a,b,surface-informref(b,a,whenIsDinner)),
                    testref(a,b,whenIsDinner),ts(0)) ok (y/n)?y.

took 0.716 sec.

Explanation:
  intentionalAct(a,b,testref(a,b,whenIsDinner),ts(0))
  adoptPlan(a,b,testref(a,b,whenIsDinner),informref(b,a,whenIsDinner),
            ts(0))
  lintend(a,(knowref(a,whenIsDinner)
             and intend(a,do(b,informref(b,a,whenIsDinner)))),ts(0))
  credulousBA(a,knowref(b,whenIsDinner))
  credulousI(a,ts(0))
  pickForm(a,b,surface-request(a,b,informref(b,a,whenIsDinner)),
           testref(a,b,whenIsDinner),ts(0))

***Updating Discourse Model***

Interpretation: testref(a,b,whenIsDinner) (turn number 1)
  expressed(do(a,testref(a,b,whenIsDinner)),1)

Linguistic Intentions of testref(a,b,whenIsDinner):
  knowref(a,whenIsDinner)
  and intend(a,do(b,informref(b,a,whenIsDinner)))

Suppositions Added:
  expressed(knowref(a,whenIsDinner),1)
  expressed(intend(a,do(b,informref(b,a,whenIsDinner))),1)

Agent a adopted plan to achieve: informref(b,a,whenIsDinner)
```

Speaker `b` now replies: *Is it at seven thirty?*

```
Explaining shouldTry(b,a,A,ts(1)),intentionalAct(b,a,A,ts(1)),
           decomp(U,A)

Answer:
  shouldTry(b,a,askif(b,a,dinnerAtSevenThirty),ts(1)),
  intentionalAct(b,a,askif(b,a,dinnerAtSevenThirty),ts(1)),
  decomp(surface-request(b,a,informif(a,b,dinnerAtSevenThirty)),
         askif(b,a,dinnerAtSevenThirty))

took 1.667 sec.

Explanation:
  intentionalAct(b,a,askif(b,a,dinnerAtSevenThirty),ts(1))
  acceptance(b,askif(b,a,dinnerAtSevenThirty),ts(1))
  lintend(b,(not knowif(b,dinnerAtSevenThirty)
             and intend(b,knowif(b,dinnerAtSevenThirty))
             and intend(b,do(a,informif(a,b,dinnerAtSevenThirty)))),
          ts(1))
  expectedReply(do(a,testref(a,b,whenIsDinner)),
                suspectRef(b,whenIsDinner,dinnerAtSevenThirty),
                do(b,askif(b,a,dinnerAtSevenThirty)),ts(1))

***Updating Discourse Model***

Interpretation: askif(b,a,dinnerAtSevenThirty) (turn number 2)
  expressed(do(b,askif(b,a,dinnerAtSevenThirty)),2)

Linguistic Intentions of askif(b,a,dinnerAtSevenThirty):
  not knowif(b,dinnerAtSevenThirty)
  and intend(b,knowif(b,dinnerAtSevenThirty))
  and intend(b,do(a,informif(a,b,dinnerAtSevenThirty)))

Suppositions Added:
  expressedNot(knowif(b,dinnerAtSevenThirty),2)
  expressed(intend(b,knowif(b,dinnerAtSevenThirty)),2)
  expressed(intend(b,do(a,informif(a,b,dinnerAtSevenThirty))),2)

Agent b performed expected act: askif(b,a,dinnerAtSevenThirty)

***Generating Utterance***

<<<surface-request(b,a,informif(a,b,dinnerAtSevenThirty))
```

Speaker a recognizes that Speaker b has produced a yes-no question, misunderstanding her request as a test:

```
      >>>surface-request(b,a,informif(a,b,dinnerAtSevenThirty))

***Interpreting Utterance***

Explaining utter(b,a,bequest(b,a,informif(a,b,dinnerAtSevenThirty)),ts(1))

Is formula pickForm(b,a,surface-request(b,a,informif(a,b,dinnerAtSevenThirty)),
                    askif(b,a,dinnerAtSevenThirty),ts(1)) ok (y/n)?y.

took 1.650 sec.

Explanation:
  otherMisunderstanding(b,a,mistake(b,askref(a,b,whenIsDinner),
                                    testref(a,b,whenIsDinner)),
                        askif(b,a,dinnerAtSevenThirty),ts(1))
  lintend(b,(not knowif(b,dinnerAtSevenThirty)
             and intend(b,knowif(b,dinnerAtSevenThirty))
             and intend(b,do(a,informif(a,b,dinnerAtSevenThirty)))),ts(1))
  credulousBA(b,suspectRef(b,whenIsDinner,dinnerAtSevenThirty))
  pickForm(b,a,surface-request(b,a,informif(a,b,dinnerAtSevenThirty)),
           askif(b,a,dinnerAtSevenThirty),ts(1))

***Updating Discourse Model***

Interpretation: askif(b,a,dinnerAtSevenThirty) (turn number 2)
  expressed(do(b,askif(b,a,dinnerAtSevenThirty)),2)

Linguistic Intentions of askif(b,a,dinnerAtSevenThirty):
  not knowif(b,dinnerAtSevenThirty)
  and intend(b,knowif(b,dinnerAtSevenThirty))
  and intend(b,do(a,informif(a,b,dinnerAtSevenThirty)))

Suppositions Added:
  expressedNot(knowif(b,dinnerAtSevenThirty),2)
  expressed(intend(b,knowif(b,dinnerAtSevenThirty)),2)
  expressed(intend(b,do(a,informif(a,b,dinnerAtSevenThirty))),2)

Agent b mistook askref(a,b,whenIsDinner) for testref(a,b,whenIsDinner):
  expressed(mistake(b,askref(a,b,whenIsDinner),testref(a,b,whenIsDinner)),2)
```

Speaker a then produces a third-turn repair:

No, I'm *asking* you.

```
Explaining shouldTry(a,b,A,ts(2)),intentionalAct(a,b,A,ts(2)),decomp(U,A)

Answer:
  shouldTry(a,b,inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner)))),ts(2)),
  intentionalAct(a,b,inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner)))),ts(2)),
  decomp(surface-inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner)))),
         inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner)))))

took 4.033 sec.

Explanation:
  intentionalAct(a,b,inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner)))),ts(2))
  lintend(a,(intend(a,do(a,askref(a,b,whenIsDinner)))
             and intend(a,knowif(b,intend(a,do(a,askref(a,b,whenIsDinner)))))),
          ts(2))
  makeThirdTurnRepair(a,b,informref(b,a,whenIsDinner),ts(2))

***Updating Discourse Model***

Interpretation:
      inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner)))) (turn number 3)
  expressed(do(a,inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner))))),3)

Linguistic Intentions of inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner)))):
  intend(a,do(a,askref(a,b,whenIsDinner)))
  and intend(a,knowif(b,intend(a,do(a,askref(a,b,whenIsDinner)))))

Suppositions Added:
  expressed(intend(a,do(a,askref(a,b,whenIsDinner))),3)
  expressed(intend(a,knowif(b,intend(a,do(a,askref(a,b,whenIsDinner))))),3)

m performed third turn repair

***Generating Utterance***

<<<surface-inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner))))
```

Speaker b recognizes his misunderstanding:

```
      >>>surface-inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner))))

***Interpreting Utterance***

Explaining utter(a,b,inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner)))),
                 ts(2))

Is formula
    pickForm(a,b,
          surface-inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner)))),
          inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner)))), ts(2))
ok (y/n)?y.

took 3.533 sec.

Explanation:
  activationPersists(do(a,testref(a,b,whenIsDinner)),2)
  selfMisunderstanding(a,b,mistake(b,askref(a,b,whenIsDinner),
                                   testref(a,b,whenIsDinner)),
                       inform(a,b,
                              intend(a,do(a,askref(a,b,whenIsDinner)))),
                       ts(2))
  lintend(a,(intend(a,do(a,askref(a,b,whenIsDinner))) and
             intend(a,knowif(b,intend(a,do(a,askref(a,b,whenIsDinner)))))),
          ts(2))
  pickForm(a,b,
           surface-inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner)))),
           inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner)))),ts(2))

***Updating Discourse Model***

Interpretation:
     inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner)))) (turn number 3)
  expressed(do(a,inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner))))),3)

Linguistic Intentions of
      inform(a,b,intend(a,do(a,askref(a,b,whenIsDinner)))):
  intend(a,do(a,askref(a,b,whenIsDinner)))
  and intend(a,knowif(b,intend(a,do(a,askref(a,b,whenIsDinner)))))
```

```
Suppositions Added:
  expressed(intend(a,do(a,askref(a,b,whenIsDinner))),3)
  expressed(intend(a,
            knowif(b,intend(a,do(a,askref(a,b,whenIsDinner))))), 3)

Agent b mistook askref(a,b,whenIsDinner) for testref(a,b,whenIsDinner):
  expressed(mistake(b,askref(a,b,whenIsDinner),testref(a,b,whenIsDinner)),3)
```

Speaker b produces a fourth-turn repair: *Oh, I don't know.*

```
Explaining shouldTry(b,a,A,ts(3)),intentionalAct(b,a,A,ts(3)),decomp(U,A)

***Reconstructing Turn Number 1***

  expressed(do(a,askref(a,b,whenIsDinner)),alt(1))
  expressedNot(knowref(a,whenIsDinner),alt(1))
  expressed(intend(a,knowref(a,whenIsDinner)),alt(1))
  expressed(intend(a,do(b,informref(b,a,whenIsDinner))),alt(1))

Answer:
 shouldTry(b,a,inform(b,a,not knowref(b,whenIsDinner)),ts(3)),
 intentionalAct(b,a,inform(b,a,not knowref(b,whenIsDinner)),ts(3)),
 decomp(surface-inform(b,a,not knowref(b,whenIsDinner)),
        inform(b,a,not knowref(b,whenIsDinner)))

took 16.050 sec.

Explanation:
  intentionalAct(b,a,inform(b,a,not knowref(b,whenIsDinner)),ts(3))
  makeFourthTurnRepair(b,a,inform(b,a,not knowref(b,whenIsDinner)),
                       ts(3),ts(alt(1)))
  lintend(b,(not knowref(b,whenIsDinner)
             and intend(b,knowif(a,not knowref(b,whenIsDinner)))),
          ts(alt(1)))
  expectedReply(do(a,askref(a,b,whenIsDinner)),
                not knowref(b,whenIsDinner),
                do(b,inform(b,a,not knowref(b,whenIsDinner))),
                ts(alt(1)))
  reconstructTS(1,ts(3),ts(alt(1)))
```

```
***Updating Discourse Model***

Interpretation: inform(b,a,not knowref(b,whenIsDinner)) (turn number 4)
  expressed(do(b,inform(b,a,not knowref(b,whenIsDinner))),4)

Linguistic Intentions of inform(b,a,not knowref(b,whenIsDinner)):
  not knowref(b,whenIsDinner)
  and intend(b,knowif(a,not knowref(b,whenIsDinner)))

Suppositions Added:
  expressedNot(knowref(b,whenIsDinner),4)
  expressed(intend(b,knowif(a,not knowref(b,whenIsDinner))),4)

r performed fourth turn repair

***Generating Utterance***

<<<surface-inform(b,a,not knowref(b,whenIsDinner))
```

Speaker a takes this utterance as an acceptance of her initial request.

```
      >>>surface-inform(b,a,not knowref(b,whenIsDinner))

***Interpreting Utterance***

Explaining utter(b,a,inform(b,a,not knowref(b,whenIsDinner)),ts(3))

Is formula pickForm(b,a,surface-inform(b,a,not knowref(b,whenIsDinner)),
                    inform(b,a,not knowref(b,whenIsDinner)),ts(3)) ok (y/n)?y.

took 9.434 sec.

Explanation:
  activationPersists(do(a,askref(a,b,whenIsDinner)),3)
  activationPersists(do(a,askref(a,b,whenIsDinner)),2)
  intentionalAct(b,a,inform(b,a,not knowref(b,whenIsDinner)),ts(3))
  acceptance(b,inform(b,a,not knowref(b,whenIsDinner)),ts(3))
  lintend(b,(not knowref(b,whenIsDinner)
            and intend(b,knowif(a,not knowref(b,whenIsDinner)))),ts(3))
  expectedReply(do(a,askref(a,b,whenIsDinner)),
                not knowref(b,whenIsDinner),
                do(b,inform(b,a,not knowref(b,whenIsDinner))),ts(3))
  credulousBA(b,not knowref(b,whenIsDinner))
  pickForm(b,a,surface-inform(b,a,not knowref(b,whenIsDinner)),
           inform(b,a,not knowref(b,whenIsDinner)),ts(3))
```

```
***Updating Discourse Model***

Interpretation: inform(b,a,not knowref(b,whenIsDinner)) (turn number 4)
  expressed(do(b,inform(b,a,not knowref(b,whenIsDinner))),4)

Linguistic Intentions of inform(b,a,not knowref(b,whenIsDinner)):
  not knowref(b,whenIsDinner)
  and intend(b,knowif(a,not knowref(b,whenIsDinner)))

Suppositions Added:
  expressedNot(knowref(b,whenIsDinner),4)
  expressed(intend(b,knowif(a,not knowref(b,whenIsDinner))),4)

Agent b performed expected act: inform(b,a,not knowref(b,whenIsDinner))
```

E.2 The 'Susan wants nails' example

The typographical conventions of the preceding examples apply. The system as Susan took took 13 seconds; the system as Bridget took 15.

We begin by assuming that Susan (`s`) wants Bridget (`b`) to tell her if she has any nails, that Susan believes that Bridget will know if she has any nails, that she herself does not want the fingernail type of nail, and that she knows if she wants that type or not.

```
fact hasGoal(s, do(b, informif(b, s, have(b, nails))), ts(0)).
fact believe(s, knowif(b, have(b, nails))).
fact believe(s, not typeWanted(nails, fingerNails)).
fact believe(s, knowif(s, typeWanted(nails, fingerNails))).
```

We assume that after the first turn the second agent, Bridget, wants Susan to tell her if the type of nails she wants are fingernails and she believes that Susan knows if she wants this type or not.

```
fact believe(b,knowif(s,typeWanted(nails,fingerNails))).
fact hasGoal(b, do(s,informif(s,b,typeWanted(nails,fingerNails))), ts(1)).
```

We also assume that Bridget will believe that she knows if she has nails or not once she has added the supposition `active(not typeWanted(nails, fingernails))` to her model of the discourse.[1]

[1]This assumption is necessary because the model does not yet include an account of belief revision.

First, Susan generates a request to try to satisfy her goal:
You don't have any nails, do you?

```
| ?- startDialogue1.

Explaining shouldTry(s,b,A,(0)),intentionalAct(s,b,A,(0)),decomp(U,A)

Answer:
  shouldTry(s,b,askif(s,b,have(b,nails)),ts(0)),
  intentionalAct(s,b,askif(s,b,have(b,nails)),ts(0)),
  decomp(surface-request(s,b,informif(b,s,have(b,nails))),
         askif(s,b,have(b,nails)))

took 0.650 sec.

Explanation:
  intentionalAct(s,b,askif(s,b,have(b,nails)),ts(0))
  adoptPlan(s,b,askif(s,b,have(b,nails)),informif(b,s,have(b,nails)),ts(0))
  lintend(s,(not knowif(s,have(b,nails))
              and intend(s,knowif(s,have(b,nails)))
              and intend(s,do(b,informif(b,s,have(b,nails))))),ts(0))

***Updating Discourse Model***

Interpretation: askif(s,b,have(b,nails)) (turn number 1)
  expressed(do(s,askif(s,b,have(b,nails))),1)

Linguistic Intentions of askif(s,b,have(b,nails)):
  not knowif(s,have(b,nails))
  and intend(s,knowif(s,have(b,nails)))
  and intend(s,do(b,informif(b,s,have(b,nails))))

Suppositions Added:
  expressedNot(knowif(s,have(b,nails)),1)
  expressed(intend(s,knowif(s,have(b,nails))),1)
  expressed(intend(s,do(b,informif(b,s,have(b,nails)))),1)

Agent s adopted plan to achieve: informif(b,s,have(b,nails))

***Generating Utterance***

<<<surface-request(s,b,informif(b,s,have(b,nails)))
```

Bridget interprets this utterance as an `askif`:

```
| ?- startDialogue2.

      >>>surface-request(s,b,informif(b,s,have(b,nails)))

***Interpreting Utterance***

Explaining utter(s,b,request(s,b,informif(b,s,have(b,nails))),ts(0))

Is formula pickForm(s,b,surface-request(s,b,informif(b,s,have(b,nails))),
                    askif(s,b,have(b,nails)),ts(0)) ok (y/n)?y.

took 0.700 sec.

Explanation:
  intentionalAct(s,b,askif(s,b,have(b,nails)),ts(0))
  adoptPlan(s,b,askif(s,b,have(b,nails)),
            informif(b,s,have(b,nails)),ts(0))
  lintend(s,(not knowif(s,have(b,nails))
             and intend(s,knowif(s,have(b,nails)))
             and intend(s,do(b,informif(b,s,have(b,nails))))),ts(0))
  credulousBA(s,knowif(b,have(b,nails)))
  credulousI(s,ts(0))
  pickForm(s,b,surface-request(s,b,informif(b,s,have(b,nails))),
           askif(s,b,have(b,nails)),ts(0))
```

```
***Updating Discourse Model***

Interpretation: askif(s,b,have(b,nails)) (turn number 1)
  expressed(do(s,askif(s,b,have(b,nails))),1)

Linguistic Intentions of askif(s,b,have(b,nails)):
  not knowif(s,have(b,nails))
  and intend(s,knowif(s,have(b,nails)))
  and intend(s,do(b,informif(b,s,have(b,nails))))

Suppositions Added:
  expressedNot(knowif(s,have(b,nails)),1)
  expressed(intend(s,knowif(s,have(b,nails))),1)
  expressed(intend(s,do(b,informif(b,s,have(b,nails)))),1)

Agent s adopted plan to achieve: informif(b,s,have(b,nails))
```

Bridget, who is not sure what type of nails Susan needs, decides to ask:
Fingernails?

```
Explaining shouldTry(b,s,A,ts(1)),intentionalAct(b,s,A,ts(1)),decomp(U,A)

Answer:
  shouldTry(b,s,askif(b,s,typeWanted(nails,fingerNails)),ts(1)),
  intentionalAct(b,s,askif(b,s,typeWanted(nails,fingerNails)),ts(1)),
  decomp(surface-request(b,s,
                       informif(s,b,typeWanted(nails,fingerNails))),
         askif(b,s,typeWanted(nails,fingerNails)))

took 1.317 sec.

Explanation:
  intentionalAct(b,s,askif(b,s,typeWanted(nails,fingerNails)),ts(1))
  adoptPlan(b,s,askif(b,s,typeWanted(nails,fingerNails)),
            informif(s,b,typeWanted(nails,fingerNails)),ts(1))
  lintend(b,(not knowif(b,typeWanted(nails,fingerNails))
             and intend(b,knowif(b,typeWanted(nails,fingerNails)))
             and intend(b,
                   do(s,informif(s,b,typeWanted(nails,fingerNails))))),
          ts(1))
```

```
***Updating Discourse Model***

Interpretation: askif(b,s,typeWanted(nails,fingerNails)) (turn number 2)
  expressed(do(b,askif(b,s,typeWanted(nails,fingerNails))),2)

Linguistic Intentions of askif(b,s,typeWanted(nails,fingerNails)):
  not knowif(b,typeWanted(nails,fingerNails))
  and intend(b,knowif(b,typeWanted(nails,fingerNails)))
  and intend(b,do(s,informif(s,b,typeWanted(nails,fingerNails))))

Suppositions Added:
  expressedNot(knowif(b,typeWanted(nails,fingerNails)),2)
  expressed(intend(b,knowif(b,typeWanted(nails,fingerNails))),2)
  expressed(intend(b,
            do(s,informif(s,b,typeWanted(nails,fingerNails)))),2)

Agent b adopted plan to achieve:
  informif(s,b,typeWanted(nails,fingerNails))

***Generating Utterance***

<<<surface-request(b,s,informif(s,b,typeWanted(nails,fingerNails)))
```

Susan recognizes that although she had expected an answer, Bridget first needs to know more about the type of nails she wants.

```
     >>>surface-request(b,s,informif(s,b,typeWanted(nails,fingerNails)))

***Interpreting Utterance***

Explaining utter(b,s,request(b,s,informif(s,b,typeWanted(nails,fingerNails))),ts(1))

Is formula
     pickForm(b,s,
         surface-request(b,s,informif(s,b,typeWanted(nails,fingerNails))),
         askif(b,s,typeWanted(nails,fingerNails)),ts(1)) ok (y/n)?y.

took 1.100 sec.

Explanation:
  intentionalAct(b,s,askif(b,s,typeWanted(nails,fingerNails)),ts(1))
  adoptPlan(b,s,askif(b,s,typeWanted(nails,fingerNails)),
            informif(s,b,typeWanted(nails,fingerNails)),ts(1))
  lintend(b,(not knowif(b,typeWanted(nails,fingerNails))
             and intend(b,knowif(b,typeWanted(nails,fingerNails)))
```

```
            and intend(b,do(s,informif(s,b,typeWanted(nails,fingerNails))))),
         ts(1))
  credulousBA(b,knowif(s,typeWanted(nails,fingerNails)))
  credulousI(b,ts(1))
  pickForm(b,s,surface-request(b,s,informif(s,b,typeWanted(nails,fingerNails))),
          askif(b,s,typeWanted(nails,fingerNails)),ts(1))


***Updating Discourse Model***

Interpretation: askif(b,s,typeWanted(nails,fingerNails)) (turn number 2)
  expressed(do(b,askif(b,s,typeWanted(nails,fingerNails))),2)

Linguistic Intentions of askif(b,s,typeWanted(nails,fingerNails)):
  not knowif(b,typeWanted(nails,fingerNails))
  and intend(b,knowif(b,typeWanted(nails,fingerNails)))
  and intend(b,do(s,informif(s,b,typeWanted(nails,fingerNails))))

Suppositions Added:
  expressedNot(knowif(b,typeWanted(nails,fingerNails)),2)
  expressed(intend(b,knowif(b,typeWanted(nails,fingerNails))),2)
  expressed(intend(b,do(s,informif(s,b,typeWanted(nails,fingerNails)))),2)

Agent b adopted plan to achieve: informif(s,b,typeWanted(nails,fingerNails))
```

Susan responds by telling Bridget that she does not need fingernails:[2]
No, nails to nail into the wall. When I get bored here I'm going to go put up those pictures.

```
Explaining shouldTry(s,b,A,ts(2)),intentionalAct(s,b,A,ts(2)),decomp(U,A)

Answer :
  shouldTry(s,b,informif(s,b,typeWanted(nails,fingerNails)),ts(2)),
  intentionalAct(s,b,informif(s,b,typeWanted(nails,fingerNails)),ts(2)),
  decomp(surface-inform(s,b,not typeWanted(nails,fingerNails)),
         informif(s,b,typeWanted(nails,fingerNails)))

took 2.433 sec.

Explanation:
  lintend(s,not typeWanted(nails,fingerNails),ts(0))
  intentionalAct(s,b,informif(s,b,typeWanted(nails,fingerNails)),ts(2))
  acceptance(s,informif(s,b,typeWanted(nails,fingerNails)),ts(2))
```

[2]She also tells Bridget the type of nails she does need and why she needs them, but we do not yet handle more than one speech act per turn.

```
  lintend(s,(knowif(s,typeWanted(nails,fingerNails))
             and intend(s,knowif(b,typeWanted(nails,fingerNails)))),
          ts(2))
  expectedReply(do(b,askif(b,s,typeWanted(nails,fingerNails))),
                knowif(s,typeWanted(nails,fingerNails)),
                do(s,informif(s,b,typeWanted(nails,fingerNails))),ts(2))


***Updating Discourse Model***

Interpretation: informif(s,b,typeWanted(nails,fingerNails)) (turn number 3)
  expressed(do(s,informif(s,b,typeWanted(nails,fingerNails))),3)

Linguistic Intentions of informif(s,b,typeWanted(nails,fingerNails)):
  not typeWanted(nails,fingerNails)

Suppositions Added:
  expressedNot(typeWanted(nails,fingerNails),3)

Agent s performed expected act: informif(s,b,typeWanted(nails,fingerNails))

***Generating Utterance***

<<<surface-inform(s,b,not typeWanted(nails,fingerNails))
```

Bridget takes this utterance as evidence Susan has understood her question.

```
      >>>surface-inform(s,b,not typeWanted(nails,fingerNails))

***Interpreting Utterance***

Explaining utter(s,b,inform(s,b,not typeWanted(nails,fingerNails)),ts(2))

Is formula
    pickForm(s,b,
             surface-inform(s,b,not typeWanted(nails,fingerNails)),
             inform(s,b,not typeWanted(nails,fingerNails)), ts(2)) o
ok (y/n)?n.

Is formula
    pickForm(s,b,
             surface-inform(s,b,not typeWanted(nails,fingerNails)),
             informif(s,b,not typeWanted(nails,fingerNails)),ts(2))
ok (y/n)?y.

took 4.400 sec.

Explanation:
  lintend(s,not typeWanted(nails,fingerNails),ts(0))
  intentionalAct(s,b,informif(s,b,typeWanted(nails,fingerNails)),ts(2))
  acceptance(s,informif(s,b,typeWanted(nails,fingerNails)),ts(2))
  lintend(s,(knowif(s,typeWanted(nails,fingerNails))
             and intend(s,knowif(b,typeWanted(nails,fingerNails)))),
          ts(2))
  expectedReply(do(b,askif(b,s,typeWanted(nails,fingerNails))),
                knowif(s,typeWanted(nails,fingerNails)),
                do(s,informif(s,b,typeWanted(nails,fingerNails))),ts(2))
  credulousBA(s,knowif(s,typeWanted(nails,fingerNails)))
  pickForm(s,b,surface-inform(s,b,not typeWanted(nails,fingerNails)),
           informif(s,b,typeWanted(nails,fingerNails)),ts(2))
  credulousBA(s,not typeWanted(nails,fingerNails))
```

```
***Updating Discourse Model***

Interpretation: informif(s,b,typeWanted(nails,fingerNails)) (turn number 3)
  expressed(do(s,informif(s,b,typeWanted(nails,fingerNails))),3)

Linguistic Intentions of informif(s,b,typeWanted(nails,fingerNails)):
  not typeWanted(nails,fingerNails)

Suppositions Added:
  expressedNot(typeWanted(nails,fingerNails),3)

Agent s performed expected act: informif(s,b,typeWanted(nails,fingerNails))
```

Bridget now answers Susan's original question:
No.

```
Explaining shouldTry(b,s,A,ts(3)),intentionalAct(b,s,A,ts(3)),decomp(U,A)

Answer:
  shouldTry(b,s,informif(b,s,have(b,nails)),ts(3)),
  intentionalAct(b,s,informif(b,s,have(b,nails)),ts(3)),
  decomp(surface-informif(b,s,have(b,nails)),
         informif(b,s,have(b,nails)))

took 8.283 sec.

Explanation:
  expectedReply(do(s,askif(s,b,have(b,nails))),
                E1,do(b,informif(b,s,have(b,nails))),ts(3))
  activationPersists(do(s,askif(s,b,have(b,nails))),3)
  activationPersists(do(s,askif(s,b,have(b,nails))),2)
  intentionalAct(b,s,informif(b,s,have(b,nails)),ts(3))
  acceptance(b,informif(b,s,have(b,nails)),ts(3))
  lintend(b,(knowif(b,have(b,nails))
             and intend(b,knowif(s,have(b,nails)))),ts(3))
```

```
***Updating Discourse Model***

Interpretation: informif(b,s,have(b,nails)) (turn number 4)
  expressed(do(b,informif(b,s,have(b,nails))),4)

Linguistic Intentions of informif(b,s,have(b,nails)):
  knowif(b,have(b,nails))
  and intend(b,knowif(s,have(b,nails)))

Suppositions Added:
  expressed(knowif(b,have(b,nails)),4)
  expressed(intend(b,knowif(s,have(b,nails))),4)

Agent b performed expected act: informif(b,s,have(b,nails))

***Generating Utterance***

<<<surface-informif(b,s,have(b,nails))
```

Susan takes this utterance as evidence that Bridget has understood and answered her request.

```
      >>>surface-informif(b,s,have(b,nails))

***Interpreting Utterance***

Explaining utter(b,s,informif(b,s,have(b,nails)),ts(3))

Is formula pickForm(b,s,surface-informif(b,s,have(b,nails)),
                    informif(b,s,have(b,nails)),ts(3)) ok (y/n)?y.

took 9.217 sec.

Explanation:
  activationPersists(do(s,askif(s,b,have(b,nails))),3)
  activationPersists(do(s,askif(s,b,have(b,nails))),2)
  intentionalAct(b,s,informif(b,s,have(b,nails)),ts(3))
  acceptance(b,informif(b,s,have(b,nails)),ts(3))
  lintend(b,(knowif(b,have(b,nails))
             and intend(b,knowif(s,have(b,nails)))),ts(3))
  expectedReply(do(s,askif(s,b,have(b,nails))),
                knowif(b,have(b,nails)),
                do(b,informif(b,s,have(b,nails))),ts(3))
  credulousBA(b,knowif(b,have(b,nails)))
  pickForm(b,s,surface-informif(b,s,have(b,nails)),
          informif(b,s,have(b,nails)),ts(3))
```

```
***Updating Discourse Model***

Interpretation: informif(b,s,have(b,nails)) (turn number 4)
  expressed(do(b,informif(b,s,have(b,nails))),4)

Linguistic Intentions of informif(b,s,have(b,nails)):
  knowif(b,have(b,nails)) and intend(b,knowif(s,have(b,nails)))

Suppositions Added:
  expressed(knowif(b,have(b,nails)),4)
  expressed(intend(b,knowif(s,have(b,nails))),4)

Agent b performed expected act: informif(b,s,have(b,nails))
```

Appendix F

How Prioritized Theorist works

The system presented here has been implemented using Poole's (1991) compiler for Theorist, modified to incorporate priorities. The compiler takes input facts and defaults and translates them into Prolog clauses that implement the model elimination proof procedure, modified to add default assumptions to the proof tree where necessary and possible. The great quantity of these clauses makes it difficult to determine the overall structure of the program they implement.[1] Thus, I will not attempt to explain them here. Instead, we shall consider a meta-level interpreter for Theorist. An interpreter reveals the overall structure of Theorist more clearly, because it is more compact. In a meta-level interpreter, instead of building explanation predicates for each input predicate, the interpreter uses a single meta-level explanation predicate that treats the input predicates as a parameter. This approach is slightly less efficient than compiling the predicates beforehand, because one must repeatedly test for the existence of a relevant fact or default. One must also test for the contrapositives for each assertion, unless they are already in the knowledge base.

Figure F.1) gives VanArragon's (1990) interpreter for the propositional case. It assumes that all contrapositives have been precomputed by the knowledge engineer.

In the figure, we use the following conventions for variables:

- `G`, a literal or conjunction of literals, is the goal to be explained.
- `NG` is the negation of `G`.
- `N`, `M`, and `P` are priority levels where `N` is the priority level of defaults that may be used to explain `G`; `M` is the priority level we are trying to prove; and `P` is the priority level of an arbitrary default.
- `A` is a list of literals corresponding to all the subgoals that occur in the proof tree that leads to the current subgoal.

[1]For example, translating a knowledge base that contains only the following Theorist formulae:

```
default (1, birdsfly(X)): flies(X) <- bird(X).
fact emu(X) => bird(X).
fact emu(X) => not flies(X).
```

generates more than 100 Prolog clauses.

```
ex((G1,G2),N,A,D1,D3)     :- ex(G1,N,A,D1,D2),    ex(G2,N,A,D2,D3).
ex(true,N,A,D,D)          :- true.
ex(G,N,A,D1,D2)           :- f((G<-B)),           ex(B,N,[G|A],D1,D2).
ex(G,N,A,D,D)             :- neg(G,NG),           mem(NG,A).
ex(G,N,A,D,D)             :- mem([G,M],D).
ex(G,N,A,D1,[[G,M]|D2])   :- d(M,G),              M =< N, neg(G,NG),
                          \+ pr(NG,[],D1),        enlarge(NG,M,D1,D2).

pr((G1,G2),A,D)           :- pr(G1,A,D),          pr(G2,A,D).
pr(true,A,D)              :- true.
pr(G,A,D)                 :- f((G<-B)),           pr(B,[G|A],D).
pr(G,A,D)                 :- neg(G,NG),           mem(NG,A).
pr(G,A,D)                 :- mem([G,M],D).

enlarge(NG,M,D1,D3)       :-   block(Status,NG,M,D1,D2),
                             unblock(Status,NG,M,D1,D2,D3).

block(blocked,NG,M,D1,D2):- M >= 2, M1 is M-1,  bl(NG,M1,[],D1,D2), !.
block(okay, NG,M,D1,D1)   :- true.

unblock(okay, NG,M,D1,D2,D1)     :- true.
unblock(blocked,NG,M,D1,D2,D4)   :- mem([X,P],D2), \+ mem([X,P],D1),
                                    neg(X,NX), ex(NX,P,[],D1,D3),
                                    enlarge(NG,M,D3,D4).

bl((G1,G2),N,A,D1,D3)     :- bl(G1,N,A,D1,D2),    bl(G2,N,A,D2,D3).
bl(true,N,A,D,D)          :- true.
bl(G,N,A,D1,D2)           :- f((G<-B)),           bl(B,N,[G|A],D1,D2).
bl(G,N,A,D,D)             :- neg(G,NG),           mem(NG,A).
bl(G,N,A,D,D)             :- mem([G,M],D).
bl(G,N,A,D1,[[G,M]|D1])   :- d(M,G),              M =< N, neg(G,NG),
                          \+ pr(NG,[],D1).

mem(G,[G|_])              :- true.
mem(G,[_|L])              :- mem(G,L).
neg(not(X),X)             :- !.
neg(X,not(X))             :- true.
```

Figure F.1: Van Arragon's (1990) interpreter for Prioritized Theorist

- D is a list of atoms corresponding to defaults.

The predicate `f((G <- B))` is true if `G <- B` is a fact[2]; `d(M,G)` is true if `G` has been input as a default with priority level `M`. The interpreter itself uses the following eight predicates:

1. `ex(G,N,A,D1,D2)` is true if the goal `G` with priority `N` can be explained given facts `A` (the *ancestor* list) and defaults `D2`, where the defaults `D1` have been assumed already. More formally,

   $\mathcal{F} \cup \mathtt{D2} \models \bigwedge_{a \in \mathtt{A}} \supset \mathtt{G}$

2. `pr(G,A,D)` is true if the goal `G` can be proven given the facts `A` and defaults `D`.

3. `enlarge(NG,M,D1,D2)` is true if `D1` is enlarged to become `D2` so that `NG` cannot be explained with `D2` at level `M`−`1`.

4. `block(Status,NG,M,D1,D2)` is true if:

   - `Status` is "`blocked`" and `NG` can be explained with `D2` (including `D1`) at level `M`−`1`.
   - `Status` is "`okay`" and `NG` cannot be explained with `D1` at level `M`−`1`.

5. `unblock(Status,NG,M,D1,D2,D3)` is true if `Status` is "`okay`" or if `Status` is "`blocked`" (meaning that `NG` is blocked by `D2`) and we can add to `D1` to get `D3` so `NG` cannot be explained at level `M`.

6. `bl` is like `ex` without the priority constraint.

7. `mem(G,L)` is true if `G` is in list `L`.

8. `neg(G,NG)` is true if `NG` is the negation of `G`.

Looking at the predicate `ex` in detail, in the figure, the third clause for `ex` adds `G` to the ancestor list; the fourth allows `G` to be explained if its negation is already a member of the ancestor list; the fifth allows `G` to be explained if it has been assumed previously; and the sixth clause attempts to use a default to explain `G`, relying on negation as failure to check the consistency of the resulting set of defaults and on the predicate `enlarge` to verify that no priority constraints have been violated.

As stated in Chapter 4, the process of testing whether a default δ of priority m satisfies the priority constraints, given a set of accumulated assumptions D, involves the following steps:

1. Using D, try to explain $\neg\delta$ with defaults of priority $m-1$. If $\neg\delta$ cannot be explained then D is large enough and δ satisfies the priority constraints.

[2] The interpreter assumes that for each input fact, all of its contrapositives are also facts. One of the functions of Poole's compiler is to generate these contrapositives automatically.

2. If $\neg\delta$ can be explained, δ has been *blocked*; however, if to explain $\neg\delta$ one had to make some new assumptions δ_p of priority p, try to *unblock* δ by explaining $\neg\delta_p$ using defaults of priority higher (*i.e.,* value lower) than p. (If no new assumptions were required then δ cannot be explained.)

3. If unblocking is successful, then add the new assumptions required to block δ_p to D to form D' and return to step 1 to verify that $\neg\delta$ cannot be explained with D'; if unsuccessful then δ cannot be explained.

This process terminates when either $\neg\delta$ cannot be explained (step 1), no new assumptions were needed in order to explain $\neg\delta$ (step 2), or δ_p could not be explained (step 3).