A Process Model for Recognizing Communicative Acts and Modeling Negotiation Subdialogues

Sandra Carberry*
University of Delaware

Lynn Lambert†
Christopher Newport University

Negotiation is an important part of task-oriented expert-consultation dialogues. This paper presents a plan-based model for understanding cooperative negotiation subdialogues. Our system infers both the communicative actions that people pursue when speaking and the beliefs underlying these actions. Beliefs, and the strength of these beliefs, are recognized from the surface form of utterances, from discourse acts, and from the explicit and implicit acceptance of previous utterances. Our algorithm for recognizing discourse actions combines linguistic, world, and contextual knowledge in a unified framework. By combining these different knowledge sources, we are able to recognize complex discourse acts such as expressing doubt, to identify the relationship of utterances to one another, and to model negotiation subdialogues. Since negotiation is an integral part of multi-agent activity, our process model addresses an important aspect of cooperative interaction and thus is a step toward an intelligent and robust natural language consultation system.

1 Introduction

In a typical expert-consultation dialogue, one participant (hereafter referred to as the executing agent or EA) has a goal that he[1] wants to achieve and is working with the other participant (referred to as the consulting agent or CA) to construct a plan for achieving this goal. Although both the plan construction process and the conversation are collaborative activities this does not mean that people always believe what they are told. In fact, part of the collaborative activity of conversation is negotiation of conflicting beliefs. This negotiation is particularly important in task-oriented expert-consultation dialogues, since the participants must resolve any conflicting beliefs in order to work together effectively to devise a plan that is both well-formed and addresses the executing agent's needs. Thus a robust natural language consultation system must be able to handle negotiation subdialogues.

Even though there is wide agreement that negotiation is an integral part of multi-agent activity, previous natural language understanding systems have been unable to handle negotiation subdialogues such as the following:

* Department of Computer Science, Newark, DE 19716, USA
† Department of Physics, Computer Science, and Engineering, Newport News, VA 23606, USA
1 For exposition purposes, we will use the masculine gender when referring to EA and the feminine gender when referring to CA.

(1)	S1:	*Who is teaching CS360?*
(2)	S2:	*Dr. Smith is teaching CS360.*
(3)	S1:	*But isn't CS360 an undergraduate course?*
(4)	S2:	*Yes. CS360 is an undergraduate course.*
(5)		*Dr. Smith teaches both graduate and undergraduate courses.*
(6)	S1:	*Who handles the CS360 lab?*

For example, existing systems do not recognize when an agent is expressing doubt at a previous response as in utterance (3), when an agent is attempting to resolve a conflict suggested by the other participant as in utterances (4)-(5), or when an agent is implicitly conveying acceptance of a communicated proposition as in utterance (6). These shortcomings prevent existing natural language systems from being able to handle dialogues in which one agent initially does not accept the proposition conveyed by the other agent and initiates a negotiation subdialogue to resolve their differences in belief.

We have developed a plan-based model of dialogue that addresses these limitations. Our analysis of naturally occurring dialogue indicates that one way that people express doubt at a proposition P_{doubt} is by contending that some other conflicting proposition P_i is true. Our process model includes an algorithm for recognizing such expressions of doubt, as well as other complex discourse acts. The algorithm uses a multi-strength belief model and a combination of linguistic, world, and contextual knowledge. Our implemented system can recognize implicit as well as explicit acceptance of a communicated proposition, multiple expressions of doubt at the same proposition, expressions of doubt at both immediately preceding and earlier utterances, and negotiation subdialogues embedded within other negotiation subdialogues.

In the remainder of this paper, we describe our system and how this process is performed. Section 2 describes the kinds of expressions of doubt found in our corpus analysis, and Section 3 discusses the factors that must be taken into account in recognizing the kind of expression of doubt that we have been studying. Section 4 presents our process model for recognizing complex discourse acts (such as expressions of doubt) and assimilating them into the dialogue context. First it discusses why it is necessary to capture varying degrees of belief, describes the multi-strength model of belief used in our system, and discusses how our description of actions avoids assuming that a speaker will automatically adopt a communicated proposition. Then it introduces the notion of an action that requires evidence for its recognition and presents our recognition algorithm that uses a combination of linguistic, world, and contextual knowledge. Section 5 steps through an extended example that illustrates our system's ability to recognize complex discourse acts and model negotiation subdialogues. Section 6 discusses the evaluation of our system and our plans for future work, and Section 7 discusses related research. The examples in this paper are taken from a university advisement domain, since this is the domain in which we have implemented our system.

2 Motivation from Naturally Occurring Dialogues

To identify how speakers express doubt, we analyzed a corpus of naturally occurring dialogues in the domains of financial planning, university courses, real estate, pets, taxes, and travel. The real estate, pet, and financial planning(Transcripts, 1982) dialogues were transcribed from radio talk shows, the tax and travel(Transcripts, 1992) dialogues were transcribed from tapes of simulated interactions, and the university course dialogues(Transcripts, 1985) were transcribed from student advisement sessions. In the corpus we found instances in which a speaker expressed doubt at a proposition by contending that some other conflicting proposition was true.[2] In addition, we extracted other examples of such expressions of doubt from the dialogues in novels. These kinds of expressions of doubt can be realized as surface negative questions or tag questions and are often accompanied by the cue word *but*, as in the following example taken from the Harry Gross financial planning dialogues(Transcripts, 1982) in which S2's last utterance expresses doubt at S1's recommendation:

S1: *I would like to see that into an individual retirement account rollover in a mutual fund group.*
S2: *At my age?*
S1: *Yes.*
S2: *Uh, yeah but isn't there any risk?*

However, our corpus analysis also provided instances where surface negative and tag questions were used to seek verification, such as in the following excerpt from the set of financial planning dialogues:

S1: *And if you have more money left after you pay the taxes what difference does it make if you pay a few bucks more in taxes.*
S2: *I'm telling my wife but she won't listen.*
S1: *Well maybe she'll listen to me. If you get 200 bucks – isn't it better to have 200 bucks and uh have 200 left than to have nothing at all?*

Our recognition algorithm has only been concerned with recognizing instances in which a speaker expresses doubt by contending that some other proposition is true. However, in our corpus speakers also expressed doubt in the following ways, and our future work will include extending our system to handle these:[3]

- Drawing attention to an inconsistent feature or proposition: The speaker brings into focus a feature or proposition that is already part of the dialogue context but which is intended to discredit the proposition being doubted. These utterances were often realized as an elliptical fragment and included *"And you're how old?"*, *"Even though it's four more years?"*, and *"At my age?"*.

- Drawing attention to violated expectations: The speaker mentions an expectation that is inconsistent with the doubted proposition. An example of this from our corpus is *"You're kidding, what happened to the seventy-eight dollar fares or those sort of things?"*

- Repetition: The speaker queries the doubted proposition; this usually took the form of a declarative followed by a question mark, as in *"You have 40 thou(sand) in a mm fund?"*

- Explicit statements and questions: The speaker explicitly doubts what has been said or asks for justification; examples from our corpus include *"I'm not so sure of that."* and *"Who ever said that?"*

2 We found a very few instances in which the speaker asked the hearer if he was sure the conflicting proposition wasn't true; an example from our corpus is *"Are you sure he didn't name himself as attorney for the estate?"* taken from the Harry Gross financial planning dialogues(Transcripts, 1982). Our current system does not handle such expressions of doubt.

3 Walker(Walker, 1996) analyzed the Harry Gross financial planning dialogues(Transcripts, 1982) to identify features that distinguish acceptance from rejection. However, she did not consider expressions of doubt and some of her rejections would fall into our express doubt category.

(7)	EA:	*What is Dr. Smith teaching?*
(8)	CA:	*Dr. Smith is teaching Architecture.*
(9)	EA:	*Isn't Dr. Brown teaching Architecture?*

Figure 1
A Dialogue with an Expression of Doubt

- Cue words: The speaker uses discourse markers to convey his doubt. In addition to the cue word *but* that is often used to realize expressions of doubts in the other categories, cue words such as *Really?* and *What?* were used by themselves to express doubt and cue words such as *even though* were used to convey doubt in utterances such as *"Even though it's four more years?"*

3 Recognizing Expressions of Doubt

When a listener in a collaborative interaction does not accept the proposition that the speaker is trying to convey, a negotiation subdialogue ensues in which the participants attempt to *square-away* (Joshi, 1982) their disparate beliefs. Negotiation subdialogues often involve complex discourse acts which implicitly refer to some proposition that is part of the existing dialogue context. We have found that such complex discourse acts require evidence for their recognition. Since the motivation for our work is the recognition of communicative acts that occur in negotiation subdialogues, this section examines in detail how plausibility and evidence affect the recognition of one kind of complex discourse act, an expression of doubt.

For a listener to recognize a discourse action, it must be plausible that the speaker holds the requisite beliefs for performing the action. (A belief is *plausible* if the available evidence does not refute it.) For example, in order for a listener to interpret an utterance as felicitously asking a question in order to obtain information, it must be plausible that the speaker does not already know the information and that the speaker believes that the listener may be able to provide it. Similarly, to interpret an utterance as expressing doubt at a proposition P_{doubt} by contending P_i, it must be plausible that the speaker holds certain beliefs. Consider a university setting in which each course has only one instructor and a speaker uses the proposition P_i, that Dr. Brown teaches Architecture, to express doubt at the proposition P_{doubt}, that Dr. Smith is teaching Architecture, as in utterance (9) of Figure 1. In order to intend (9) as an expression of doubt in a collaborative dialogue,

the speaker must believe 1) that the hearer has some belief in P_{doubt}
2) that P_i is true
3) that if P_i is true, then P_{doubt} is not

The hearer must be able to plausibly ascribe each of these beliefs to the speaker in recognizing the expression of doubt. First, it must be plausible that the speaker believes that the hearer has some belief in the proposition that is being doubted, since it is pointless in a collaborative dialogue to express doubt[4] at something about which there is no disagreement. It must also be plausible that the speaker has some belief in P_i, since otherwise the hearer could not be expected to believe that the speaker was using

4 We are using *express doubt* in the sense of challenging the truth of a proposition.

a conflict between P_i and P_{doubt} to question the validity of P_{doubt}. Similarly, it must be plausible that the speaker believes that if P_i is true, then P_{doubt} is not; otherwise the hearer could not be expected to think that the speaker believes that the truth of P_i raises doubts about P_{doubt}.

Although being able to ascribe beliefs as plausible is necessary for recognition of all discourse actions, some discourse actions, such as the expressions of doubt that we consider in this paper, require further evidence. This evidence is provided by **linguistic**, **world**, and **contextual knowledge**. These knowledge sources can either provide evidence for a generic discourse act (such as an expression of doubt) or evidence that the conditions are satisfied for performing a specific discourse act (such as expressing doubt that Dr. Smith is teaching Architecture). In addition, contextual knowledge can suggest a particular interpretation when equivalent evidence exists for several specific discourse acts. These knowledge sources are discussed in the next sections.

3.1 Linguistic Knowledge

3.1.1 Evidence for a Generic Discourse Act A number of researchers(Reichman, 1978; Reichman, 1985; Grosz and Sidner, 1986; Polanyi, 1986; Cohen, 1987; Hirschberg and Litman, 1987; Litman and Allen, 1987; Schiffrin, 1987; Hinkelman, 1989; Litman and Hirschberg, 1990; Knott and Dale, 1994; Knott and Mellish, 1996; Marcu, 1997) have investigated the use in discourse of special words and phrases such as *but*, *anyway*, and *by the way*. They found that these clue words, or discourse markers, have a number of different functions, including indicating the role of an utterance in the dialogue, conveying the relationship between utterances, suggesting shifts in focus of attention, conveying the structure of the discourse, etc.

Consider again the dialogue shown in Figure 1. If EA had followed (7)-(8) with (9a)

(9a) EA: *Isn't Architecture one of our required courses?*

then EA's utterance would not be interpreted as expressing doubt but would instead be understood as merely seeking information about the Architecture course. However, if this utterance is preceded by the clue word *but*, as in (9b) below,

(9b) EA: *But isn't Architecture one of our required courses?*

then the utterance is expressing doubt, though we have difficulty ascertaining the reason for this doubt – perhaps EA believes that Dr. Smith does not teach courses that students are required to take!

Thus, clue words comprise one source of evidence in the recognition of discourse acts. In particular, a clue word can provide evidence for a generic discourse act, such as *Express-Doubt*, but it remains for other sources to resolve what is being doubted.

3.1.2 Evidence for a Specific Discourse Act Expressions of doubt do not always include clue words, as illustrated by utterance (9) in Figure 1. In the absence of a clue word, we need evidence that the speaker holds the three beliefs, listed on page 4, for performing a specific discourse act. Evidence for the second belief (that the speaker believes that P_i is true) is often provided by the surface form of the utterance, such as an utterance of the form *"Isn't P_i?"* – for example, *"Isn't Dr. Brown teaching Architecture?"* in (9). This surface form indicates a strong belief in the queried proposition while a simple yes-no question, such as *"Is Dr. Brown teaching Architecture?"*, does not. Therefore if EA were to follow (7)-(8) with *"Is Dr. Brown teaching Architecture?"*, EA would seem to have a misconception that more than one person can teach a course or perhaps be seeking information in order to subsequently express doubt – but the utterance itself is not conveying doubt that Dr. Smith is teaching Architecture. Thus the surface form of

(10) EA: *Who is teaching CS360 (a systems course)?*
(11) CA: *Dr. Smith is teaching CS360.*
(12) EA: *Isn't Dr. Smith on sabbatical?*
(13) CA: *No, Dr. Smith is not on sabbatical.*

Figure 2
A Dialogue with Two Open Propositions

the utterance is one source of evidence that the speaker holds the requisite beliefs for performing a specific discourse act.

3.2 World Knowledge

World knowledge in the form of stereotypical beliefs is another source of evidence that the speaker holds the requisite beliefs for a particular discourse act. For example, world knowledge can provide evidence for the third speaker belief listed on page 4. Suppose that it is stereotypically believed that prestigious fellowships are awarded for sabbaticals, that faculty on sabbatical do not teach, and that faculty only teach in their area of expertise. Consider the dialogue shown in Figure 2. After (13), there are two propositions that have been conveyed by CA but not yet completely accepted by EA: the proposition that Dr. Smith is not on sabbatical and the proposition that Dr. Smith is teaching CS360, communicated by utterances (13) and (11) respectively. A subsequent utterance might express doubt at one of these propositions or might forego the opportunity to doubt them, perhaps by pursuing some discourse act unrelated to either of the propositions. Consider the following three possible continuations of the dialogue:

(14a) EA: *Wasn't Dr. Smith awarded a Fulbright?*
(14b) EA: *Isn't Dr. Smith a theory person?*
(14c) EA: *Isn't Dr. Smith an excellent teacher?*

While (14a) and (14b) seem to be expressing doubt, (14c) is simply seeking further information about Dr. Smith. The reason for this difference in interpretation is that in the case of (14a) and (14b), evidence from world knowledge suggests that EA believes that P_i (the proposition that EA contends is true) implies that one of the two open propositions is false, whereas no such evidence exists in the case of (14c). In the case of (14a), since it is stereotypically believed that prestigious fellowships are awarded for sabbaticals, EA's utterance should be interpreted as expressing doubt at the proposition that Dr. Smith is not on sabbatical. In the case of (14b), since Dr. Smith being a theory person is an alternative to Dr. Smith being a systems person and it is stereotypically believed that being a systems person is necessary for teaching CS360 (a systems course), EA's utterance would instead be interpreted as expressing doubt at the proposition that Dr. Smith is teaching CS360. Thus, world knowledge in the form of stereotypical beliefs is another source of evidence that the speaker holds the requisite beliefs for performing a particular discourse act.

If EA had uttered (14c), EA's utterance would be interpreted as merely seeking new information since there is no domain knowledge suggesting that EA believes that Dr. Smith being an excellent teacher contributes to determining whether Dr. Smith is on sabbatical or to identifying the instructor of CS360. Note that (14c) demonstrates why plausibility alone is insufficient for recognition. Although there is no evidence that EA believes that Dr. Smith being an excellent teacher implies that Dr. Smith is on sabbatical or that Dr. Smith is not teaching CS360, there is also no evidence to the contrary, and thus it is plausible that EA believes that Dr. Smith being an excellent teacher indicates that he is on sabbatical or that he is not teaching CS360. This is not sufficient, however, to interpret (14c) as an expression of doubt.

3.3 Contextual Knowledge

An agent can infer from a dialogue many of the beliefs of the other participant. These acquired beliefs about the other participant's beliefs form one kind of contextual knowledge that can be used as evidence for the beliefs given on page 4. In addition, contextual knowledge determines the **salience** (or degree of prominence) of propositions at the current point in the dialogue, and salience is a factor that constrains the interpretation of coherent discourse actions. Consider the first three utterances in the dialogue shown in Figure 2. EA's acceptance of CA's telling of the proposition that Dr. Smith is teaching CS360 establishes the mutual belief that CA believes that Dr. Smith is teaching CS360[5] and thus provides evidence for the first belief given on page 4; in addition, the proposition that Dr. Smith is teaching CS360 becomes salient and is added to the dialogue context. Thus while an utterance such as (12a)

(12a) EA: *Doesn't Dr. Smith usually teach theory courses?*

might be used following (11) to express doubt at the statement that Dr. Smith is teaching CS360, it cannot be used following (11) to express doubt at the proposition that Dr. Smith teaches CS410 because 1) there is no reason for EA to believe that CA has any belief in the proposition that Dr. Smith teaches CS410 and 2) the proposition that Dr. Smith teaches CS410 is not salient at this point in the dialogue.

In addition, contextual knowledge plays two other roles in the recognition of discourse acts. First, in the case of expressions of doubt, contextual knowledge distinguishes propositions that have not yet been accepted by the speaker and thus are open for rejection. Consider again the dialogue in Figure 2. After (13), there are two propositions that have not yet been accepted by EA and are thus open for rejection by EA. If EA were to continue with (14b), repeated below,

(14b) EA: *Isn't Dr. Smith a theory person?*

then EA would again be expressing doubt at the proposition that Dr. Smith is teaching CS360 and would have implicitly conveyed acceptance of the proposition that Dr. Smith is not on sabbatical. Thus, as the conversation continues, only one proposition would remain open for rejection: the proposition that Dr. Smith is teaching CS360. This claim is supported by a combination of 1) the stack paradigm(Polanyi, 1986; Reichman, 1978; Grosz and Sidner, 1986; Litman and Allen, 1987) which treats topic structure as following a stack-like discipline, 2) focusing heuristics(McKeown, 1983) that suggest that if a speaker has more to say about a topic, then he should do so before moving back to a topic deeper on the stack, and 3) the notion of implicit acceptance (discussed in Section 4.6) that argues that passing up the opportunity to reject an assertion in a collaborative dialogue communicates acceptance of it.

Second, contextual knowledge orders propositions according to their relative salience in the current dialogue. This salience can be used to arbitrate among discourse acts for which there is equivalent evidence. Consider again the dialogue in Figure 2 and suppose that EA had continued with (14d), repeated below:

(14d) EA: *But isn't Dr. Smith an excellent teacher?*

5 Note that here EA is only accepting CA's felicitous *telling* of the proposition, but EA is not adopting the proposition as one of his own beliefs.

Here we have a clue word suggesting an expression of doubt, but the speaker could be expressing doubt either that Dr. Smith is not on sabbatical or that Dr. Smith is teaching CS360. In both cases, we lack evidence for the third speaker belief on page 4. Contextual knowledge suggests that, all other things being equal, the proposition being doubted is the proposition that Dr. Smith is not on sabbatical, since it is the most salient proposition that is open for rejection at this point in the dialogue. Thus, contextual knowledge arbitrates when equivalent evidence is available for several specific discourse acts.

3.4 Summary

In addition to the requisite speaker beliefs being plausible and the constraints on the discourse act being satisfied (such as the constraint that a proposition be salient at the current point in the dialogue), certain discourse acts require additional evidence for their recognition. Two kinds of evidence that may be used in recognizing discourse actions are 1) evidence (such as a clue word) for a generic discourse act, and 2) evidence that a speaker holds the requisite beliefs for performing a particular discourse act. Evidence for these beliefs can come from linguistic, world, or contextual knowledge. Although we have illustrated each of these knowledge sources by showing how they might provide evidence for one of the requisite beliefs for expressing doubt, it should be noted that each knowledge source might also be used as evidence for other beliefs required for expressing doubt or for beliefs for other discourse acts. For example, although it does not generally arise in the kind of interactive dialogues that we are studying, world knowledge in the form of stereotypical beliefs might be used as evidence that a speaker believes that a hearer has some belief in the doubted proposition P_{doubt}.

4 The Process Model

Grosz and Sidner (Grosz and Sidner, 1986) claim that a robust model of understanding must use multiple knowledge sources in order to recognize the complex relationships that utterances have to one another. We have developed an algorithm that combines linguistic, world, and contextual knowledge, such as that identified in Section 3, in order to recognize complex discourse acts, including one kind of expression of doubt. Linguistic knowledge consists of clue words and the surface form of the utterance; world knowledge includes a set of stereotypical beliefs that users generally hold and recipes for performing discourse acts; and contextual knowledge consists of a model of the user's beliefs acquired from the preceding dialogue, the current structure of the discourse, the existing focus of attention (that aspect of the task on which the participants' attention is currently centered), and the relative salience (degree of prominence) of propositions in the discourse.

The remainder of this section presents the core ideas of our algorithm. Section 4.1 shows why it is necessary to capture varying degrees of belief in a proposition and presents the multi-strength belief model used in our system. Section 4.2 describes our representation of recipes for actions and shows how our recipe for an *Inform* action refrains from *assuming* that the listener will adopt the communicated proposition as part of his own beliefs; it also presents a recipe for expressing doubt and shows how constraints on the speaker's beliefs are captured in the recipe's applicability conditions. Section 4.3 gives an overview of our dialogue model. Section 4.4 describes how chaining is used to hypothesize a sequence of higher-level discourse acts that a speaker may be performing; Section 4.5 introduces the notion of a discourse action that requires evidence for its recognition; Section 4.6 discusses how our model recognizes implicit acceptance of a discourse act; and Section 4.7 presents our recognition algorithm that uses a combination of linguistic, world, and contextual knowledge in recognizing discourse acts.

4.1 A Multi-Strength Belief Model

4.1.1 The Need for Capturing Varying Degrees of Belief As argued in Section 3, if a speaker is expressing doubt at a proposition P_{doubt} by contending some other proposition P_i, then the speaker must have some belief in P_i. Evidence for this belief is often provided by the surface form of the speaker's utterance, such as an utterance of the form *"Isn't P_i?"* – for example, *"Isn't Dr. Brown teaching Architecture?"* But if we treat such an utterance as conveying certain belief that P_i is true, then we cannot handle situations in which an utterance such as this is merely requesting verification since a speaker cannot felicitously seek verification of a proposition that he already knows is true. Therefore, since modeling only ignorance and certainty is inadequate for recognizing complex discourse acts, it is necessary to model the strength of an agent's beliefs.

4.1.2 Representing Varying Degrees of Belief We use a **multi-strength model of belief**, which captures not only ignorance and certainty about the truth of a proposition but also several degrees of belief in between. Utterances of the form *"Isn't P_i?"* are treated as conveying a strong (but uncertain) belief in P_i. In this way, our system is able to handle instances in which an utterance of the form *"Isn't P_i?"* is used to request verification (since the utterance is viewed as conveying uncertainty about P_i) as well as instances in which it is used to express doubt (since the utterance is viewed as conveying some belief in P_i).

Our multi-strength belief model maintains three degrees of belief: certain belief (a belief strength of C); strong but uncertain belief, as in *"Isn't Dr. Brown teaching Architecture?"* (a belief strength of S); and a weak belief, as in *"I think that Dr. Cayne might be an education instructor"* (a belief strength of W). Three degrees of disbelief (indicated by attaching a subscript of N, such as S_N to represent strong disbelief and W_N to represent weak disbelief) are also maintained, and one degree indicating no belief about a proposition (a belief strength of 0). We adopted three degrees of positive and negative belief in our model because that was the minimum number of degrees required for modeling the beliefs communicated in the dialogues that we examined and in the negotiation subdialogues that our system is intended to handle.

Although an agent has some specific strength of belief in a proposition, the other agent may not always know precisely what that strength of belief is but may be able to bound it — for example, he may be able to say that the first agent has *some* belief in a proposition. Our belief model uses belief intervals to capture this, where a belief interval specifies the range of strengths within which an agent's beliefs are thought to fall.

Allen (Allen and Perrault, 1980) noted the need to represent an agent's wanting to know the referent of a term in a proposition, without having to specify what that referent was. For example, if EA asks CA *"Who is teaching CS360?"*, we cannot represent CA's belief that EA wants to know the teacher of CS360 as

believe(CA, want(EA, believe(EA, Teaches(Dr.Smith, CS360))))

since this representation says that CA believes that EA wants to believe that the teacher is Dr. Smith (but EA, in asking the question, may not be predisposed to any such belief and may in fact reject *"Dr. Smith"* as the answer to the question). Allen addressed this with *knowref* and *knowif* predicates, which represented an agent's knowing the referent of a term in a proposition and knowing whether a proposition is true. Thus CA's belief that EA wants to know the teacher of CS360 in the above example might be represented as

believe(CA, want(EA, knowref(EA, _fac, Teaches(_fac, CS360))))

In our multi-strength belief model, *knowref* is treated as being certain about the referent of the term in the specified proposition and *knowif* is treated as being certain about whether a proposition is true or false.

As the dialogue progresses, the belief model must capture the changing beliefs of the user. When a discourse act has been successful, its goals can be used to update the belief model. For example, if the user explicitly accepts the proposition conveyed by the utterance *"Dr. Smith is teaching Architecture"* (perhaps by saying *"Yes, I'll accept that"*), then the system can update its belief model to include the belief that the user himself believes that Dr. Smith is teaching Architecture. However, explicit acceptance is less common than implicit acceptance; Section 4.6 discusses implicit acceptance and its recognition.

Since we do not currently have a response generation component, our system processes the utterances of both participants, alternating between playing the role of CA and the role of EA. Note that this differs from playing the role of a third-party observer — when the system plays the role of EA, the system has access to EA's beliefs (including EA's beliefs about the current dialogue model and EA's beliefs about CA), and when the system plays the role of CA, it has access to CA's beliefs. However, whenever the system assumes the role of a participant and processes a new utterance, it is assumed that this participant has correctly interpreted previous utterances and has a correct model of the preceding dialogue.

4.1.3 Related Work on Modeling Belief Young (Young, 1987) built a model in which the beliefs of the user are part of an explicit, missing, or stereotype module (he used the system's beliefs as a stereotype). Although this system provides needed differentiation among beliefs which the system knows that the user holds, those that the system has attributed to the user, and those about which the system has no knowledge, this model still does not contain degrees of partial belief which are essential for modeling discourse acts such as expressing doubt. Balim and Wilks(Ballim and Wilks, 1991) developed a nested belief model that captures an agent's beliefs about other agents' beliefs. Their system combines belief ascription based on stereotypes with belief ascription based on perturbations of the system's own beliefs, but they do not represent how strongly a belief is held.

Galliers (Galliers, 1991; Galliers, 1992) has specified a non-numeric theory of belief revision which relates strength of belief to persistence of belief. She points out that a belief model for communication must contain a multi-strength model of beliefs that can be modified as the conversation proceeds. She uses endorsements (Cohen, 1985) in an assumption-based truth maintenance system (ATMS (DeKleer, 1986)) to specify a system that orders beliefs according to how strongly they are held. This ordering is used to calculate which beliefs should be revised when beliefs are challenged in the course of conversation.

Walker (Walker, 1991; Walker, 1992) has examined dialogues in which people repeat what they already know either in question or statement form (e.g., *"I have four children." "OK. Four children."*). Walker claims that this repetition by the second speaker is given so that the first speaker realizes that her utterance was understood and believed. That is, cooperative listeners often provide some evidence to speakers to indicate that the listener believes the speaker's claims. Like Galliers, Walker has based the strength of belief on the amount and kind of evidence available for that belief. Cohen and Levesque (Cohen and Levesque, 1991a) also found this kind of corroboration. Our work has not investigated the belief reasoning process or how much evidence is needed for an agent to have a particular amount of confidence in a belief. Instead, it has been concerned with taking into account different communicated strengths of belief and the impact that the different belief strengths have on the recognition of discourse acts.

Our multi-strength belief model is very simple and is only intended to meet our system's need for representing how strongly an agent holds a particular belief. We recently became aware[6] of work by Driankov on a logic in which belief/disbelief pairs capture how strongly a proposition is believed(Driankov, 1988; Bonarini, Cappelletti, and Corrao, 1990). This work appears to be the only formally defined and well-developed logic that models strength of belief. With the exception that Driankov's logic does not include a state of *weak belief*, it appears to provide the representational and reasoning capability needed by our system and we intend to investigate it for future use.

4.2 Discourse Recipes

In previous work, we noted the need to differentiate among domain, problem-solving, and discourse actions (Lambert and Carberry, 1991; Elzer, 1995). In task-oriented consultation dialogues, the participants are constructing a plan for achieving some domain goal, such as owning a home, and the resultant plan will consist of **domain actions** such as applying for a mortgage. In order to construct the domain plan, the participants pursue **problem-solving actions** such as evaluating alternative domain actions or correcting an action in the partially constructed domain plan. Domain and problem-solving actions have been investigated by many researchers (Allen and Perrault, 1980; Perrault and Allen, 1980; Wilensky, 1981; Litman and Allen, 1987; van Beek and Cohen, 1986; Ramshaw, 1989; Carberry, 1990).

Discourse actions are communicative actions that are executed by the dialogue participants in order to obtain or convey the information needed to pursue the problem-solving actions necessary for constructing the domain plan. Examples of very different discourse actions include answering a question, informing, and expressing doubt. Although our system models domain, problem-solving, and discourse actions, this paper is only concerned with recognizing discourse acts, particularly complex discourse acts such as expressing doubt.

Our system's knowledge about how to perform actions is contained in a library of discourse, problem-solving, and domain recipes (Pollack, 1990). Our representation of a recipe includes a header giving the action defined by the recipe, the recipe type, preconditions, applicability conditions, constraints, a body, effects, and a goal. The recipe type is primitive, specialization, or decomposition. If the recipe type is primitive, then the body of the recipe is empty and the header action corresponds with a primitive action in the domain. In a specialization recipe, the body gives a set of alternative ways of performing the header action(Pollack, 1990; Kautz, 1990). For example, one might earn credit in a course either by taking the course for credit or getting credit by exam. In a decomposition recipe, the body gives a set of subactions which constitute performing the header action. A # preceding a subaction in the body of a decomposition recipe indicates that the subaction can be performed any number of times (including zero).[7] Constraints limit the allowable instantiation of variables in each component of a recipe (Litman and Allen, 1987). For example, a variable might have a constraint requiring that it be instantiated with a proposition that is salient at the current point in the discourse. Which instantiations of variables will satisfy the constraints is part of the shared knowledge of the participants.[8]

6 We would like to thank one of the anonymous reviewers and Ingrid Zukerman for bringing this work to our attention.

7 In this work, we are interested in understanding, not generating, responses. However, a generation system would pursue an action preceded by # if its applicability conditions are satisfied and the system does not believe that the action's goal will be satisfied if the action is omitted. The belief reasoning techniques described in (Cawsey et al., 1992) can be used in modeling this.

8 In this research, we have assumed that the participants have equivalent knowledge of language and

Applicability conditions(Carberry, 1987) are conditions that must be satisfied in order for a recipe to be reasonable to pursue in a given situation. The applicability conditions of our discourse recipes capture attitudes (beliefs and wants) that the agent of the action must hold in order for it to be felicitous (Searle, 1970). Applicability conditions differ from preconditions in that one can plan to satisfy preconditions but it is generally anomalous to try and satisfy applicability conditions. For example, in order for _agent1 to inform _agent2 of _proposition, _agent1 must believe that _proposition is true and must not believe that _agent2 already believes _proposition. It would be anomalous for _agent1 to try and adopt the proposition as one of his beliefs solely for the sake of being able to inform someone else of it, and similarly it would be anomalous for _agent1 to get _agent2 to disbelieve a proposition so that _agent1 can subsequently inform him of it.[9]

Belief intervals are used in the applicability conditions to specify the range of strengths that an agent's beliefs may assume. Intervals such as $[\mathtt{b}_i:\mathtt{b}_j]$ specify a strength of belief between $\mathtt{b}_i$ and $\mathtt{b}_j$ inclusive. For example, Figure 3 displays the recipes for the *Inform* and *Tell* discourse acts. The goal of the *Inform* action, `believe(_agent2, _proposition, [C:C])`, is that `_agent2` be certain that `_proposition` is true. On the other hand, `believe(_agent2, _proposition,` $[\mathtt{C}_N:\mathtt{S}]$`)` means that `_agent2` is not convinced that `_proposition` is true (i.e., `_agent2` could have any belief ranging from being certain that `_proposition` is false to having a strong belief that it is true). Thus the applicability condition `believe(_agent1, believe(_agent2, _proposition,` $[\mathtt{C}_N:\mathtt{S}]$`),[0:C])` of the *Inform* act in Figure 3 means that `_agent1` must either be ignorant about `_agent2`'s belief in `_proposition` or have some belief (possibly certain) that `_agent2` is not already certain that `_proposition` is true – i.e., `_agent1` does not believe that `_agent2` is already convinced that `_proposition` is true. In determining the belief strengths specified in the applicability conditions, we examined the beliefs that an agent must hold and tried to identify the minimum and maximum strength of belief that would make the discourse act reasonable to pursue.

We have divided the effects of an action into two subclasses: 1) the results of correctly performing the action, which are labelled **effects**, and 2) the desired effects of the action (over which the agent may lack control), which are labelled **goals**. For example, in the case of domain actions, the *effect* of applying for graduate study is that one has applied, while the *goal* is that one be accepted for graduate study. This distinction between effects and goals is particularly important in the case of discourse actions, where the agent cannot be assured that an action will have its intended result.

Variables in recipes are represented as lowercase strings preceded by an underscore, with the string reflecting the variable's type; for example, _course1 and _course2 refer to variables of type *course*.

In Allen's seminal model of plan recognition(Allen, 1979), the bodies of operators could contain either goals to be achieved or action names with parameters. In our system, preconditions are represented as goals to be achieved while the bodies of recipes specify actions. Since the recipe for each action in our recipe library contains a single goal in its goal field, this suffices since the goal makes clear the purpose of the action in a plan.

maintain equivalent discourse models, and we have not addressed the problem of recognizing miscommunication.

9 How applicability conditions on discourse acts are checked during planning is an interesting question that requires further research. Consider an agent who wants to determine whether a proposition is true; the agent might accomplish this by asking another agent about the proposition. An applicability condition on asking another agent is that the speaker wants to know the other agent's belief about the proposition. But when did this *want* come into existence? It certainly must be satisfied at the time the question is asked, but instead of being part of the initial state, it appears to result from the speaker's decision about how to obtain the desired information.

Discourse Recipe

Action: Inform(_agent1, _agent2, _proposition)
{*_agent1 informs _agent2 of _proposition*}
Recipe-Type: Decomposition
Appl Cond: believe(_agent1, _proposition, [C:C])
believe(_agent1, believe(_agent2, _proposition, [C_N:S]), [0:C])
Body: Tell(_agent1, _agent2, _proposition)
#Address-Believability(_agent1, _agent2, _proposition)
Effects: believe(_agent2, want(_agent1, believe(_agent2, _proposition, [C:C])), [C:C])
Goal: believe(_agent2, _proposition, [C:C])

Discourse Recipe

Action: Tell(_agent1, _agent2, _proposition)
{*_agent1 tells _agent2 of _proposition*}
Recipe-Type: Decomposition
Appl Cond: believe(_agent1, _proposition, [C:C])
Body: Surface-Say-Prop(_agent1, _agent2, _proposition)
#Address-Understanding(_agent1, _agent2, _proposition)
Effects: told-about(_agent1, _agent2, _proposition)
Goal: believe(_agent2, believe(_agent1, _proposition, [C:C]), [C:C])

Figure 3
Recipes for Inform and Tell Discourse Acts

However, in a richer domain where an action could be used to achieve several different goals,[10] it would be necessary to specify the intended goal in the recipe body and chain from it to the desired action in order to capture the motivation for performing the action. During plan recognition, our system matches goals against preconditions of other actions, and it matches actions against the subactions in the bodies of the recipes for other actions. The Effects field is not used for chaining during plan recognition; however, as discussed in Section 4.6.2, the effects and goals in discourse recipes are used for updating a model of the user's beliefs.

4.2.1 Formulation of Discourse Recipes The bodies of our discourse recipes are based on work by other researchers(Allen and Perrault, 1980; Searle, 1970; Cohen and Perrault, 1979), dialogues in which we have participated, the naturally occurring dialogues that we examined, and our hypotheses about how our system might be expanded in the future. For example, consider the discourse recipes for *Obtain-Info-Ref*, *Ask-Ref*, and *Answer-Ref* shown in the Appendix. To obtain information about a proposition via dialogue, _agent1 must both ask another agent about the proposition (*Ask-Ref*) and the second agent must provide the requested information (*Answer-Ref*); this is typical of naturally occurring dialogue and is captured in the body of our *Obtain-Info-Ref* discourse act. The applicability conditions of the *Obtain-Info-Ref* act, such as the condition that _agent1 believe that _agent2 knows the information, are based on criteria identified in (Searle, 1970; Cohen and Perrault, 1979; Allen and Perrault, 1980). The body of the *Ask-Ref* action consists of making the request itself and making the request acceptable; this is because in our own interactions we have encountered situations in which an agent will make a request and then justify it to the listener. The applicability conditions of *Ask-Ref* refer to _agent2's beliefs about the proposition; for example, one applicability condition is that _agent1 wants to know the term that _agent2 believes will satisfy a proposition. We contend that this captures what a speaker wants in asking a listener about a proposition (ie., the speaker wants to know the listener's beliefs about the proposition), and this formulation also allows the *Ask-Ref* to be used as a subaction of a *Test-Knowledge* act in a tutoring system.[11] Note that the fact that a speaker who is seeking information really wants to know correct information about a proposition was captured in the applicability conditions of the *Obtain-Info-Ref* discourse act.

Some of our discourse recipes, such as *Obtain-Info-Ref*, include subactions by both the initiating agent as well as the other participant. This captures the intention of the initiating agent to perform his required subactions as well as the intention that the other agent follow through on her role in the plan for this action — thus, once the second agent recognizes that the initiating agent wants to obtain information, the second agent will also recognize that the initiating agent intends for her to play the role of providing that information. While an agent can construct a plan that includes acts by another agent, the planning agent cannot guarantee the other agent's behavior and thus such discourse plans can fail. (See (Chu-Carroll and Carberry, 1994) for research on dialogues in which agents do not always follow through on their intended role yet still fulfill their collaborative responsibilities.)

A different approach would be to maintain such knowledge about adjacency pairs (Schegloff and Sachs, 1973) and expected continuations in a transition network separate from the discourse recipes, as was done by (Reithinger and Maier, 1995). The advantage of this approach is that fewer discourse recipes are needed and continuations are generalized.

10 For example, one might read a book to gain knowledge or to entertain oneself.
11 Searle(Searle, 1970) notes that there are two kinds of questions, ones whose objective is to obtain knowledge and ones whose objective is to test another's knowledge.

Such a representation would enable us to remove the actions that address acceptance from our discourse recipes but still capture the expectations for them in the transition network. However, the disadvantage of this approach is that the higher-level discourse act would no longer constrain the possible continuations. For example, an *Evaluate-Answer* discourse act is an expected follow-up to an *Answer-Ref* when they are part of a higher-level *Test-Knowledge* discourse act but not when the *Answer-Ref* is part of an *Obtain-Info-Ref* discourse act. Further research is needed to identify the best mechanism for capturing the requisite discourse knowledge.

4.2.2 The Inform Discourse Recipe As noted by Grosz and Sidner(Grosz and Sidner, 1990), the assumption that one participant will slavishly respond to the wishes of the other participant does not reflect collaborative interaction. In Cohen's formulation of speech act operators (Cohen and Perrault, 1979), the effect of an inform was that the hearer believed that the speaker believed the proposition. He postulated a convince act that would cause the hearer to believe the proposition, but this act was left undeveloped and its definition did not allow for the participants to negotiate their beliefs. The effect of an inform act in Allen's system (Allen and Perrault, 1980) is that the hearer believes the communicated proposition — this definition would seem to say that the hearer always accepts the information provided by the speaker.[12] Although Allen's model was only concerned with recognizing the intention to perform an inform act, using his formulation to model negotiation dialogues (where inform actions may not automatically accomplish their purpose) is problematic. In Perrault's persistence model of belief(Perrault, 1990), the hearer adopts a communicated proposition unless he has evidence to the contrary, in which case his original belief persists. Thus, models such as Allen's cannot account for a hearer who does not accept a communicated proposition and Perrault's model cannot account for a hearer who changes his beliefs about a proposition after negotiation.

We want to overcome these limitations and be able to handle negotiation subdialogues in which participants attempt to come to some agreement about their disparate beliefs. Thus the body of the recipe for our *Inform* act (Figure 3) contains two subactions: one in which the speaker tells the hearer a proposition and a second in which the participants address the believability of the communicated proposition and try to come to agreement. In addition, as discussed in the preceding section, the effects of our discourse recipes are often different from the goals. Although this does not solve the problem of recognizing perlocutionary effects, it does allow us to capture the notion that one can, for example, perform an *Inform* act without the hearer adopting the communicated proposition. Thus the *goal* of a discourse recipe is a desired perlocutionary effect (an effect that the speaker wishes the action to have, e.g., `believe(hearer, P, [C:C])` in the case of an *Inform* action), and the *effects* of a discourse recipe are the illocutionary effects (that is, the effect that the speech act has when it is performed and recognized by the hearer, e.g., `believe(hearer, want(speaker, believe(hearer, P, [C:C])), [C:C])` in the case of an *Inform* action).

4.2.3 An Express-Doubt Discourse Recipe Figure 4 presents our discourse recipe for expressing doubt. Note that its applicability conditions capture the requisite beliefs listed on page 4. The second applicability condition in Figure 4 excludes certain belief in _proposition2; this is because the body of the recipe is an action of conveying uncertain belief in _proposition2 and represents instances where an expression of doubt is realized

12 In personal communication, Allen has said that the effect of his Inform action was intended to capture the agent's goal in performing the action. In (Allen, 1979) he mentions the need for a Decide-to-Believe act, but nothing further is done with it.

Discourse Recipe

Action:	Express-Doubt(_agent1, _agent2, _proposition1, _proposition2)

{*_agent1 expresses doubt to _agent2 about _proposition1 by contending that _proposition2 is true*}

Recipe-Type:	Decomposition
Appl Cond:	believe(_agent1, believe(_agent2, _proposition1, [S:C]), [S:C])
	believe(_agent1, _proposition2, [W:S])
	believe(_agent1, _proposition2 $\rightarrow \neg$_proposition1, [S:C])
Constraints:	salient(_proposition1)
Body:	Convey-Uncertain-Belief(_agent1, _agent2, _proposition2)
Effects:	believe(_agent2, believe(_agent1, _proposition1, [S_N:W_N]), [S:C])
	believe(_agent2, believe(_agent1,_proposition2 $\rightarrow \neg$_proposition1, [S:C]), [S:C])
	believe(_agent2, want(_agent1, Resolve-Conflict(_agent2, _agent1, _proposition1, _proposition2)), [S:C])
Goal:	want(_agent2, Resolve-Conflict(_agent2, _agent1, _proposition1, _proposition2))

Figure 4
A Recipe for an Express-Doubt Discourse Act

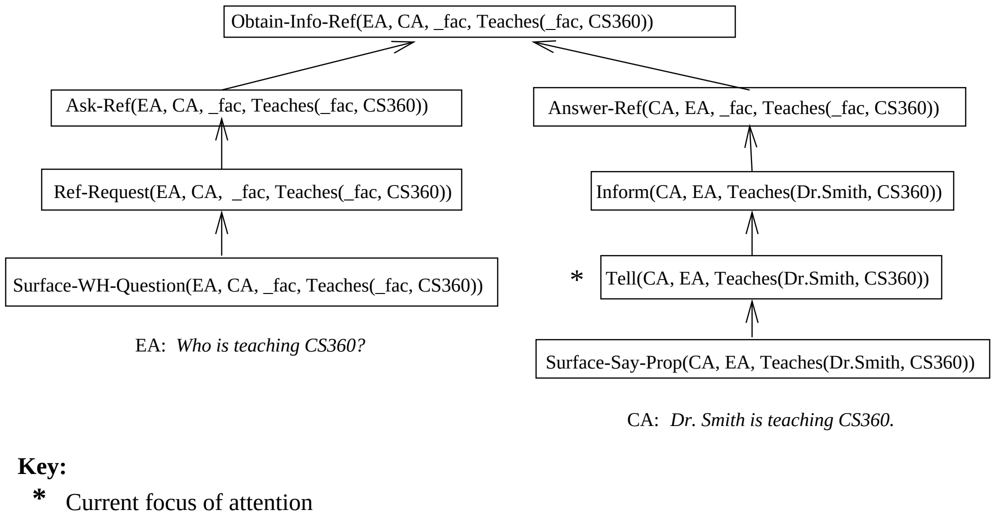

Figure 5
Sample Discourse Tree

as a surface negative question or a tag question. Note also that one of the effects of the *Express-Doubt* discourse act is that the listener believes that the speaker wants to resolve the conflict between the two propositions and that the goal of the *Express-Doubt* discourse act is that the listener also wants to resolve the conflict. The mutual desire for conflict resolution resulting from a successful *Express-Doubt* discourse act leads to a negotiation subdialogue (initiated by the *Express-Doubt* action).

4.3 The Dialogue Model

We maintain a structure called a **dialogue model** that captures the system's beliefs about the existing dialogue context. The discourse level of the dialogue model contains a tree structure called the **discourse tree**. Each node of the discourse tree represents a discourse or communicative act that has been initiated by one of the dialogue participants, and the children of a node represent discourse acts that are being pursued in order to perform the parent action. For example, asking and answering a question (*Ask-Ref* and *Answer-Ref*) are part of obtaining information (*Obtain-Info-Ref*) in the discourse tree in Figure 5. The lowest uncompleted[13] action in the discourse tree is marked as the **focus of attention**; it represents the first expectation for subsequent utterances. In Figure 5, the focus of attention is the *Tell* action. The **active path** consists of the sequence of actions along the path from the action that is the focus of attention to the root node. The actions on the active path provide successive expectations about the role of the next utterance in the dialogue; actions closer to the current focus of attention are regarded as more salient than those further back on the active path.[14] For example, in the discourse tree of Figure 5, the first expectation is that if EA does not understand CA's previous statement, then EA will now choose to address her understanding of it and thereby contribute to the *Tell* act that is the existing focus of attention. The next expectation is

13 By uncompleted, we mean not as yet known to be completed.
14 The active path is a sequence of actions; this work has not considered multi-threaded discourse, a topic that Rosé et. al. (Rosé et al., 1995) have begun to investigate.

(10)	EA:	*Who is teaching CS360 (a systems course)?*
(11)	CA:	*Dr. Smith is teaching CS360.*
(12)	EA:	*Isn't Dr. Smith on sabbatical?*
(13)	CA:	*No, Dr. Smith is not on sabbatical.*
(14a)	EA:	*Wasn't Dr. Smith awarded a Fulbright?*
(14b)	EA:	*Isn't Dr. Smith a theory person?*
(14c)	EA:	*Isn't Dr. Smith an excellent teacher?*

Figure 6
A Sample Dialogue with Surface-Negative Questions

that if EA has any doubt about the proposition conveyed by CA, then EA will choose to address its believability, thereby contributing to the *Inform* discourse act that is the next action on the active path. (As shown in Figure 3, *Address-Understanding* is a subaction in the recipe for the *Tell* discourse act, and *Address-Believability* is a subaction in the recipe for the *Inform* discourse act.)

4.4 Hypothesizing Discourse Acts by Chaining

Our process model starts with the semantic representation of a new utterance and uses plan inference rules (Allen and Perrault, 1980; Carberry, 1988) along with constraint satisfaction (Litman and Allen, 1987) to hypothesize chains of actions $A_1, A_2, \ldots, A_n$ that the speaker might be intending to perform with the utterance. In such a chain, action A_i contributes to the performance of its successor action A_{i+1}. For example, the semantic representation of an utterance such as *"Dr. Smith is teaching Architecture"* is

Surface-Say-Prop(_agent1, _agent2, Teaches(Dr.Smith, Architecture))

A Surface-Say-Prop is a subaction in the recipe for a *Tell* discourse act, which in turn is a subaction in the recipe for an *Inform* discourse act. Thus chaining from the surface utterance produces a sequence of hypothesized discourse acts, each of which plays a role in the performance of its successor on the chain.

We have expanded on Litman and Allen's notion of constraint satisfaction (Litman and Allen, 1987) and Allen and Perrault's use of beliefs (Allen and Perrault, 1980). As described earlier, many of the applicability conditions in our discourse recipes are beliefs that the agent of the action must hold in order for the action to be felicitous. Our recognition algorithm requires that the system be able to plausibly ascribe these beliefs in hypothesizing a new action; if the belief ascription is implausible or if the constraints of the discourse recipe are not satisfied, the inference is rejected.

4.5 Evidence Actions

As we claimed in Section 3, actions such as the expressions of doubt in utterances (14a) and (14b) (repeated in Figure 6) require evidence for their recognition. Let us further examine why this is the case. Figure 7 illustrates a situation in which we have several discourse acts (with different degrees of salience) that the agent might be expected to pursue. The solid boxes show some of the actions that are part of the existing dialogue context[15] and the dashed boxes show a chain of actions inferred from the new

15 These are actions on the active path of the dialogue model; the actions that are deepest on the active path are closer to the current focus of attention and are therefore regarded as more salient.

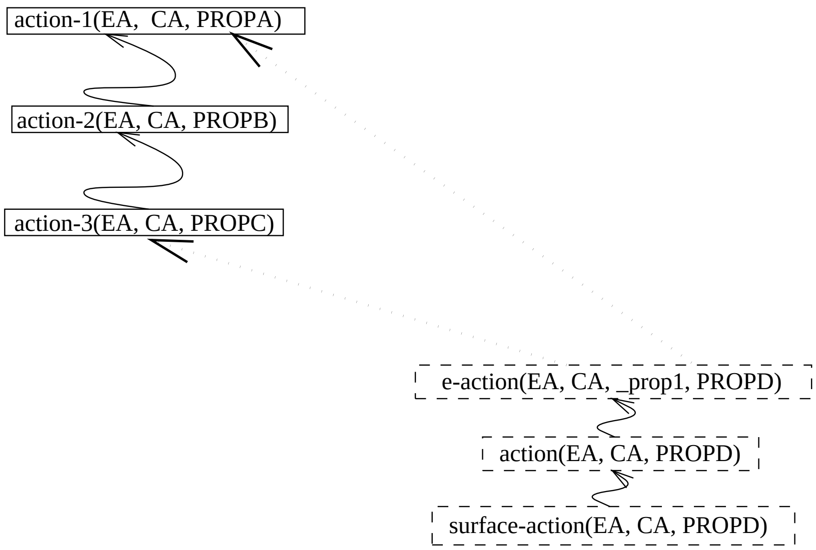

Figure 7
Relating an Inference Path to the Existing Dialogue Context

utterance. As depicted in the figure, the process model can infer a chain of actions starting with some surface speech act `surface-action(EA, CA, PROPD)`, up to some other action, `action(EA, CA, PROPD)`, up to some other action, `e-action(EA, CA, _prop1, PROPD)`. The action `e-action` contains two propositions. One of these, `PROPD`, is instantiated by chaining from the earlier action, `action(EA, CA, PROPD)`. However, the other proposition, `_prop1`, cannot be instantiated by plan chaining; it must be instantiated by unification with a proposition from the existing dialogue context. For example, if `e-action` is identified as contributing to `action-3` in the existing dialogue context, then `_prop1` might be instantiated as `PROPC`. On the other hand, if `e-action` is identified as contributing to `action-1`, then `_prop1` might be instantiated as `PROPA`. Chaining might suggest three possibilities (`e-action` could contribute to `action-3` or to `action-1` or to neither of them), and the relative salience of the propositions in the existing dialogue context is not sufficient to identify this relationship.

As a concrete example, consider again the dialogue segment in Figure 6 consisting of utterances 10-13 followed by one of utterances 14a-14c. After utterance (13), the actions on the active path of the dialogue model include (among others)

action-1: Inform(CA, EA, Teaches(Smith,CS360))
action-2: Address-Unacceptance(EA,CA, Teaches(Smith,CS360), On-Sabbatical(Smith))
action-3: Inform(CA, EA, ¬On-Sabbatical(Smith))

If EA utters 14a, 14b, or 14c, three inference paths can be constructed: one that links up to action-1, one that links up to action-3, and one that does not link up to any action on the active path. If CA utters (14a), then CA's action should be identified as contributing to action-3 above and thus the proposition being doubted should be recognized as ¬On-Sabbatical(Smith), even though this proposition did not appear explicitly in EA's utterance. However, if EA utters (14b), then EA's action should be identified as contributing only to action-1 and the proposition being doubted therefore should be recognized as Teaches(Smith, CS360); in this case, we are rejecting the inference path that links up to action-3 even though action-3 is more salient at this point in the dialogue. On the other hand, if EA utters (14c), then EA's action should be recognized as not contributing to any of the actions on the active path and interpreted as merely seeking information about Dr. Smith. Since chaining and salience alone are insufficient to identify the correct interpretation, we need some additional mechanism.

We define an **evidence-action** (abbreviated **e-action**) to be an action that introduces a new parameter which cannot be directly instantiated by chaining from the utterance. We contend that such actions require evidence for their recognition. In our model, the relationship between `_prop1` (the proposition whose instantiation must be inferred from the existing dialogue context) and `PROPD` (a proposition instantiated by chaining from the current utterance) is modeled in the applicability conditions of a recipe for `e-action`. For example, *Express-Doubt*, whose recipe was given in Figure 4 on page 16, is an example of an e-action. The parameter `_proposition1` cannot be instantiated from plan chaining from the surface utterance because `_proposition1` does not appear in the body of the *Express-Doubt* recipe; therefore, *Express-Doubt* is an e-action because it contains a parameter (`_proposition1`) that must be instantiated by unification with a proposition extracted from the existing dialogue context. The relationship that `_proposition1` has to the proposition contained in the utterance that is expressing doubt is modeled in the last applicability condition of the *Express-Doubt* recipe (see Figure 4). This applicability condition states that the agent of the *Express-Doubt* action believes that `_proposition2` implies that `_proposition1` does not hold. As we've shown earlier, plan chaining and plausibility are insufficient for recognizing an *Express-Doubt* discourse act; evidence is required.

4.5.1 Types of Evidence Our recognition algorithm captures the kinds of evidence identified in Section 3: 1) evidence provided by world knowledge, contextual knowledge, and the surface form of the utterance indicating that the applicability conditions for an e-action are satisfied, and 2) linguistic evidence from clue words suggesting a generic discourse action.

Grosz and Sidner claim that when evidence is available from one source, less evidence should be required from others (Grosz and Sidner, 1986). Thus, if there is evidence indicating that the applicability conditions for a discourse act hold, then less linguistic evidence suggesting the discourse act should be required. This is the case for interpreting (9) (repeated below) as an expression of doubt.

(7) EA: *What is Dr. Smith teaching?*
(8) CA: *Dr. Smith is teaching Architecture.*
(9) EA: *Isn't Dr. Brown teaching Architecture?*

Even though there is no linguistic clue word suggesting an *Express-Doubt* discourse act, there is enough evidence from the surface form of the utterance and from world and contextual knowledge to correctly interpret (9) as an expression of doubt at the proposition conveyed by (8).

Let us examine this evidence in more detail. The applicability conditions of the *Express-Doubt* discourse act (Figure 4 on page 16) specify that EA must have some belief in each of the following:

(a) that CA believes that Dr. Smith is teaching Architecture;
(b) that Dr. Brown is teaching Architecture; and
(c) that Dr. Brown teaching Architecture is an indication that Dr. Smith is *not* teaching Architecture.

Belief (c) models how the proposition in utterance (9) (that Dr. Brown is teaching Architecture) relates to the proposition in the existing dialogue context (that Dr. Smith is teaching Architecture). Therefore, evidence that EA holds belief (c) (that Dr. Brown teaching Architecture is an indication that Dr. Smith is *not* teaching Architecture) is particularly significant since it shows *how* the utterance relates to the preceding discourse.

The system has evidence for all three applicability conditions. The system's evidence that EA holds belief (a) is provided by beliefs derived from the goal of the *Tell* discourse act. In utterance (8), CA initiates a *Tell* discourse act as part of an *Inform* discourse act; thus immediately after (8), both the *Inform* and the *Tell* are part of the existing dialogue context. If (9) is indeed an expression of doubt, then it contributes to the *Inform* act by addressing the believability of the communicated proposition. In this case, EA will have implicitly conveyed that EA understood CA's previous utterance – i.e., EA will have passed up the opportunity to contribute to the *Tell* discourse act that is a child of the *Inform* act, and will have thereby implicitly conveyed that the *Tell* act was successful. This notion of implicit acceptance is discussed further in Section 4.6. Since the goal of CA's *Tell* act is that EA believe that CA believes that Dr. Smith teaches Architecture, the hypothesis that the *Tell* act has completed successfully (and therefore that its goal has been achieved) provides evidence that (a) is a belief held by EA.

The surface form of (9) provides evidence that EA believes (b), since it conveys an uncertain but still strong belief that Dr. Brown is teaching Architecture. Finally, if the system's model of a stereotypical user indicates that users typically believe that each course has only one instructor, then this world knowledge provides evidence that EA believes (c). Thus, the system has evidence for all three of the applicability conditions. In addition, contextual knowledge indicates that the single constraint on the *Express-Doubt* discourse act is satisfied — namely, that proposition P_{doubt} be salient at this point in the dialogue. Thus the system would recognize (9) as an expression of doubt.

However, *"Isn't Dr. Smith an excellent teacher?"* would not be recognized as an expression of doubt because the system would have no evidence that EA believes that being an excellent teacher suggests that Dr. Smith is not teaching Architecture. Thus, world knowledge helps the system to correctly differentiate between utterances that are expressions of doubt and those that are not.

On the other hand, if there is sufficient linguistic knowledge suggesting a particular discourse action, then the applicability conditions should be attributed to the speaker as long as they are plausible. So, if the clue word *but* is used, then a non-acceptance discourse action such as expressing doubt should be easier to recognize (i.e., should require less evidence that the applicability conditions hold) than if the clue word is not present. Thus, if EA said *"But isn't Dr. Smith an excellent teacher?"*, then even though there is no world knowledge indicating that all of the applicability conditions hold, the linguistic clue word is sufficient evidence to interpret this utterance as an expression of doubt.

The concept of accommodation in conversation(Lewis, 1979; Thomason, 1990) (removing obstacles to the speaker's goals) suggests that a listener might recognize a surface negative question as an expression of doubt by *accommodating* a belief about some incompatibility between the proposition conveyed by the surface negative question and a proposition that might be doubted. But in the extreme case, this means that any surface negative question could be recognized as an expression of doubt. We contend that there should be evidence for such recognition. This is similar to Pollack's model of plan recognition(Pollack, 1990) that can account for user misconceptions; instead of inferring a relationship between every query and the speaker's goal, Pollack requires that the system apply only well-motivated rules that hypothesize principled variations of the system's own beliefs and that the system treat as incoherent any queries that cannot be explained via these rules. (Pollack's example of incoherence is the query *"I want to talk to Kathy, so I need to find out how to stand on my head."*) In our model we look for evidence of incompatibility, and in our implemented system this evidence takes the form of stereotypical beliefs about the domain. While our implementation does not include other means of deducing an incompatibility, they are not precluded by our theory but are left for future work. Moreover, it should be noted that if the speaker intends for the hearer to recognize the expression of doubt from the incompatibility between the doubted proposition and the proposition that the speaker is contending is true, then the speaker must believe that the belief about the incompatibility is a mutual belief. Our stereotypical beliefs fall into this category. In other cases, the clue word *but* causes the listener to accommodate a belief about incompatibility that he might not otherwise have done and thereby recognize the expression of doubt.

So in the case of complex discourse acts such as expressing doubt, the system should require evidence for recognizing the discourse act and should prefer to recognize discourse acts for which there is multiple evidence: both linguistic clue words suggesting the generic discourse act and evidence suggesting that the applicability conditions for a particular discourse act are satisfied. However, the system should be willing to accept just one kind of evidence when that is all that is available.

4.6 Implicit Acceptance

In a collaborative task-oriented dialogue, the participants are working together to construct a plan for accomplishing a task. If the collaboration is to be successful, the participants must agree on the plan being constructed and the actions being taken to construct it. Thus, since a communicated proposition is presumed to be relevant to this plan construction process, the dialogue participants are obligated to communicate as soon as possible any discrepancies in belief about such propositions(Walker and Whittaker, 1990;

Chu-Carroll and Carberry, 1995b) and to enter into a negotiation subdialogue in which they attempt to *square away* (Joshi, 1982) their disparate beliefs.

In our earlier work (Carberry, 1985; Carberry, 1989), we claimed that a cooperative participant must accept a response or pursue discourse goals directed toward being able to accept the response. As we noted there, this acceptance need not be explicitly communicated to the other participant; for example, failure to initiate a negotiation subdialogue conveys **implicit acceptance** of the proposition communicated by an *Inform* action. This notion of implicit acceptance is similar to an expanded form of Perrault's default reasoning about the effects of an inform act (Perrault, 1990). Our model captures this by recognizing implicit acceptance when an agent foregoes the opportunity to address acceptance of an action and moves on to pursue other discourse actions.

4.6.1 Acceptance Actions If a statement is intended to answer a question, the listener in a collaborative dialogue must note when he believes that the statement does not suffice as a complete answer. However, doing so implies that the listener believes the statement, since it is inefficient to address a proposition's completeness as an answer if one does not accept the proposition. Similarly, if a listener does not believe a communicated proposition, he must convey this disagreement as soon as possible(Walker and Whittaker, 1990). But by questioning the validity of a proposition, the listener conveys that he believes that he understood the utterance. As Clark and Schaefer note(Clark and Schaefer, 1989), by passing up the opportunity to ask for a repair, a listener conveys that he has understood an utterance. Thus we hypothesize that listeners convey their acceptance (or lack of acceptance) in a multi-stage acceptance phase: 1) understanding, 2) believability, 3) completeness.[16]

Acceptance can be communicated explicitly or implicitly. We include actions that address acceptance in the body of six of our discourse recipes. These recipes were selected because they allow us to capture acceptance of a question (the recipes for *Ask-Ref* and *Ask-If*), acceptance of the answer to a question (the recipes for *Answer-Ref* and *Answer-If*), and acceptance of a statement (the recipes for *Inform* and *Tell*). In this research, we have been primarily concerned with one aspect of acceptance: believing the proposition communicated by an *Inform* action. For example, the actions in the body of the *Inform* recipe (see Figure 3 on page 13) are: 1) the speaker (`_agent1`) *tells* the listener (`_agent2`) the proposition that the speaker wants the listener to believe; and 2) the speaker and listener *address believability* by discussing whatever is necessary[17] in order for the listener and speaker to come to an agreement about this proposition. This second action, and the subactions executed as part of performing it, account for subdialogues that address the believability of the proposition conveyed in the *Inform* action. Other actions related to acceptance are captured in other discourse recipes. For example, the *Tell* action has a body containing a *Surface-Say-Prop* action and an *Address-Understanding* action; the latter enables both participants to ensure that the utterance has been understood. Similarly, the *Answer-Ref* action contains an *Inform* action and an *Address-Answer-Acceptability* action which ensures that the *Inform* action is sufficient to answer the question. Further research is needed to model the full range of actions that address acceptance and to recognize utterances resulting from them.

16 Questions must also be accepted and assimilated into a dialogue. Our model has recently been expanded to address the acceptance of questions(Bartelt, 1996), but we are concentrating on statements in this paper.

17 Since our system does not generate responses, we do not model what the speakers need to discuss; however, if a speaker expresses doubt at a proposition by contending that a second proposition is true, then the speaker is introducing this second proposition and its relationship to the first proposition as items that need to be discussed.

(15)	EA:	*Who is teaching CS360 (a systems course)?*
(16)	CA:	*Dr. Smith is teaching CS360.*
(17a)	EA:	*Isn't Dr. Smith a theory person?*
(17b)	EA:	*Who handles the CS360 lab?*

Figure 8
Dialogues Conveying Different Implicit Acceptance

The discourse tree reflects the order of acceptance actions. As discussed above, lack of understanding should be addressed before believability. This is reflected in the discourse tree that results from a statement, such as the one in Figure 5 on page 17, where the *Tell* action (whose recipe contains an *Address-Understanding* action) is a descendant of the *Inform* action (whose recipe contains an *Address-Believability* action); in addition, since the statement in Figure 5 is intended to answer a question, the *Inform* act is a descendant of the *Answer-Ref* action (whose recipe contains an *Address-Answer-Acceptability* action). Since the *Tell* is the current focus of attention, it must be completed before other actions are pursued. Thus if the listener believes that the *telling* has not been successful (i.e., the listener does not fully understand the utterance), then the listener will pursue discourse acts that contribute to its *Address-Understanding* subaction. Once the *Tell* has been successfully completed, then attention reverts back to the *Inform* act. The *Inform* must be successfully completed before other higher-level acts are pursued further. Thus if the *Inform* has not been successful (i.e., the listener does not accept the communicated proposition), then the listener will pursue discourse acts that contribute to its *Address-Believability* subaction.

We have concentrated primarily on recognizing the acceptance and non-acceptance of propositions communicated by *Inform* actions: i.e., modeling negotiation subdialogues in which participants do not automatically believe everything that they are told. Others (Allen and Schubert, 1991; Clark and Schaefer, 1989; Traum and Hinkelman, 1992; Traum, 1994) have investigated how understanding and lack of understanding are communicated and can be recognized.

4.6.2 Modeling Implicit Acceptance Our system models implicit acceptance in collaborative dialogue as passing up the opportunity to express lack of acceptance. For example, consider the two dialogue variations shown in Figure 8. Figure 5 depicts the discourse tree constructed from utterances (15) and (16) in Figure 8, with the current focus of attention, the *Tell* action, marked with an asterisk. In attempting to assimilate (17a) into this discourse tree, the system's first expectation is that (17a) will address the understanding of (16) if EA does not understand it (i.e., as part of the *Tell* action which is the current focus of attention in Figure 5). The next expectation is that (17a) will relate to the *Inform* action in Figure 5, by addressing the believability of the proposition conveyed by (16). The system finds that the best interpretation of (17a) is that of expressing doubt at the proposition that Dr. Smith is teaching CS360, thus confirming the secondary expectation that (17a) is addressing the believability of the proposition conveyed by (16). This recognition of (17a) as part of the *Inform* action in Figure 5 indicates that EA has implicitly indicated understanding, by passing up the opportunity to address understanding in the *Tell* action that appears at a lower level in the discourse tree and by moving instead to a relevant higher-level action; (17a) is thus (implicitly) conveying that the *Tell* action has been successful.

Thus, when an utterance contributes to an ancestor of an action A_i and all of A_i's applicability conditions, except those negated by the goal, are still satisfied, then A_i is assumed to have completed successfully; if that were not true, the dialogue participants would have been required to address those actions.[18] When an action is recognized as successful, the system updates its model of the user's beliefs with the effects and goals of the completed action. For example, in determining whether (17a) in Figure 8 is expressing doubt at (16) (thereby implicitly indicating that (16) has been understood and that the *Tell* action has therefore been successful), the system tentatively hypothesizes that the effects and goals of the *Tell* action hold, resulting in the tentative belief that EA believes that CA believes that Dr. Smith is teaching CS360. If the system determines that this *Express-Doubt* action is the most coherent interpretation of (17a), it attributes the hypothesized beliefs to EA.

Now consider a dialogue in which utterances (15) and (16) in Figure 8 are instead followed by utterance (17b). In this case, the system finds that the best interpretation of (17b) does not contribute to any of the actions in the existing discourse tree; instead (17b) is identified as initiating an entirely new *Obtain-Info-Ref* action at the discourse level, resulting in a new discourse tree. Since EA has gone on to pursue some other discourse action unrelated to any of the acts that were part of the previous discourse tree, the system recognizes not only EA's understanding of (16) but also EA's implicit acceptance of the proposition conveyed by (16). That is, because the system interprets EA's utterance as foregoing the opportunity to initiate a negotiation subdialogue to address the acceptance of the proposition communicated by the *Inform* action, the system recognizes that the *Inform* action has been successful and that EA has implicitly conveyed acceptance of the proposition that Dr. Smith is teaching CS360.

4.7 The Recognition Algorithm

Our recognition algorithm, outlined in Figure 9, assimilates a new utterance into the existing dialogue context and identifies discourse acts that the speaker is pursuing. It proceeds as follows. Start with the semantic representation of the utterance and extract from it two kinds of linguistic information: 1) clue words that might suggest a generic discourse act, and 2) beliefs that are conveyed by the surface form of the utterance. In our implemented system, possible clue words are explicitly noted in the semantic representation of the utterance, and beliefs conveyed by the surface form of an utterance are extracted from the applicability conditions of the recipe for the surface speech act. For example, the surface form of an utterance such as *"Isn't Dr. Smith on sabbatical?"* conveys that the speaker has a strong but uncertain belief in the queried proposition; this is captured in the applicability conditions of the recipe for a *Surface-Neg-YN-Question* discourse act (see the Appendix).

Next use plan inference rules to hypothesize sequences of actions $A_1, A_{i_2}, \ldots, A_{i_{d_i}}$ (inference paths) such that A_1 is the surface action directly associated with the speaker's utterance and $A_{i_{d_i}}$ is an action on the active path in the existing dialogue context. By requiring that an inference path link up with an action that is already part of the existing dialogue context, we are capturing the expectation that the new utterance will contribute to an action that has already been initiated. This corresponds to a focusing heuristic that

18 By requiring that all applicability conditions, except those negated by the goal, still be satisfied in order for the action to be viewed as successful, we eliminate situations in which the agent of the *Inform* act has become convinced by the other participant that the proposition he was trying to convey is really false. In such cases, the applicability condition `believe(_agent1, _proposition)` of the *Inform* will no longer be true and thus the *Inform* act will not be viewed as completing successfully. We have not addressed situations in which the participants cannot resolve their disagreements and *agree to disagree*.

A_1 = surface action associated with the speaker's utterance
LE = clue words extracted from semantic representation of utterance
D = dialogue model
B = listener's beliefs
A_f = action at current focus of attention in D

;;Construct inference paths that link up to active path of dialogue model
S$\leftarrow\{P_i = A_1, A_{i_2}, \ldots, A_{i_{d_i}}$ | on-active-path$(A_{i_{d_i}}, D)$
$\wedge P_i$ is an inference path constructed from $A_1\}$

;;Eliminate inference paths with unsatisfied constraints or implausible applicability conditions
For each $P_i \in S$ Do
Begin
$B_i \leftarrow B$
If $A_{i_{d_i}} \neq A_f$
Then $B_i \leftarrow B_i \cup$ {beliefs that A_f and all actions on the active path between A_f and $A_{i_{d_i}}$ have completed successfully}
If $(\exists A_j)(\exists C_k)\ A_j \in P_i \wedge$ is-constraint$(C_k, A_j) \wedge \neg C_k$
Then $S \leftarrow S - P_i$
Else If $(\exists A_j)(\exists AC_k)\ A_j \in P_i \wedge$ is-app-cond$(AC_k, A_j) \wedge\neg$plausible(AC_k, B_i)
Then $S \leftarrow S - P_i$
End

;;Determine how much evidence is available for each e-action
$S_0 \leftarrow \emptyset$, $S_1 \leftarrow \emptyset$, $S_2 \leftarrow \emptyset$
For each $P_i \in S$ Do
If $(\exists A_j)\ A_j \in P_i \wedge$ e-action(A_j)
Then Begin
If ling-evid$(A_j$,LE$) \wedge$ app-cond-evid(A_j, B_i)
Then $S_2 \leftarrow S_2 \cup \{P_i\}$
Else If ling-evid$(A_j$,LE$) \vee$ app-cond-evid(A_j, B_i)
Then $S_1 \leftarrow S_1 \cup \{P_i\}$
End
Else $S_0 \leftarrow S_0 \cup \{P_i\}$;;S_0 contains inference paths with no e-actions

;;Select inference paths containing actions with the most evidence
If $S_2 \neq \emptyset$ Then $S \leftarrow S_2$;;S contains inference paths with multiple evidence
Else If $S_1 \neq \emptyset$ Then $S \leftarrow S_1$;;S contains inference paths with evidence
Else $S \leftarrow S_0$;;S contains inference paths with no e-actions

;;Select inference path containing the action closest to current focus of attention
If $S \neq \emptyset$
Then $P \leftarrow P_i \mid P_i \in S \wedge P_i = A_1, A_{i_2}, \ldots, A_{i_{d_i}}$
$\wedge\neg(\exists P_j)\ P_j \in S\ \wedge P_j = A_1, A_{j_2}, \ldots, A_{j_{d_j}}$
$\wedge$ closer-to-curr-discourse-focus$(A_{j_{d_j}}, A_{i_{d_i}}$,D)
Else Begin
$B \leftarrow B \cup$ {beliefs that all actions on active path have completed successfully}
$S \leftarrow \{P_i = A_1, A_{i_2}, \ldots, A_{i_n} \mid P_i$ is an inference path constructed from A_1
$\wedge$ no-elts-on-active-path$(P_i$,D)
$\wedge\ (\forall A_j)(\forall C_k)(A_j \in P_i \wedge$ is-constraint$(C_k, A_j) \rightarrow C_k)$
$\wedge\ (\forall A_j)(\forall AC_k)(A_j \in P_i \wedge$ is-app-cond$(AC_k, A_j) \rightarrow$ plausible$(AC_k, B))$
$P \leftarrow P_i \mid P_i \in S \wedge \neg(\exists P_j)\ P_j \in S \wedge$ links-closer-to-ps-dom-focus$(P_j, P_i$,D)
End
;; Assimilate utterance into dialogue model
Add $P = A_1, A_{p_2}, \ldots, A_{p_k}$ to D
Mark A_{p_2} as new focus of attention in D

Figure 9
Pseudo-code Outlining Recognition Algorithm

captures expectations for new utterances in an ongoing dialogue (Carberry, 1990). For any inference path, if $A_{i_{d_i}}$ is not the focus of attention in the existing dialogue context, then $A_{i_{d_i}}$ must be an ancestor of the action that is the focus of attention; tentatively hypothesize that each of the actions on the active path between the existing focus of attention (the focus of attention immediately prior to the new utterance) and $A_{i_{d_i}}$ have completed successfully[19] and use this hypothesis in reasoning about the actions on the inference path. If the applicability conditions for any of the actions on an inference path are implausible or if the constraints are not satisfied, reject the inference path.

For actions that are e-actions, determine how much evidence is available for the action. Reject any inference paths containing an e-action for which there is neither linguistic evidence suggesting the generic discourse act (such as the clue word *but* suggesting an *Express-Doubt* action)[20] nor evidence from the surface form of the utterance, world knowledge, and contextual knowledge indicating that the applicability conditions for the particular discourse action are satisfied. If there is an e-action for which both kinds of evidence exist (both linguistic evidence for the generic discourse act and evidence that the applicability conditions are satisfied), then consider only inference paths containing an e-action for which there is such multiple evidence and select the inference path $A_1, A_{i_2}, \ldots, A_{i_{d_i}}$ for which $A_{i_{d_i}}$ is closest to the focus of attention in the existing dialogue context. Otherwise, if there is an inference path containing an e-action for which one kind of evidence exists, then select the inference path $A_1, A_{i_2}, \ldots, A_{i_{d_i}}$ for which $A_{i_{d_i}}$ is closest to the focus of attention in the existing dialogue context.

If a satisfactory inference path containing an e-action cannot be found, then consider inference paths that contain no e-actions.[21] If there is more than one such inference path, select the one that links up to an action that is closest to the focus of attention on the discourse level. If there is no inference path linking up to an action on the existing discourse level, then select the inference path that links up to an action that is closest to the focus of attention on the problem-solving and domain levels. (Our dialogue model actually contains three levels: domain, problem-solving, and discourse. However, this paper is primarily concerned with recognizing actions on the discourse level; we will briefly discuss the domain and problem-solving levels in Section 5.1.) This latter case corresponds to initiating a new discourse segment, and thus a new discourse tree is constructed at the discourse level.

Our algorithm identifies a best interpretation of the speaker's utterance. However, since the algorithm uses heuristics, its interpretation can be incorrect and miscommunication can result. Our current system does not include mechanisms for detecting and recovering from such errors. Clark and Schaeffer(Clark and Schaefer, 1989) discuss second, third, and fourth turn repairs in discourse, and McRoy and Hirst(McRoy and Hirst, 1995) provide an excellent formal model of repair in dialogue.

19 The action at the focus of attention and some of its ancestor actions may have completed successfully, which becomes evident when the participants choose not to address them further. For example, as a result of providing an answer to a question, the active path may include the discourse actions *Answer-Ref*, *Inform*, and *Tell* with the *Tell* discourse act being the current focus of attention; if the other participant then performs a discourse act that is a subaction of the *Answer-Ref* but not of the *Inform*, then he has accepted the proposition conveyed by the *Inform* and it has completed successfully.

20 Our system currently maintains a list of clue words and discourse acts that each clue word might suggest.

21 Due to length restrictions, we have omitted a part of the algorithm that deals with focusing heuristics that are not needed for the kinds of utterances addressed in this paper; an example of utterances needing the full algorithm is given in (Lambert and Carberry, 1991).

5 Modeling Negotiation Subdialogues

The preceding sections have provided the key mechanisms necessary for modeling negotiation subdialogues. Our recipes differentiate between the effects and the goals of a discourse act. Thus, instead of assuming that a communicated proposition will automatically be accepted by the listener, the effect of our *Inform* action is only that the listener believes that the speaker wants the listener to believe the communicated proposition, while its goal is that the listener will actually adopt the proposition as one of his own beliefs. In addition, the body of the *Inform* discourse recipe contains not only an action capturing the *telling* of the proposition but also an action capturing the participants' addressing the believability of the communicated proposition. Our algorithm for recognizing discourse actions and assimilating them into the dialogue model can recognize when an agent is expressing doubt at a communicated proposition by contending that some other proposition is true. Our ability to recognize implicit as well as explicit acceptance of a communicated proposition enables us to identify when an agent has adopted a communicated proposition as part of his beliefs.

This section describes our implementation and demonstrates our system's capability with two extended negotiation subdialogues that illustrate 1) the role of linguistic, contextual, and world knowledge in resolving expressions of doubt, 2) expressions of doubt at both immediately preceding and earlier utterances, 3) multiple expressions of doubt at the same proposition, 4) negotiation subdialogues embedded within other negotiation subdialogues, and 5) explicit and implicit acceptance. The recipes for the discourse acts used in these examples can be found in the Appendix.

5.1 Implementation

Our system for recognizing complex discourse acts and handling negotiation subdialogues has been integrated into the tripartite dialogue model presented in (Lambert and Carberry, 1991). This dialogue model contains three levels of tree structures, one for each kind of action discussed in Section 4.2 (domain, problem-solving, and discourse) with links among the actions on different levels. At the lowest level, the discourse actions are represented; these actions may contribute to the problem-solving actions at the middle level which, in turn, may contribute to the domain actions at the highest level. Figure 10 illustrates the tripartite dialogue model for a situation in which CA has previously answered a question about the cost of registering for CS180, and then EA asks *"When does CS180 meet?"*. Note that the discourse level in Figure 10 only reflects the current query about when CS180 meets, since previous queries have already achieved their discourse goals. Since this paper is concerned almost exclusively with the discourse level of the dialogue model, we will not discuss the overall tripartite model further, except to note that the construction of a new discourse tree requires that the system identify its relationship to existing or new actions at the problem-solving and domain levels (Lambert and Carberry, 1991).

Our system has been implemented in Common Lisp on a Sun Sparcstation and tested in a university-advisement domain. Figure 11 lists some of the beliefs included in the system's model of a stereotypical user. In our current implementation, only the clue word *but* is recognized as linguistic evidence for an *Express-Doubt* discourse act. In future work, we will expand the clue words taken into account by our system.

22 $\Rightarrow$ is *usually implies* rather than strict implication. The semantics of this predicate is that there may be a small number of cases where the antecedent is true and the consequent is not. This is similar to a default rule. For example, the On-Campus rule might be viewed as $\forall y$: on-campus(y) $\wedge$ faculty(y) $\wedge$ M $\neg$on-sabbatical(y) $\rightarrow$ $\neg$on-sabbatical(y). However, as with any default rule, there

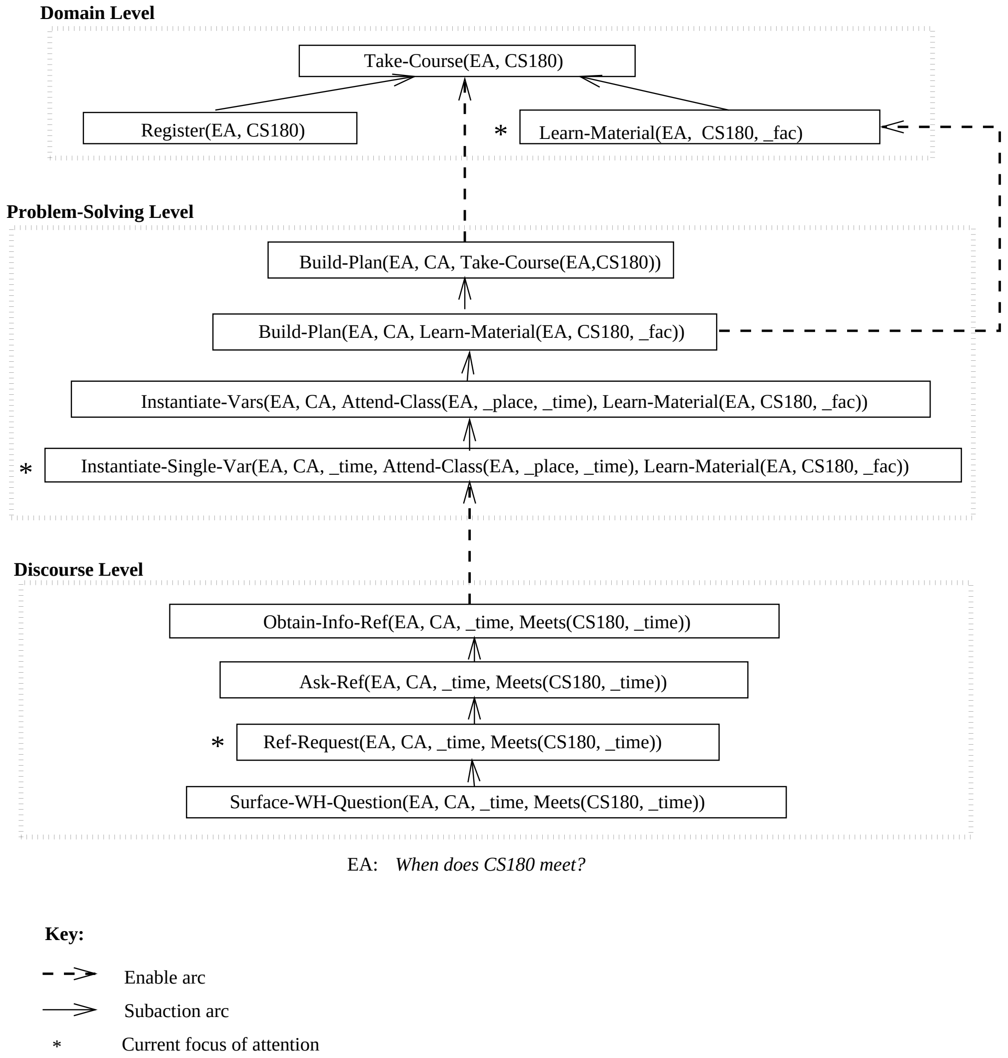

Figure 10
A Simple Tripartite Dialogue Model

Sabbatical rule: Teachers on sabbatical usually do not teach.
$(\forall x\ \forall y\ \text{course(y)} \wedge \text{faculty(x)} \wedge \text{on-sabbatical(x)} \Rightarrow \neg\text{teaches(x, y)})$

On Campus rule: Teachers on campus usually are not on sabbatical.
$(\forall y\ \text{faculty(y)} \wedge \text{on-campus(y)} \Rightarrow \neg\text{on-sabbatical(y)})$

One Course rule: Teachers usually teach only one course a semester.
$(\forall x\ \forall y\ \forall z \neq y\ \text{faculty(x)} \wedge \text{course(y)} \wedge \text{course(z)} \wedge \text{teaches(x, y)} \Rightarrow \neg\text{teaches(x, z)})$

One Professor rule: Each course usually has only one instructor.
$(\forall x\ \forall y\ \forall z \neq y\ \text{course(x)} \wedge \text{faculty(y)} \wedge \text{faculty(z)} \wedge \text{teaches(y, x)} \Rightarrow \neg\text{teaches(z, x)})$

Expertise rule: Teachers usually do not teach courses outside their area of expertise.
$(\forall x\ \forall y\ \forall z\ \text{faculty(x)} \wedge \text{course(y)} \wedge \text{area(z)} \wedge \text{specialty(x, z)} \wedge \neg\text{in-area(y, z)}$
$\Rightarrow \neg\text{teaches(x, y)})$

Figure 11
A Few Stereotypical Beliefs[22]

(18)	EA:	*What is Dr. Smith teaching?*
(19)	CA:	*Dr. Smith is teaching Architecture.*
(20)	EA:	*Isn't Dr. Brown teaching Architecture?*
(21)	CA:	*No, Dr. Brown is not teaching Architecture.*
(22)		*Dr. Brown is on sabbatical.*
(23)	EA:	*But wasn't Dr. Brown on campus yesterday?*
(24)	CA:	*Yes, Dr. Brown was on campus yesterday.*
(25)		*Dr. Brown was giving a University colloquium.*
(26)	EA:	*OK.*
(27)		*But isn't Dr. Smith a theory person?*

Figure 12
Extended Negotiation Subdialogue

5.2 An Extended Example

Figure 12 contains an extended negotiation dialogue (portions of this dialogue have been given earlier). This dialogue illustrates a number of features that our system can handle. Utterances (18) and (19) establish the initial context in which CA has provided information in response to a question from EA. Utterances (20), (23), and (27) illustrate the use of world knowledge in resolving expressions of doubt. Utterance (20) is an expression of doubt that initiates a negotiation subdialogue, indicating that EA has not accepted the proposition communicated by CA. Utterances (21) and (22) attempt to resolve this doubt by stating that Dr. Brown is not teaching Architecture and providing support for this claim. Utterance (23) expresses doubt at the supporting information; utterances (24) and (25) attempt to resolve this doubt; and utterance (26) explicitly accepts the proposition communicated by (25). Thus utterances (23)-(26) constitute a negotiation subdialogue embedded within the negotiation subdialogue of utterances (20)-(27). Although utterance (26) explicitly conveys acceptance of at least the most salient communicated proposition (the proposition that Dr. Brown was giving a University colloquium), there are still several propositions that have not yet been completely accepted by EA and are thus open for rejection. Utterance (27) again expresses doubt at the proposition that Dr. Smith is teaching Architecture, and implicitly conveys acceptance of the propositions that Dr. Brown is on sabbatical and that Dr. Brown is not teaching Architecture.

The rest of this section works through the details of how our system processes these utterances, recognizes the discourse acts they are pursuing, and incrementally builds the discourse tree of the dialogue model. In our examples, the system will switch between playing the role of EA and the role of CA. However, when processing an utterance, the system will have access only to the beliefs of the participant whose identity it has assumed (namely, the listener), along with the correct dialogue model at the time the utterance is made.

Utterance 18: Establishing the Initial Context

The system first plays the role of CA (the listener) and must understand EA's utterance of (18). The semantic representation of (18) is:

could be exceptions; for example, one might be on sabbatical but have returned to campus to give a colloquium.

```
Surface-WH-Question(EA, CA, _course, Teaches(Dr.Smith, _course))
```

The Surface-WH-Question is a subaction in the body of a recipe for a *Ref-Request* discourse act; the *Ref-Request* is a subaction in the recipe for an *Ask-Ref* discourse act; and the *Ask-Ref* is a subaction in the recipe for an *Obtain-Info-Ref* discourse act. Therefore the following chain of actions is hypothesized:

```
Obtain-Info-Ref(EA, CA, _course, Teaches(Dr.Smith, _course))
                                 ↑
Ask-Ref(EA, CA, _course, Teaches(Dr.Smith, _course))
                                 ↑
Ref-Request(EA, CA, _course, Teaches(Dr.Smith, _course))
                                 ↑
Surface-WH-Question(EA, CA, _course, Teaches(Dr.Smith, _course))
```

As each of the above actions is inferred, the system checks that its constraints are satisfied and that its applicability conditions are plausible. Since this is the only chain of actions suggested by plan inferencing on the discourse level, the system *recognizes* these discourse actions; it then infers problem-solving actions from the discourse actions and eventually domain actions from the problem-solving actions. As actions are recognized, the system updates its model of EA's beliefs, wants, and knowledge from the actions' applicability conditions. Figure 13 shows the initial tripartite dialogue model that is produced. Since this paper is primarily concerned with the recognition of actions on the discourse level, the remainder of our figures will only display the discourse level and will omit the problem-solving and domain level actions.

Utterance (19): Answering the Question

The system is now playing the role of EA (listener) and must understand CA's utterance of (19). The semantic representation of (19) is:

```
Surface-Say-Prop(CA, EA, Teaches(Dr.Smith, Arch))
```

Chaining suggests that the surface speech act might be part of a *Tell* action which might be part of an *Inform* action since the surface speech act and the *Tell* act are part of the body of the *Tell* and *Inform* acts respectively. The applicability conditions for all of these actions are plausible.

The system tries to extend the inference chain from the *Inform* action. An *Inform* can be part of the recipes for several discourse actions, including *Give-Background* and *Answer-Ref*. However, these actions are e-actions and, with the exception of *Answer-Ref*, inference of these e-actions is rejected. For example, *Give-Background* is an e-action because it relates the proposition in the current utterance to some other proposition, the proposition about which background is being given. The recipe for *Give-Background* contains a constraint that there be a particular relationship between the proposition in the *Inform* action in its body and some other proposition conveyed by CA. Since CA has made no previous utterances, there is no other proposition conveyed by CA and thus this constraint cannot be satisfied. Consequently *Give-Background* is rejected.[23] A full discussion of the *Give-Background* action and its recipe can be found in (Lambert, 1993).

23 Although CA can provide background information prior to conveying the proposition about which the background is being given, the *Give-Background* action in these instances will be recognized in assimilating CA's second utterance (the utterance about which the background is being given) (Lambert and Carberry, 1991).

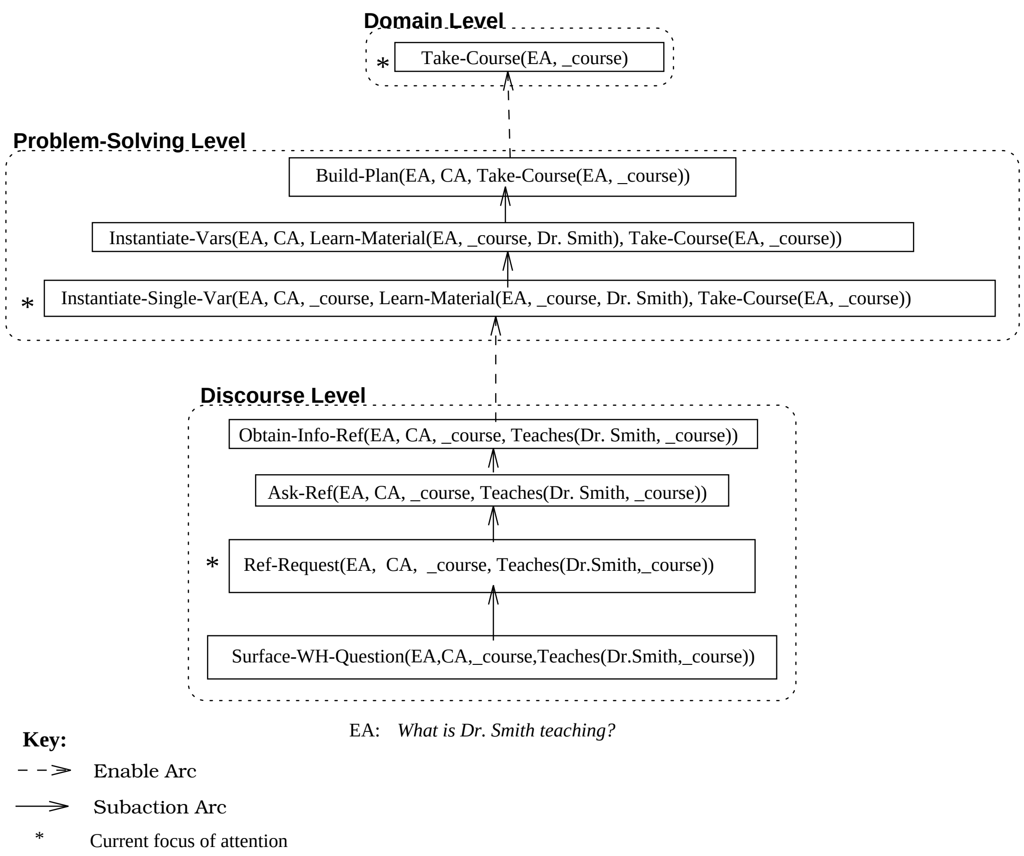

Figure 13
Tripartite Dialogue Model for Utterance (18)

On the other hand, `Answer-Ref(CA, EA, _term, _proposition)` can be inferred from `Inform(CA, EA, Teaches(Dr.Smith, Arch))` and the system has evidence for its recognition. *Answer-Ref* is an e-action since the parameters `_term` and `_proposition` cannot be instantiated from the *Inform* action that precedes it on the inference chain. As discussed in Section 4.5.1, evidence for e-actions may take one of two forms: 1) evidence from world and contextual knowledge and the surface form of the utterance indicating that the applicability conditions for a particular e-action are satisfied, and 2) linguistic evidence from clue words suggesting a generic discourse action. In this case, there are no clue words, so any evidence must be from world and contextual knowledge or the surface form of the utterance. `Answer-Ref(CA, EA, _term, _proposition)` can be a subaction in the body of `Obtain-Info-Ref(EA, CA, _term, _proposition)`; unifying with the *Obtain-Info-Ref* action that is part of the existing discourse tree causes the parameters `_proposition` and `_term` to be instantiated as `Teaches(Dr.Smith, _course)` and `_course` respectively in both the *Obtain-Info-Ref* and *Answer-Ref* actions.

World and contextual knowledge provide evidence that the applicability conditions of *Answer-Ref* are satisfied with these instantiations. The third applicability condition in the *Answer-Ref* recipe captures the required relationship between the new parameter `_proposition` and the parameter `_propanswer` that appears in the *Inform* discourse act. It indicates that CA must believe that `_propanswer` (where `_propanswer` is instantiated from the *Inform* act as `Teaches(Dr.Smith, Arch)`) is an instance of the queried proposition, `_proposition`, with the queried term `_term` instantiated. Since the system (playing the role of EA in this case) believes that the participants have equivalent knowledge about language and how terms can be instantiated[24] and since the system believes that the two propositions unify, the system has evidence that the third applicability condition is satisfied.

In addition, there is evidence that the other two applicability conditions are satisfied. With the above instantiations, these applicability conditions become:

Applicability conditions for Answer-Ref

```
believe(CA,want(EA,knowref(EA,_course,believe(CA,Teaches(Dr.Smith,_course),
                                                      [C:C]))),[W:C])
believe(CA, ¬knowref(EA,_course,believe(CA,Teaches(Dr.Smith,_course),
                                                [C:C])), [W:C])
```

If utterance 19 is in fact an *Answer-Ref* that contributes to the *Obtain-Info-Ref* that is part of the existing dialogue context, then CA has recognized and is responding to the *Ask-Ref* that is a child of the *Obtain-Info-Ref* in the existing dialogue context. Consequently, in considering the *Answer-Ref* interpretation, the system can hypothesize that CA has recognized the *Ask-Ref* and can tentatively attribute to CA the belief that the *Ask-Ref*s applicability conditions were satisfied.

Beliefs attributed to CA
(by virtue of hypothesis that CA has recognized the Ask-Ref)

```
want(EA, knowref(EA, _course, believe(CA, Teaches(Dr.Smith, _course),
                                            [C:C]))
¬knowref(EA, _course, believe(CA, Teaches(Dr.Smith, _course), [C:C]))
```

24 This does not mean that the instantiation will result in a true proposition, only that it is a legal instantiation of the term. For example, CS180 is a legal instantiation of the _course term in the proposition Teaches(Jones _course) although Teaches(Jones, CS180) may be false. We have not addressed the problem of misconceptions about class membership.

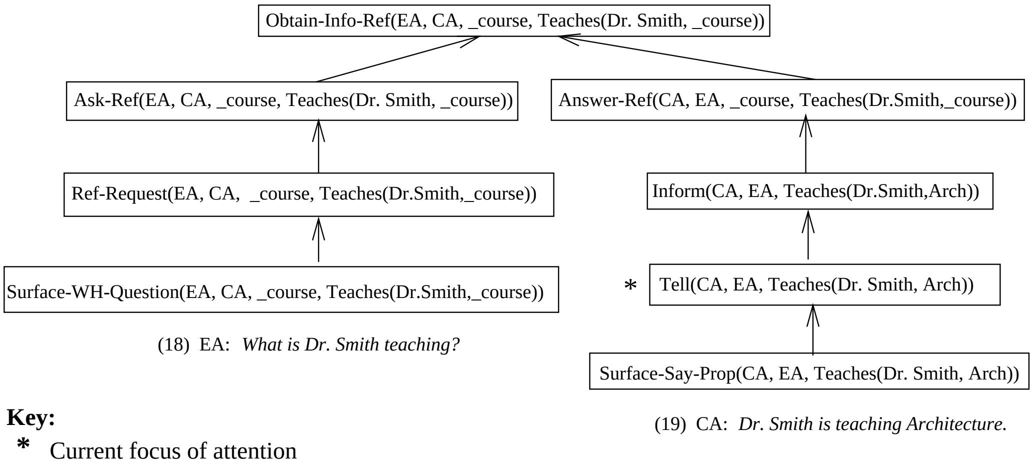

Figure 14
Discourse Tree for First Two Utterances in Figure 12

This is equivalent to tentatively hypothesizing that CA has recognized the intentions communicated by utterance 18 and inferred the discourse level of the dialogue model depicted in Figure 13. Thus the system's model of CA's beliefs (resulting from CA's recognition of the *Ask-Ref*) provides evidence that the applicability conditions of the *Answer-Ref* discourse act are satisfied.

Since this inference chain is the only one containing an e-action for which there is evidence, the system recognizes CA's utterance as providing Architecture as the answer to EA's question about what Dr. Smith is teaching and thereby contributing to the *Obtain-Info-Ref* action initiated by EA. The updated discourse tree is shown in Figure 14, with the new focus of attention marked with an asterisk.

Utterance 20: Initial Expression of Doubt

The system is again playing the role of CA (listener) and must understand EA's utterance of (20). The semantic representation of (20) is:

```
Surface-Neg-YN-Question(EA, CA, Teaches(Dr.Brown, Arch))
```

The surface form of (20) suggests that EA *thinks* that Dr. Brown is teaching Architecture, but is not certain of it. This belief is captured in the applicability condition of the recipe for a *Surface-Neg-YN-Question*. Since we assume a noise-free medium and well-formed utterances, surface speech acts always execute successfully and are correctly recognized. Thus the beliefs captured in the applicability conditions of the surface speech act are immediately entered into the system's model of EA's beliefs. The most salient interpretation of (20), that it is addressing the understanding of (19) and thus contributing to the *Tell* discourse act that is the current focus of attention in the dialogue, is rejected.[25]

25 In our implemented system, it is rejected because there is no recipe for the *Address-Understanding* action that is part of the body of the recipe for the *Tell* discourse act, and thus it is not possible to construct an inference path from the utterance to the *Address-Understanding* act. In the future, our expanded system will include such recipes, and the interpretation will be rejected because of lack of evidence for the e-action on the inference path or because its constraints are not satisfied.

The system can construct an inference path suggesting that the utterance contributes to the *Inform* discourse act that is the parent of the *Tell* act in the existing discourse tree. In particular, by chaining from subactions to parent actions (actions whose recipes contain the subaction), the system constructs an inference path containing the following chain of actions:

```
                  Inform(CA, EA, _proposition1)
                                ↑
          Address-Believability(CA, EA, _proposition1)
                                ↑
Address-Unacceptance(EA, CA, _proposition1, Teaches(Dr.Brown, Arch))
                                ↑
   Express-Doubt(EA, CA, _proposition1, Teaches(Dr.Brown, Arch))
                                ↑
     Convey-Uncertain-Belief(EA, CA, Teaches(Dr.Brown, Arch))
                                ↑
     Surface-Neg-YN-Question(EA, CA, Teaches(Dr.Brown, Arch))
```

If the last action on this inference path is unified with the *Inform act* in the existing discourse tree, then _proposition1 in the recipe for *Address-Believability* will be instantiated as Teaches(Dr.Smith, Arch), indicating that EA uttered (20) in order to express doubt at the proposition that Dr. Smith is teaching Architecture and thereby contribute to addressing the believability of that proposition. This interpretation would indicate that EA had passed up the opportunity to contribute to the *Tell* discourse act that is the focus of attention in the existing discourse tree. Thus when the system considers this interpretation, it hypothesizes that the *Tell* act has been successful and that its goal has been achieved, and it tentatively adds

```
believe(EA, believe(CA, Teaches(Dr.Smith, Arch), [C:C]), [C:C])
```

to the belief model.

As we have seen previously, *Express-Doubt* is an e-action since it is the action on the inference path at which the parameter _proposition1 is first introduced. Therefore, although the applicability conditions for each of the actions on the above inference path are plausible, we need evidence for the *Express-Doubt* act. There is no linguistic clue suggesting that (20) is an *Express-Doubt* action. The system then checks to see if it has evidence that the applicability conditions for the *Express Doubt* action hold. The applicability conditions are:

```
believe(EA, believe(CA, Teaches(Dr. Smith, Arch), [S:C]), [S:C])
believe(EA, Teaches(Dr. Brown, Arch), [W:S])
believe(EA, Teaches(Dr. Brown, Arch) → ¬Teaches(Dr. Smith, Arch), [S:C])
```

The system's belief model provides evidence for the first applicability condition, that EA believes that CA believes that Dr. Smith teaches Architecture, since it has been tentatively updated to include the goal of the *Tell* discourse act, as noted above. The belief model also provides evidence for the second applicability condition, since it has been updated to include the beliefs captured in the applicability conditions of the recipe for the surface speech act. The system's model of a stereotypical user contains the beliefs given in Figure 11, including the belief that there is only one professor per course. This stereotypical belief provides evidence for the final applicability condition (that EA believes that Dr. Brown teaching Architecture implies that Dr. Smith is *not* teaching Architecture). Since users typically believe that only one teacher is used per course, perhaps EA does also. If EA believes that there is only one professor per course and that Dr. Brown is teaching Architecture, then EA would believe that Dr. Smith would not be teaching Architecture. So the system has evidence for all three of the applicability conditions in the *Express-Doubt* recipe. In addition, the constraint of the *Express-Doubt* action is satisfied since the proposition that Dr. Smith teaches Architecture is a parameter of an action on the active path and thus is salient.

Since there is evidence from world and contextual knowledge and the surface form of the utterance that the applicability conditions hold for interpreting (20) as an expression of doubt and since there is no evidence for any other e-action, the system infers that this is the correct interpretation and stops. Thus, (20) is interpreted as an *Express-Doubt* action, as shown in Figure 15.

Utterances 21-22: Attempted Resolution of Conflict

The system is now playing the role of EA (listener) and must assimilate CA's utterances (21)-(22). The semantic representation of (21) is:

```
Surface-Say-Prop(CA, EA, ¬Teaches(Dr.Brown, Architecture))
```

Plan chaining indicates that the *Surface-Say-Prop* may be part of `Tell(CA, EA, ¬Teaches(Dr.Brown, Architecture))` which might be part of `Inform(CA, EA, ¬Teaches(Dr.Brown, Architecture))` which might be part of `Resolve-Conflict(CA, EA, _proposition1, _proposition2)` which might in turn be part of `Address-Unacceptance(EA, CA, _proposition1, _proposition2)`. If this is the *Address-Unacceptance* action that is part of the existing discourse tree in Figure 15, then the *Express-Doubt* and *Convey-Uncertain-Belief* actions in Figure 15 have completed successfully. Thus in considering this interpretation, the system hypothesizes that these actions have been successful and tentatively updates its belief model to reflect the effects and goals of these actions. In particular, the following two beliefs (among others) are tentatively added to the system's model of CA's beliefs:

```
believe(CA, believe(EA, Teaches(Dr.Brown,Arch)→¬Teaches(Dr.Smith,Arch),
                         [S:C]), [S:C])
believe(CA, believe(EA,Teaches(Dr.Brown,Arch),[W:S]), [S:C])
```

Resolve-Conflict (see the Appendix) is an e-action since it introduces two new propositions (the propositions about which there is conflict) that cannot be instantiated by chaining from the *Inform* action in its body, and the system must be able to determine what conflict the utterance is trying to resolve. If the *Address-Unacceptance* action is unified with the *Address-Unacceptance* that is part of the existing discourse tree, then the conflicting propositions, `_proposition1` and `_proposition2`, are instantiated as `Teaches(Dr.Smith, Arch)` and `Teaches(Dr.Brown, Arch)` respectively in both *Address-Unacceptance* and *Resolve-Conflict*. The system has evidence for the *Resolve-Conflict* action with these instantiations. The constraints that `_proposition1` and `_proposition2` be salient and that `_proposition2` and `_proposition3` be the opposite of one another are satisfied. First, `Teaches(Dr.Smith, Arch)` and `Teaches(Dr.Brown, Arch)` are the propositions instantiating `_proposition1` and `_proposition2`, and they are salient since they are part of an action on the active path of the existing discourse tree. Second, the proposition instantiating `_proposition2` is the opposite of the proposition conveyed by CA's current utterance. Evidence for the first two applicability conditions, 1) that CA believes that EA believes that Dr. Brown's teaching Architecture implies that Dr. Smith is *not* teaching Architecture and 2) that CA believes that EA has an uncertain belief in the proposition that Dr. Brown teaches Architecture, is provided by

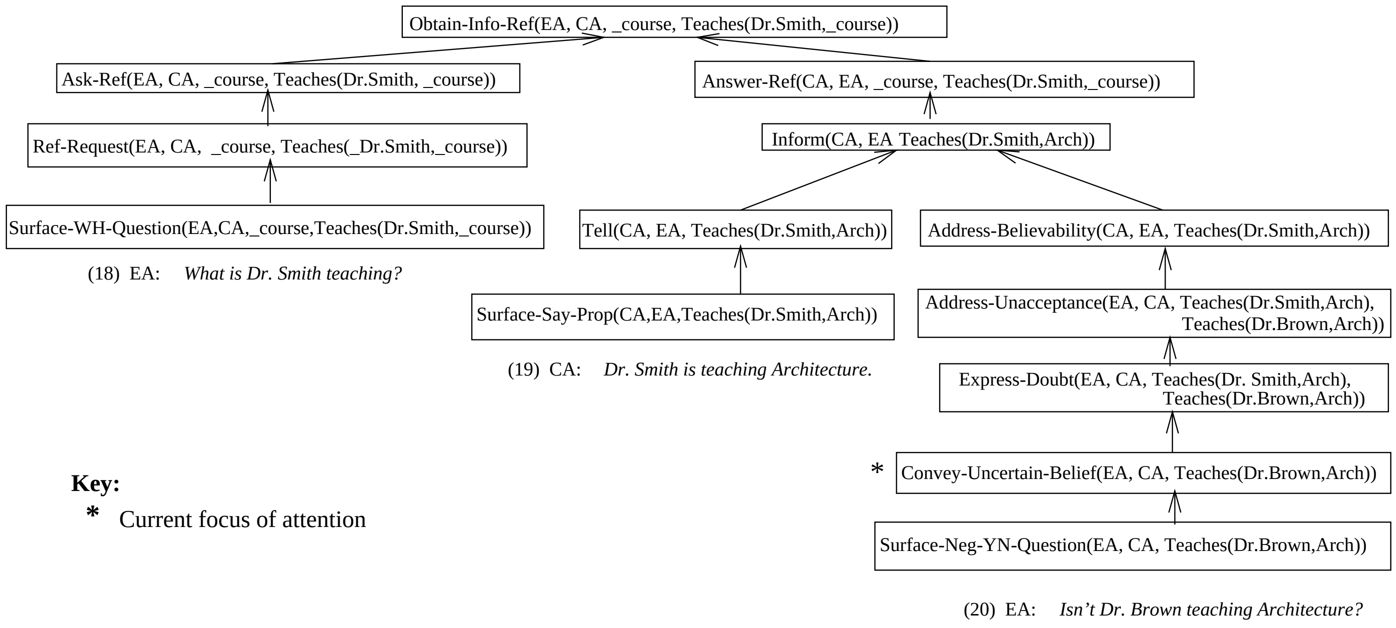

Figure 15
Discourse Tree for Utterances (18) - (20)

the system's tentatively updated model of CA's beliefs. Evidence for the final applicability condition, that CA believes that Dr. Smith is teaching Architecture, is also provided by the system's model of CA's beliefs. When CA's *Inform* action in (19) was recognized, the system updated its model of CA's beliefs to include the beliefs contained in the applicability conditions for the *Inform* act; thus the belief model indicates that CA believes that Dr. Smith is teaching Architecture. Since the system has evidence for the e-action on the inference path (and since there are no other inference paths containing e-actions), the system recognizes this chain of actions and interprets (21) as informing EA that Dr. Brown is not teaching Architecture as part of attempting to resolve the conflict suggested by EA. Thus the *Resolve-Conflict* action is recognized as contributing to the *Address-Unacceptance* action that was begun in (20).

The semantic representation of (22) is:

```
Surface-Say-Prop(CA, EA, on-sabbatical(Dr.Brown))
```

The *Surface-Say-Prop* is part of `Tell(CA, EA, on-sabbatical(Dr.Brown))` which is part of `Inform(CA, EA, on-sabbatical(Dr.Brown))`. Further chaining suggests that the *Inform* action could be part of several other actions. We will discuss two of these possibilities, *Address-Acceptance* and *Explain-Claim*. In the *Address-Acceptance* case, CA might be uttering (22) to support the statement that she made in (21); in the *Explain-Claim* case, CA might be uttering (22) to explain why the supposedly conflicting propositions are not really in conflict.

Let us examine the *Address-Acceptance* case first. `Address-Acceptance(CA, EA, _proposition1)` is part of a recipe for `Address-Believability(CA, EA, _proposition1)`, which in turn is part of a recipe for `Inform(CA, EA, _proposition1)`. If this is CA's immediately preceding *Inform* act, then unifying with this *Inform* act will cause `_proposition1` to be instantiated as `¬Teaches(Dr.Brown, Arch)` in the *Inform*, *Address-Believability*, and *Address-Acceptance* actions. *Address-Acceptance* is an e-action, since it is the action on the inference path at which a new proposition is first introduced. If `_proposition3` in the recipe for *Address-Acceptance* (see the Appendix) is instantiated with `Teaches(Dr.Brown, Arch)`, then the constraints are obviously satisfied. (Note that `_proposition2` in the recipe for *Address-Acceptance* is instantiated with `on-sabbatical(Dr.Brown)` as a result of chaining from the surface speech act to the *Inform* act in the body of the *Address-Acceptance* action.) The system has evidence that the applicability conditions are satisfied with these instantiations. Evidence for the first applicability condition is provided by the system's model of a stereotypical user, which indicates that it is generally believed that professors on sabbatical do not teach. Evidence for the second applicability condition is provided by the system's model of CA's beliefs, which was updated to contain the effect of utterance (20)'s *Convey-Uncertain-Belief* action – namely, that CA believes that EA has some belief that Dr. Brown is teaching Architecture. Thus there is evidence for recognizing CA's utterance as addressing acceptance of the proposition that Dr. Brown is not teaching Architecture by offering support for it.

Now let us examine the *Explain-Claim* case. `Explain-Claim(CA, EA, _proposition1, _proposition2)` is part of `Resolve-Conflict(CA, EA, _proposition1, _proposition2)` and unifying with the *Resolve-Conflict* that is already part of the existing discourse tree causes `_proposition1` and `_proposition2` in the *Resolve-Conflict* and *Explain-Claim* actions to be instantiated with `Teaches(Dr.Smith, Arch)` and `Teaches(Dr.Brown, Arch)` respectively. `_proposition3` in the recipe for *Explain-Claim* has been instantiated with `on-sabbatical(Dr.Brown)` by chaining from the surface speech act to the *Explain-Claim* action. *Explain-Claim* is an e-action. However, the system lacks evidence for its second applicability condition, that CA believes that Dr. Brown being on

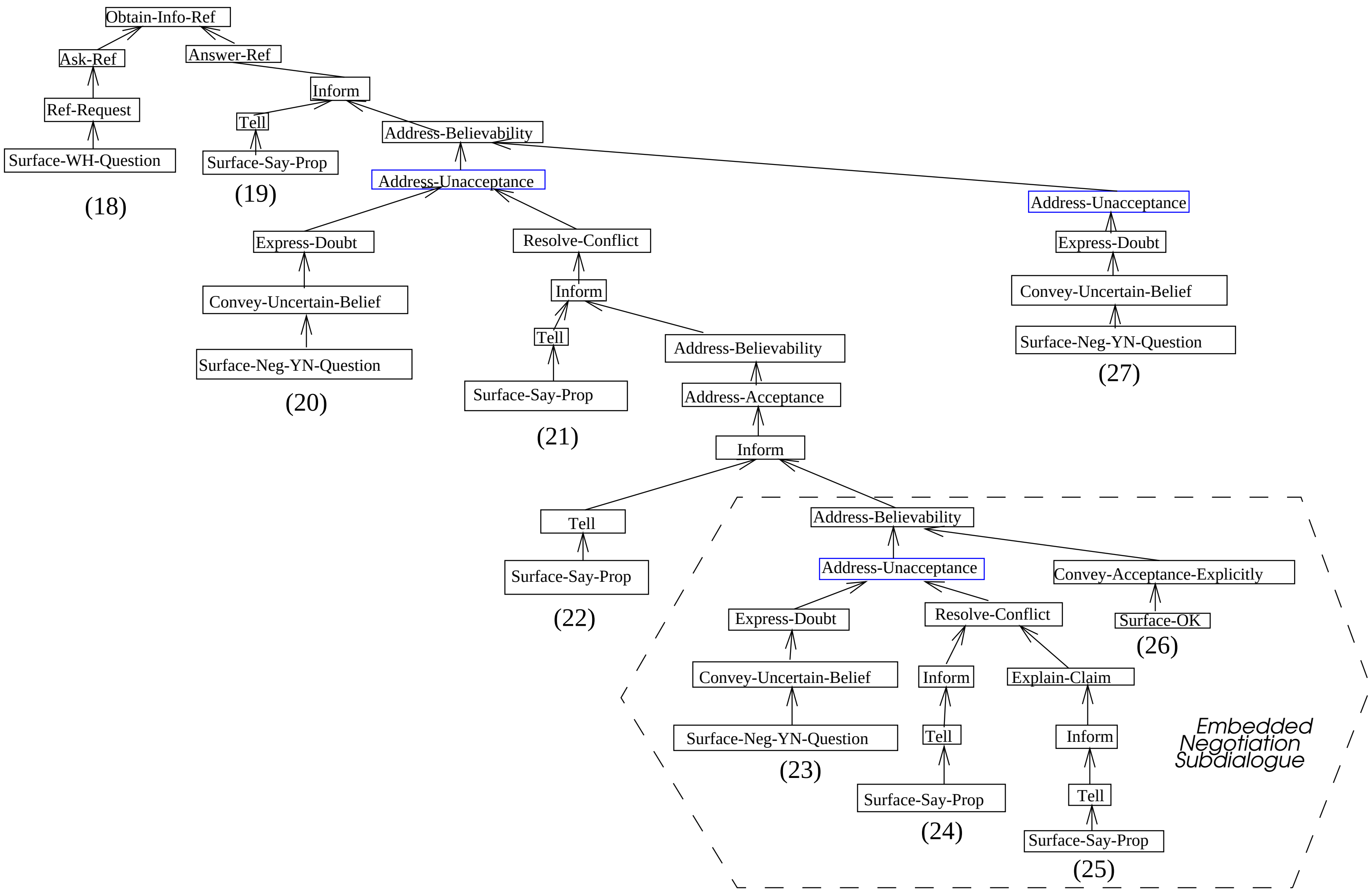

Figure 16
Discourse Tree for Dialogue in Figure 12

sabbatical implies that Dr. Brown teaching Architecture and Dr. Smith teaching Architecture are not in conflict with one another. Thus this potential interpretation is rejected. Since the inference path containing the *Address-Acceptance* discourse act is the only one whose e-action has evidence supporting its recognition, the system recognizes (22) as addressing the acceptance of the proposition conveyed by (21) — namely, that Dr. Brown is not teaching Architecture. Thus, CA's response in (21) and (22) indicates that CA is trying to resolve EA's and CA's conflicting beliefs. The structure of the discourse tree after these utterances is shown in Figure 16, above the numbers (18) - (22).[26]

Utterances 23-26: Embedded Negotiation Subdialogue

The system is now playing the role of CA (listener) and must assimilate EA's utterance of (23). The semantic representation of (23) is

```
Surface-Neg-YN-Question(EA, CA, on-campus(Dr.Brown, Yesterday))
Clueword(But)
```

The *Surface-Neg-YN-Question* in utterance (23) is one way to *Convey-Uncertain-Belief*. The linguistic clue *but* suggests that EA is executing a non-acceptance discourse action; this non-acceptance action might be addressing (22), (21) or (19) since the propositions conveyed by these utterances have not yet been accepted by EA and are thus open for rejection. Let us consider the proposition conveyed by (22) since it is the most salient open proposition at this point in the dialogue and thus the most expected candidate. Plan chaining suggests that the *Convey-Uncertain-Belief* could be part of an *Express-Doubt* action, which in turn could be part of an *Address-Unacceptance* action, which could be part of an *Address-Believability* action, which could be part of the *Inform* action in (22). As in utterance (20), there is evidence that the applicability conditions for the e-action (the *Express-Doubt* action) hold: for example, world knowledge indicates that a typical user believes that professors who are on campus are not on sabbatical, providing evidence for the third applicability condition. Thus, there is both linguistic evidence for a generic non-acceptance discourse act and evidence from world and contextual knowledge and the surface form of the utterance that the applicability conditions and constraints are satisfied for the specific action of expressing doubt at the proposition that Dr. Brown is on sabbatical. Since no other e-action has both kinds of evidence, (23) is interpreted as expressing doubt at the proposition conveyed by (22).

The system now reverts to playing the role of EA (listener) and must assimilate the next two utterances in which CA resolves the doubt that EA has expressed in (23), by agreeing that Dr. Brown was on campus yesterday but explaining the purpose of his visit (one that is an exception to the rule that people on sabbatical are not on campus). Plan inferencing for utterance (24) is identical to that of utterance (21) and will not be described further.

From the *Surface-Say-Prop* in (25), plan inference rules suggest that the *Surface-Say-Prop* is part of a *Tell* action that is part of an *Inform* action. As was the case for utterance (22), the *Inform* action can be part of several different higher-level actions, including *Address-Acceptance* and *Explain-Claim*. Since *Address-Acceptance* is a subaction in a recipe for *Address-Believability* and *Address-Believability* is a subaction in a recipe for *Inform*, CA might be trying to offer support for the *Inform* act of (24), `Inform(CA, EA, on-campus(Dr.Brown, Yesterday))`. However, this time the applicability conditions for the *Address-Acceptance* action are implausible. In particular, as a result of the effect of the *Convey-Uncertain-Belief* action in (23), the system's model of CA's

26 For space reasons, only the action names are shown.

beliefs indicates that CA believes that EA has some belief in the proposition that Dr. Brown was on campus yesterday. The second applicability condition of the recipe for addressing the acceptance of the proposition conveyed by (24), `believe(CA, believe(EA, ¬on-campus(Dr.Brown, Yesterday), [W:S]), [W:C])`, conflicts with this belief – i.e., *Address-Acceptance* is reasonable to pursue only when an agent has some reason to believe that the listener disbelieves the proposition in question. Since the second applicability condition is implausible, the inference path containing the *Address-Acceptance* action is rejected.

However, the system does have evidence for interpreting (25) as an *Explain-Claim*. `Explain-Claim(CA, EA, _proposition1, _proposition2)` is part of the recipe for `Resolve-Conflict(CA, EA, _proposition1, _proposition2)`. If this is the *Resolve-Conflict* action that is closest to the focus of attention in the existing discourse tree, then unification will cause `_proposition1` and `_proposition2` in *Resolve-Conflict* and *Explain-Claim* to be instantiated respectively with `on-sabbatical(Dr.Brown)` and `on-campus(Dr. Brown, Yesterday)`. `_proposition3` in the recipe for *Explain-Claim* was instantiated with `Give(Dr.Brown, University-Colloquium)` during chaining from the surface speech act. The system has evidence for the e-action *Explain-Claim* because it has evidence that its applicability conditions hold (i.e., that CA believes that EA believes that Dr. Brown's being on campus implies that he is not on sabbatical from the effect of the *Express-Doubt* action; that CA believes that Dr. Brown's giving a University colloquium implies that being on campus is not in conflict with being on sabbatical, from the model of stereotypical beliefs; and that CA believes that EA believes that Dr. Brown was on campus yesterday, from the effect of the *Convey-Uncertain-Belief* discourse act accomplished by (23)). Since this is the only inference path containing an e-action for which the system has evidence, utterance (25) is interpreted as contributing to resolving the conflict suggested in (23) by explaining the claim that the propositions do not really conflict in this instance.

The system now reverts to playing the role of CA (listener) and must assimilate EA's utterances. In (26), EA indicates explicit acceptance of the most salient *Inform* action, so the system is able to determine that EA has accepted CA's response in (25). Other inform actions remain open for rejection and must still be implicitly or explicitly accepted. In this dialogue, the *Inform* actions in (22) and (21) are implicitly accepted in utterance (27). Although utterance (27) might cause one to hypothesize that (26) was indicating explicit acceptance of all of the propositions conveyed by utterances (21)-(25), it is not possible to decide with certainty from a simple "*ok*" exactly how many *Inform* actions EA is accepting. Thus our system assumes that the speaker accepts as little as possible, which is the most salient *Inform* action.

Utterances (23)-(26) illustrate our model's handling of negotiation subdialogues embedded within other negotiation subdialogues. The subtree contained within the dashed lines in Figure 16 shows the structure of this embedded negotiation subdialogue.

Utterance 27: Multiple Expressions of Doubt and Implicit Acceptance

The system is still playing the role of CA (listener). The semantic representation of EA's next utterance is

```
Surface-Neg-YN-Question(EA, CA, Specialty(Dr.Smith, Theory))
Clueword(But)
```

The linguistic clue *but* in (27) again suggests non-acceptance. Since (25) has been explicitly accepted, the propositions open for rejection are those conveyed in (22), (21), and (19). Once again, chaining from the surface speech act can produce a chain of actions containing an *Express-Doubt* action and terminating with one of the *Inform* actions that

(28)	EA:	*When does CS510 meet?*
(29)	CA:	*CS510 meets on Monday night at 7PM.*
(30)	EA:	*But isn't Dr. Jones teaching CS510?*
(31)	CA:	*No, Dr. Jones is not teaching CS510.*
(32)		*Dr. Hart is teaching CS510.*
(33)	EA:	*But isn't CS510 a graduate course?*
(34)	CA:	*Yes, CS510 is a graduate course.*
(35)		*Dr. Hart teaches both graduate and undergraduate courses.*
(36)	EA:	*What courses are prerequisites for CS510?*

Figure 17
A Second Negotiation Subdialogue

is on the active path of the existing discourse tree. If the *Inform* action is `Inform(CA, EA, Teaches(Dr.Smith, Arch))`, then the *Express-Doubt* action will be instantiated as `Express-Doubt(EA, CA, Teaches(Dr.Smith,Arch), Specialty(Dr.Smith,Theory))`. The system has evidence that this action's applicability conditions are satisfied. The evidence for the first two applicability conditions is similar to the evidence for interpreting utterance (20) as expressing doubt. World knowledge provides evidence for the third applicability condition. The system's model of stereotypical user beliefs indicates that it is typically believed that faculty only teach courses in their field. Other system knowledge states that Architecture and Theory are different fields. So in this case, the system's world knowledge provides evidence that Dr. Smith's being a theory person is an indication to the user that Dr. Smith does not teach Architecture. Thus the system has two kinds of evidence for interpreting (27) as expressing doubt at the proposition conveyed by (19): linguistic evidence for a generic *Express-Doubt* discourse act and evidence that the applicability conditions are satisfied for the particular discourse act of expressing doubt at the proposition that Dr. Smith is teaching Architecture. Since the system does not have multiple evidence for any of the other interpretations, the system recognizes (27) as again expressing doubt about the proposition conveyed by (19). Thus, the system is able to recognize and assimilate a second expression of doubt at the proposition conveyed in (19), even after intervening dialogue. The discourse tree for the entire dialogue is given in Figure 16.

Since EA's utterance reverts back to addressing the acceptance of the proposition conveyed by (19), EA has foregone the opportunity to challenge the claims made in utterances (22) and (21). Since the belief model indicates that the applicability conditions of the *Inform* actions are still satisfied (except those negated by achievement of the goal), the system infers that EA has implicitly accepted the statements in (22) and (21), that Dr. Brown is on sabbatical and that Dr. Brown is not teaching Architecture, and the system updates its model of EA's beliefs.

5.3 A Second Example

Figure 17 contains a second negotiation dialogue. Due to space limitations, we will only discuss two interesting features of the dialogue and its processing by our system. Utterance (30) illustrates the use of a linguistic clue word to suggest an expression of doubt. In interpreting utterance (30), the system constructs an inference path containing the action

```
Express-Doubt(EA, CA, Meets(CS510, Mon7PM), Teaches(Dr.Jones, CS510))
```

Although the system does not have evidence that all of the applicability conditions for this *Express-Doubt* action are satisfied, the linguistic clue *But* does provide evidence for the generic *Express-Doubt* act. Since this is the only inference path containing an e-action for which there is evidence, the system recognizes (30) as expressing doubt at the proposition that CS510 meets on Monday at 7PM by contending that Dr. Jones is teaching CS510. In this case, the system lacked evidence for the third applicability condition in the recipe for *Express-Doubt*. But having recognized that EA is expressing doubt, it attributes to EA the beliefs captured in the applicability conditions. In particular, the system attributes to EA the belief that Dr. Jones teaching CS510 implies that CS510 would not meet on Monday at 7PM, though it has no idea why EA believes that this implication holds – perhaps EA believes that Dr. Jones has to be home to take care of his children at night.

When utterance (33) occurs, there are three propositions that have not yet been accepted by EA, and the system considers the possibility that EA is performing one of three express doubt actions, namely

```
Express-Doubt(EA, CA, Meets(CS510, Mon7PM), Graduate-Course(CS510))
Express-Doubt(EA, CA, ¬Teaches(Dr.Jones, CS510), Graduate-Course(CS510))
Express-Doubt(EA, CA, Teaches(Dr.Hart, CS510), Graduate-Course(CS510))
```

In all three cases, the system lacks evidence that the third applicability condition in the *Express-Doubt* recipe is satisfied. However, the linguistic clue word *But* provides evidence for a generic *Express-Doubt* action. Since the system has equivalent evidence for all three of the *Express-Doubt* acts, contextual knowledge is used to choose among them. Since the proposition `Teaches(Dr.Hart, CS510)` is closest to the existing focus of attention in the discourse tree, it is the most salient of the three propositions that are open for rejection. Utterance (33) is therefore interpreted as expressing doubt at the proposition that Dr. Hart is teaching CS510 by contending that it is a graduate-level course. Thus contextual knowledge arbitrates when equivalent evidence is available for several specific discourse acts.

6 Evaluation and Future Work

We undertook an evaluation of our prototype system both to assess whether it derived appropriate interpretations of utterances and to identify areas for further research. We obtained eight human volunteers, six of whom are not engaged in NLP research and two of whom are involved in unrelated NLP projects. The subjects were given a set of world knowledge stereotypically believed in the domain, such as that faculty on sabbatical do not normally teach. The subjects were presented with a set of dialogues and asked to analyze several utterances from each dialogue. The selected utterances did not include simple questions initiating the dialogue or straightforward answers to questions, since it seemed likely that the subjects would agree with the system's interpretation and thus the results would be biased in favor of the system. The selected utterances did include surface-negative questions (both with and without a clue word of *"but"*), statements interpreted by our system as support for a previous assertion or as explanations about why a proposition was not in conflict with a previous claim, and examples of implicit acceptance.

For each utterance selected for analysis, the subjects were given a suggested interpretation,[27] and asked whether the suggested interpretation was reasonable and whether

27 The subjects were not told that the suggested interpretation was the one produced by our system but only that we were trying to determine how utterances in a discourse should be interpreted.

they could identify a better interpretation. For 15 of 20 utterances, the subjects unanimously believed that the system's interpretation was best. It should be noted there was unanimous agreement that utterance (42) below should be interpreted as an expression of doubt but that utterance (39) should not.

Dialogue-A:	(37)	EA:	*Who is teaching architecture?*
	(38)	CA:	*Dr. Smith is teaching architecture.*
	(39)	EA:	*Isn't Dr. Smith an excellent teacher?*
Dialogue-B:	(40)	EA:	*Who is teaching architecture?*
	(41)	CA:	*Dr. Smith is teaching architecture.*
	(42)	EA:	*But isn't Dr. Smith an excellent teacher?*

There were two categories of utterances where the subjects disagreed. In the case of surface-negative questions that did not express doubt, such as utterance (39) above, the suggested interpretation given to the subjects was that the speaker was seeking information about whether the queried proposition was true. When the subjects did not interpret the utterance as an expression of doubt (see below), five of them contended that a better interpretation would be that EA was seeking verification of the queried proposition. Since our system already recognizes from the surface-negative question that the speaker has a strong (but uncertain) belief in the queried proposition, it is easy to extend our system so that it can explicitly identify a *Seek-Verification* discourse act.

The other category for which there was disagreement was surface-negative questions where a clue word was not present and the stereotypical domain knowledge did not provide a conflict. In two of five instances, some subjects used their own experience to identify a mutual belief that might suggest a conflict, such as the belief that sometimes certain faculty are not allowed to teach graduate level courses. While this knowledge cannot be captured as a default rule, it does represent a kind of shared experiential knowledge that would provide weak evidence for a potential conflict. However, it should be noted that our subjects were split on how these problematic cases should be interpreted, agreeing with the system's interpretation slightly more than half the time. There was also another such surface-negative question where one subject viewed the system's interpretation as reasonable but argued that an expression of doubt would be a better interpretation. In order to derive this interpretation, the subject posited an attribute for the speaker that was neither evident from the dialogue nor stereotypically true. (The other subjects agreed that the system's interpretation was best.) These examples bear on the issue of accommodation mentioned in Section 4.5.1, since one could argue that the subjects who interpreted the utterances as expressions of doubt were trying to accommodate an incompatibility. This is particularly true in the last instance where the subject found it necessary to resort to non-shared knowledge in making the interpretation. However, it is unclear whether a speaker would expect a listener to recognize such utterances as expressions of doubt without additional clues. As noted below, our future research will consider other forms of evidence (gestural and intonational) in order to resolve such ambiguous utterances.

After they had finished analyzing the dialogues, we asked the subjects to construct three dialogues containing an expression of doubt and to explain why the expression of doubt should be interpreted as such. While these dialogues provided no contradictions to our approach, they did provide a couple of interesting examples, such as the following dialogue, that suggest areas for future work.

(43) EA: *We have basil, parsley, and oregano, but we need marjoram.*
(44) CA *Isn't marjoram the same as oregano?*

Clearly (44) is expressing doubt at the claim conveyed by (43), but it relies on shared world knowledge that if a list contains X items, the X items are presumed to be different. Our system does not currently include such knowledge.

Our subjects commented that intonation and facial gesture might alter their interpretation of the utterances in the dialogues; we are beginning research that will take these kinds of evidence into account(Carberry et al., 1996). In addition, we will be expanding the kinds of world knowledge incorporated into our system, and will be considering both the strength of different pieces of evidence and how several pieces of weak evidence affect interpretation. We would also like to extend our use of linguistic clues to include a wide variety of clue words and phrases and to recognize the functions that these words can play. In addition, we are developing a plan-based response generation component(Chu-Carroll and Carberry, 1994). Initial work on this component includes a subsystem that can identify what evidence to present to a user when conflicts arise(Chu-Carroll and Carberry, 1995b; Chu-Carroll and Carberry, 1998) and what information to request when the system cannot rationally decide whether to accept a proposition conveyed by the user(Chu-Carroll and Carberry, 1995a; Chu-Carroll and Carberry, 1998).

We will also be investigating the scale-up of our system as we extend its coverage. Part of the motivation for the content of the current discourse recipes was their future extension to other domains, such as tutoring. For example, as discussed in Section 4.2.1, the formulation of our *Ask-Ref* recipe allows it to be used as a subaction of a future *Test-Knowledge* discourse act since the recipe does not presume that the speaker is ignorant about the correct value of the requested term. This should aid in extending the kinds of discourse acts that can be handled. Although transporting our system to another domain will require encoding new domain knowledge and new domain recipes, the recipes for discourse and problem-solving acts are domain-independent and thus will remain unchanged. Moreover, the knowledge captured in our recipes is communicative knowledge shared by dialogue participants; we believe that such communicative knowledge (such as how to express doubt) is finite although the possible intentions (such as the intention of expressing doubt at Dr. Smith teaching CS360) are infinite.

7 Other Related Work

7.1 Grosz and Sidner's Theory of Discourse Processing

Grosz and Sidner(Grosz and Sidner, 1986) postulated a theory of discourse structure that included linguistic, intentional, and attentional components, and they argued that the dominance and satisfaction-precedes relationships between discourse segments must be identified in order to determine discourse structure. They also noted three kinds of information that contribute to determining the purposes of discourse segments and their relationship to one another: linguistic markers, utterance-level intentions, and general knowledge about actions and objects. Subsequently Lochbaum(Lochbaum, 1994) developed an algorithm based on Grosz and Sidner's SharedPlan model(Grosz and Sidner, 1990) that recognizes discourse segment purposes and discourse structure.

We contend that, in order to understand utterances and respond appropriately, it is necessary not only to determine the structure of the discourse but also to identify the communicative acts that an agent intends to perform with an utterance.[28] For example, if a listener does not recognize when an utterance such as *"Wasn't Dr. Smith on campus yesterday?"* is expressing doubt, then the listener's response might fail to address the rea-

28 In a dialogue, Grosz and Sidner's discourse segment purpose is intended to capture the purpose of a segment consisting of a series of utterances by both participants, not the communicative intentions underlying each participant's discourse actions.

sons for this doubt. Our research provides a computational algorithm that uses multiple knowledge sources to recognize complex discourse acts, including expressions of doubt, and to identify their relationship to one another. This algorithm and our strategy for recognizing implicit acceptance enable us to model negotiation subdialogues, something that previous systems have been unable to handle.

7.2 Argument Understanding Systems

Several researchers have built argument understanding systems, but none has addressed participants coming to an agreement or mutual belief about a particular situation, either because the researchers investigated monologues only (Cohen, 1987; Cohen and Young, 1991), or because they assumed that dialogue participants do not change their minds (Flowers, McGuire, and Birnbaum, 1982; Quilici, 1991). Cohen (Cohen, 1987) developed an argument understanding system that used clue words and an evidence oracle to build a discourse structure for arguments based on which utterances served as support for other utterances. Cohen's model, however, handles only monologues, so responses to arguments are not modeled in her system. Birnbaum, Flowers and Dyer (Flowers and Dyer, 1984; McGuire, Birnbaum, and Flowers, 1981; Birnbaum, Flowers, and McGuire, 1980) developed a system that finds flaws in arguments and determines how to respond. Quilici (Quilici, 1991) created a system in which agents respond to each other's arguments based on a justification pattern that will support the agent's position. Both Quilici and Birnbaum et al., however, assume that all participants in an argument will retain their opinion throughout the course of the argument, and concentrate mainly on how to find flaws in arguments and construct responses based on those findings; they do not address actually winning arguments. Reichman (Reichman, 1981) modeled informal debates by using her idea of context spaces and expectations to determine who should respond and what possible topics might be addressed. However, she does not provide a detailed computational mechanism for recognizing the role of each utterance in a debate.

7.3 Models of Collaborative Behavior

Several models of discourse have recently been built which view conversation as a kind of collaborative behavior in which speakers try to make themselves understood and listeners work with speakers to help speakers attain this goal.

Clark and Schaefer (Clark and Schaefer, 1989) contend that utterances must be *grounded*, or understood, by both parties, but they do not address conflicts in belief, only lack of understanding. Walker (Walker, 1992) has found many occasions of redundancy in collaborative dialogues, and explains these by claiming that people repeat themselves in order to ensure that each utterance has been understood.[29] Clark and Wilkes-Gibbs (Clark and Wilkes-Gibbs, 1990) propose a collaborative model of dialogue in which referring is viewed as a collaborative process and each conversation unit is viewed as a contribution, which consists of 1) an utterance that performs a referring action, and 2) the utterances required to understand the referent described in the utterance. Heeman (Heeman, 1991) implemented this model in a plan-based collaborative model of dialogue which is able to plan and recognize referring expressions and their corrections.

Other collaborative models assume that two participants are working together to achieve a common goal (Cohen and Levesque, 1990a; Cohen and Levesque, 1991a; Cohen and Levesque, 1991b; Lochbaum, Grosz, and Sidner, 1990; Lochbaum, 1991; Grosz and Sidner, 1990; Searle, 1990). Searle (Searle, 1990) proposes a model in which the two agents

29 Another reason for repetition, she claims, is for centering(Grosz, Joshi, and Weinstein, 1995), but she concentrates on repetitions that give evidence of understanding.

working together have a joint intention, a *we intention* instead of individual intentions. Cohen and Levesque (Cohen and Levesque, 1990a; Cohen and Levesque, 1990b; Cohen and Levesque, 1990c; Cohen and Levesque, 1991a; Cohen and Levesque, 1991b) have developed a formal theory in which agents are jointly committed to accomplishing a goal, so both parties have individual intentions to accomplish the goal as part of their joint commitment. Grosz, Lochbaum, and Sidner (Grosz and Sidner, 1990; Lochbaum, Grosz, and Sidner, 1990; Lochbaum, 1991) have specified a system in which two agents are working to accomplish some common goal by building a *shared plan* in which each agent holds certain beliefs and intentions. These beliefs and intentions indicate that the agents intend to perform some joint action, and that they believe they can perform this action. All of these models indicate the need for modeling collaborative dialogue, but none suggests a system that can handle the kind of negotiation subdialogues that people often engage in when trying to negotiate their conflicts in belief, even when they are both working towards the same goal.

8 Conclusion

We have presented a plan-based model for handling cooperative negotiation subdialogues. Our system infers both the communicative actions that people pursue when speaking and the beliefs underlying these actions. Beliefs, and the strength of these beliefs, are recognized from the surface form of utterances and from the explicit and implicit acceptance of previous utterances. Our algorithm for recognizing discourse actions combines linguistic, contextual, and world knowledge in a unified framework. By combining these different knowledge sources, we are able to recognize complex discourse acts such as expressing doubt, to identify the relationship of utterances to one another, and to model negotiation subdialogues. Since negotiation is an integral part of multi-agent activity, our process model addresses an important aspect of cooperative interaction and thus is a step toward an intelligent and robust natural language consultation system.

Acknowledgments
This work was supported by the National Science Foundation under Grant No. IRI-9122026. The Government has certain rights in this material. We would like to thank Rachel Sacher for her help in our corpus analysis and the anonymous reviewers for their helpful comments on the manuscript.

9 Appendix

Discourse Recipe

Action: Address-Acceptance(_agent1, _agent2, _proposition1)
{*_agent1 tries to make _proposition1 believable to _agent2*}
Recipe-Type: Decomposition
Appl Cond: believe(_agent1, _proposition2 $\rightarrow$ $\neg$_proposition3, [S:C])
believe(_agent1, believe(_agent2, _proposition3, [W:S]), [W:C])
Constraints: opposite(_proposition1, _proposition3)
Body: Inform(_agent1, _agent2, _proposition2)
Effects: believe(_agent2, believe(_agent1, _proposition2 $\rightarrow$ _proposition1, [S:C]), [S:C])
Goal: enhance-believability(_agent1, _agent2, _proposition1)

Discourse Recipe

Action: Address-Believability(_agent1, _agent2, _proposition1)
{*_agent1 and _agent2 address the believability of _proposition1*}
Recipe-Type: Decomposition
Appl Cond: believe(_agent1, _proposition1, [C:C])
believe(_agent1, believe(_agent2, _proposition1, [C_N:S]), [0:C])
Body: #Address-Acceptance(_agent1, _agent2, _proposition1)
#Address-Unacceptance(_agent2, _agent1, _proposition1, _proposition2)
#Convey-Acceptance-Explicitly(_agent2, _agent1, _proposition1)
Effects: believability-addressed(_agent1, _agent2, _proposition1)
Goal: same-mutual-beliefs(_agent1, _agent2, _proposition1)

Discourse Recipe

Action: Address-Unacceptance(_agent1, _agent2, _proposition1, _proposition2)
{*By noting a conflicting _proposition2, _agent1 initiates negotiation of his unacceptance of _proposition1*}
Recipe-Type: Decomposition
Appl Cond: believe(_agent1, in-conflict(_proposition1, _proposition2), [W:C])
Body: Express-Doubt(_agent1, _agent2, _proposition1, _proposition2)
Resolve-Conflict(_agent2, _agent1, _proposition1, _proposition2)
Effects: unacceptance-addressed(_agent1, _agent2, _proposition1)
Goal: conflict-resolved(_proposition1, _proposition2)

Discourse Recipe

Action: Answer-Ref(_agent1, _agent2, _term, _proposition)
{*_agent1 answers _agent2's question about the referent of _term in _proposition*}
Recipe-Type: Decomposition
Appl Cond: believe(_agent1, want(_agent2, knowref(_agent2, _term, believe(_agent1, _proposition, [C:C]))), [W:C])
believe(_agent1, ¬knowref(_agent2, _term, believe(_agent1, _proposition, [C:C])), [W:C])
believe(_agent1, instantiates(_propanswer, _term, _proposition), [C:C])
Constraints: salient(_proposition)
Preconditions: question-accepted(_agent1, _agent2, _proposition)
believe(_agent1, knowref(_agent1, _term, _proposition), [C:C])
Body: Inform(_agent1, _agent2, _propanswer)
#Address-Answer-Acceptability(_agent1, _agent2, _propanswer)
Effects: believe(_agent2, answered-question(_agent1, _proposition), [W:C])
Goal: knowref(_agent2, _term, believe(_agent1, _proposition, [C:C]))

Discourse Recipe

Action: Ask-Ref(_agent1, _agent2, _term, _proposition)
{*_agent1 tries to get _agent2 to tell him the referent of the _term in _proposition*}
Recipe-Type: Decomposition
Appl Cond: want(_agent1, knowref(_agent1, _term, believe(_agent2, _proposition, [C:C])))
¬knowref(_agent1, _term, believe(_agent2, _proposition, [C:C]))
Constraints: term-in(_term, _proposition)
Body: Ref-Request(_agent1, _agent2, _term, _proposition)
#Make-Question-Acceptable(_agent1, _agent2, _proposition)
Effects: believe(_agent2, want(_agent1, Answer-Ref(_agent2, _agent1, _term, _proposition)), [C:C])
Goal: want(_agent2, Answer-Ref(_agent2, _agent1, _term, _proposition))

Discourse Recipe

Action: Convey-Uncertain-Belief(_agent1, _agent2, _proposition)
{*_agent1 conveys an uncertain belief in _proposition*}
Recipe-Type: Specialization
Body: Surface-Neg-YN-Question(_agent1, _agent2, _proposition)
Surface-Tag-Question(_agent1, _agent2 _proposition)
Effects: believe(_agent2, believe(_agent1, _proposition, [W:S]), [S:C])
Goal: believe(_agent2, believe(_agent1, _proposition, [W:S]), [S:C])

Discourse Recipe

Action: Explain-Claim(_agent1, _agent2, _proposition1, _proposition2)
{*_agent1 explains why _proposition1 and _proposition2 are not in conflict*}
Recipe-Type: Decomposition
Appl Cond: believe(_agent1, believe(_agent2, _proposition2 → ¬_proposition1, [S:C]), [W:C])
believe(_agent1, _proposition3 → ¬in-conflict(_proposition1, _proposition2), [S:C])
believe(_agent1, believe(_agent2, _proposition2, [W:C]), [S:C])
Constraints: salient(_proposition1)
salient(_proposition2)
Body: Inform(_agent1, _agent2, _proposition3)
Effects: claim-explained(_agent1, _agent2, _proposition1)
Goal: believe(_agent2, _proposition2 → ¬_proposition1, [C_N:C_N])

Discourse Recipe

Action: Express-Doubt(_agent1, _agent2, _proposition1, _proposition2)
{*_agent1 expresses doubt to _agent2 about _proposition1 by contending that _proposition2 is true*}
Recipe-Type: Decomposition
Appl Cond: believe(_agent1, believe(_agent2, _proposition1, [S:C]), [S:C])
believe(_agent1, _proposition2, [W:S])
believe(_agent1, _proposition2 → ¬_proposition1, [S:C])
Constraints: salient(_proposition1)
Body: Convey-Uncertain-Belief(_agent1, _agent2, _proposition2)
Effects: believe(_agent2, believe(_agent1, _proposition1, [S_N:W_N]), [S:C])
believe(_agent2, believe(_agent1, _proposition2 → ¬_proposition1, [S:C]), [S:C])
believe(_agent2, want(_agent1, Resolve-Conflict(_agent2, _agent1, _proposition1, _proposition2)), [S:C])
Goal: want(_agent2, Resolve-Conflict(_agent2, _agent1, _proposition1, _proposition2))

Discourse Recipe

Action: Inform(_agent1, _agent2, _proposition)
{*_agent1 informs _agent2 of _proposition*}
Recipe-Type: Decomposition
Appl Cond: believe(_agent1, _proposition, [C:C])
believe(_agent1, believe(_agent2, _proposition, [C_N:S]), [0:C])
Body: Tell(_agent1, _agent2, _proposition)
#Address-Believability(_agent1, _agent2, _proposition)
Effects: believe(_agent2, want(_agent1, believe(_agent2, _proposition, [C:C])), [C:C])
Goal: believe(_agent2, _proposition, [C:C])

Discourse Recipe
Action: Obtain-Info-Ref(_agent1, _agent2, _term, _proposition)
{*_agent1 learns from _agent2 the referent of _term in _proposition)*}
Recipe-type: Decomposition
Appl Cond: believe(_agent1, knowref(_agent2, _term, _proposition), [W:C])
¬knowref(_agent1, _term, _proposition)
want(_agent1, knowref(_agent1, _term, _proposition))
Constraints: term-in(_term, _proposition)
Body: Ask-Ref(_agent1, _agent2, _term, _proposition)
Answer-Ref(_agent2, _agent1, _term, _proposition)
Effects: information-sought(_agent1, _agent2, _proposition)
Goal: knowref(_agent1, _term, _proposition)

Discourse Recipe
Action: Ref-Request(_agent1, _agent2, _term, _proposition)
{*_agent1 requests the referent of _term in _proposition*}
Recipe-Type: Specialization
Constraint: term-in(_term, proposition)
Body: Surface-WH-Question(_agent1, _agent2, _term, _proposition)
Effects: believe(_agent2, requested(_agent1, _term, _proposition), [C:C])
Goal: believe(_agent2, want(_agent1, know-ref(_agent1, _term,
believe(_agent2, _proposition, [C:C]))), [C:C])

Discourse Recipe
Action: Resolve-Conflict(_agent1, _agent2, _proposition1, _proposition2)
{*_agent1 resolves the conflict of _proposition1 and _proposition2*}
Recipe-Type: Decomposition
Appl Cond: believe(_agent1, believe(_agent2, _proposition2 → ¬_proposition1,
[S:C]), [W:C])
believe(_agent1, believe(_agent2, _proposition2, [W:S]), [W:C])
believe(_agent1, _proposition1, [C:C])
Constraints: equalorneg(_proposition2, _proposition3)
salient(_proposition1)
salient(_proposition2)
Body: Inform(_agent1, _agent2, _proposition3)
#Explain-Claim(_agent1, _agent2, _proposition1, _proposition2)
Effects: conflict-addressed(_agent1, _agent2, _proposition1, _proposition2)
Goal: believe(_agent2, in-conflict(_proposition1, _proposition2), [C_N:W_N])

Discourse Recipe
Action: Surface-Neg-YN-Question(_agent1, _agent2, _proposition)
{*_agent1 makes a surface negative request about _proposition*}
Recipe-Type: Primitive
Appl Cond: believe(_agent1, _proposition, [S:S])
Effects: asked-about(_agent1, _agent2, _proposition)
Goal: asked-about(_agent1, _agent2, _proposition)

Discourse Recipe
Action: Surface-Say-Prop(_agent1, _agent2, _proposition)
{*_agent1 makes a surface utterance of _proposition to _agent2*}
Recipe-Type: Primitive
Appl Cond: believe(_agent1, _proposition, [C:C])
Effects: said(_agent1, _agent2, _proposition)
Goal: said(_agent1, _agent2, _proposition)

Discourse Recipe
Action: Surface-WH-Question(_agent1, _agent2, _term, _proposition)
{*_agent1 makes a surface request for the _term in _proposition*}
Recipe-Type: Primitive
Constraints: term-in(_term, _proposition)
Effects: asked-for(_agent1, _agent2, _term, _proposition)
Goal: asked-for(_agent1, _agent2, _term, _proposition)

Discourse Recipe
Action: Tell(_agent1, _agent2, _proposition)
{*_agent1 tells _agent2 of _proposition*}
Recipe-Type: Decomposition
Appl Cond: believe(_agent1, _proposition, [C:C])
Body: Surface-Say-Prop(_agent1, _agent2, _proposition)
#Address-Understanding(_agent1, _agent2, _proposition)
Effects: told-about(_agent1, _agent2, _proposition)
Goal: believe(_agent2, believe(_agent1, _proposition, [C:C]), [C:C])

References

Allen, James and Lenhart Schubert. 1991. The trains project. Technical Report 91-1, Department of Computer Science, University of Rochester, Rochester, New York.

Allen, James F. 1979. *A Plan-Based Approach to Speech Act Recognition.* Ph.D. thesis, University of Toronto, Toronto, Ontario, Canada.

Allen, James F. and C. Raymond Perrault. 1980. Analyzing Intention in Utterances. *Artificial Intelligence*, 15:143–178.

Ballim, Afzal and Yorick Wilks. 1991. Beliefs, Stereotypes, and Dynamic Agent Modeling. *User Modeling and User-Adapted Interaction*, 1(1):33–66.

Bartelt, Margaret. 1996. A Computer Program that Recognizes Rejected Questions Computationally. In *Proceedings of NCUR-10*, pages 989–993.

Birnbaum, Lawrence, Margot Flowers, and Rod McGuire. 1980. Towards an AI Model of Argumentation. In *Proceedings of the First National Conference on Artificial Intelligence*, pages 306–309.

Bonarini, Andrea, Ernesto Cappelletti, and Antonio Corrao. 1990. Network-based management of subjective judgements: A proposal accepting cyclic dependencies. Technical Report 90-067, Dipartimento di Elettronica, Politecnico di Milano, Milano, Italy.

Carberry, Sandra. 1985. A Pragmatics Based Approach to Understanding Intersentential Ellipsis. In *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics*, pages 188–197.

Carberry, Sandra. 1987. Pragmatic Modeling: Toward a Robust Natural Language Interface. *Computational Intelligence*, 3:117–136.

Carberry, Sandra. 1988. Modeling the User's Plans and Goals. *Computational Linguistics*, 14(3):23–37.

Carberry, Sandra. 1989. A Pragmatics-Based Approach to Ellipsis Resolution. *Computational Linguistics*, 15(2):75–96.

Carberry, Sandra. 1990. *Plan Recognition in Natural Language Dialogue.* ACL-MIT Press Series on Natural Language Processing. MIT Press, Cambridge, Massachusetts.

Carberry, Sandra, Jennifer Chu-Carroll, and Lynn Lambert. 1996. Modeling intention: Issues for spoken language dialogue systems. In *Proceedings of the International Symposium on Spoken Dialogue*, pages 13–24.

Cawsey, Alison, Julia Galliers, Steven Reece, and Karen Sparck Jones. 1992. Automating the librarian: A fundamental approach using belief revision. Technical Report 243, University of Cambridge Computer Laboratory, Cambridge, England.

Chu-Carroll, Jennifer and Sandra Carberry. 1994. A plan-based model for response generation in collaborative task-oriented dialogues. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, pages 799–805.

Chu-Carroll, Jennifer and Sandra Carberry. 1995a. Generating information-sharing subdialogues in expert-user consultation. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 1243–1250.

Chu-Carroll, Jennifer and Sandra Carberry. 1995b. Response generation in collaborative negotiation. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pages 136–143.

Chu-Carroll, Jennifer and Sandra Carberry. 1998. Collaborative response generation in planning dialogues. *Computational Linguistics*, 24(3):355–400.

Clark, Herbert and Edward Schaefer. 1989. Contributing to Discourse. *Cognitive Science*, pages 259–294.

Clark, Herbert and Deanna Wilkes-Gibbs. 1990. Referring as a Collaborative Process. In Philip Cohen, Jerry Morgan, and Martha Pollack, editors, *Intentions in Communication.* MIT Press, Cambridge, Massachusetts, pages 463–493.

Cohen, Paul R. 1985. *Heuristic Reasoning about Uncertainty: An Artificial Intelligence Approach.* Pitman Publishing Company.

Cohen, Philip and Hector Levesque. 1990a. Intention is Choice with Commitment. *Artificial Intelligence*, 42:213–261.

Cohen, Philip and Hector Levesque. 1990b. Persistence, Intention, and Commitment. In Philip Cohen, Jerry Morgan, and Martha Pollack, editors, *Intentions in Communication.* MIT Press, Cambridge, Massachusetts, pages 33–70.

Cohen, Philip and Hector Levesque. 1990c. Rational Interaction as the Basis for Communication. In Philip Cohen, Jerry Morgan, and Martha Pollack, editors, *Intentions in Communication.* MIT Press, Cambridge, Massachusetts, pages 221–256.

Cohen, Philip and Hector Levesque. 1991a. Confirmations and Joint Action. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 951–957.

Cohen, Philip and Hector Levesque. 1991b. Teamwork. Technical Report 504, SRI International, Menlo Park, California.

Cohen, Philip R. and C. Raymond Perrault. 1979. Elements of a Plan-Based Theory of Speech Acts. *Cognitive Science*, 3:177–212.

Cohen, Robin. 1987. Analyzing the Structure of Argumentative Discourse. *Computational Linguistics*, 13(1-2):11–24.

Cohen, Robin and Mark Anthony Young. 1991. Determining Intended Evidence Relations in Natural Language Arguments. *Computational Intelligence*, 7:110–118.

DeKleer, Johan. 1986. An Assumption-Based TMS. *Artificial Intelligence*, 28:269–301.

Driankov, Dimiter. 1988. *Towards a Many-Valued Logic of Quantified Belief.* Ph.D. thesis, Linkoping University, Department of Computer and Information Science, Linkoping, Sweden.

Elzer, Stephanie. 1995. The role of user preferences and problem-solving knowledge in plan recognition for expert consultation systems. In *Proceedings of the IJCAI Workshop on the Next Generation of Plan Recognition Systems*, pages 37–41.

Flowers, Margot and Michael E. Dyer. 1984. Really Arguing with your Computer. In *Proceedings of the National Computer Conference*, pages 653–659.

Flowers, Margot, Rod McGuire, and Lawrence Birnbaum. 1982. Adversary Arguments and the Logic of Personal Attack. In W. Lehnert and M. Ringle, editors, *Strategies for Natural Language Processing.* Lawrence Erlbaum Assoc, Hillsdage, New Jersey, pages 275–294.

Galliers, Julia Rose. 1991. Belief Revision and a Theory of Communication. Technical Report 193, University of Cambridge, Cambridge, England.

Galliers, Julia Rose. 1992. Autonomous belief revision and communication. In P. Gardenfors, editor, *Belief Revision*, Cambridge tracts in theoretical computer science. Cambridge University Press, Cambridge, England.

Grosz, Barbara and Candace Sidner. 1986. Attention, Intentions, and the Structure of Discourse. *Computational Linguistics*, 12(3):175–204.

Grosz, Barbara and Candace Sidner. 1990. Plans for Discourse. In P. Cohen, J. Morgan, and M. Pollack, editors, *Intentions in Communication*. MIT Press.

Grosz, Barbara J., Aravind K. Joshi, and Scott Weinstein. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–225.

Heeman, Peter. 1991. A Computational Model of Collaboration on Referring Expressions. Master's thesis, University of Toronto, September. Also Technical Report CSRI-251.

Hinkelman, Elizabeth. 1989. Two Constraints on Speech Act Ambiguity. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, pages 212–219.

Hirschberg, Julia and Diane Litman. 1987. Now Let's Talk About Now. In *Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics*, pages 163–171.

Joshi, Aravind K. 1982. Mutual Beliefs in Question-Answer Systems. In N. Smith, editor, *Mutual Beliefs*, pages 181–197, New York. Academic Press.

Kautz, Henry. 1990. A Circumscriptive Theory of Plan Recognition. In Philip R. Cohen, Jerry Morgan, and Martha E. Pollack, editors, *Intentions in Communication*. MIT Press, Cambridge, Massachusetts, pages 105–133.

Knott, Alistair and Robert Dale. 1994. Using linguistic phenomena to motivate a set of coherence relations. *Discourse Processes*, 18(1):35–62.

Knott, Alistair and Chris Mellish. 1996. A feature-based account of the relations signalled by sentence and clause connectives. *Language and Speech*, 39(2-3):143–183.

Lambert, Lynn. 1993. *Recognizing Complex Discourse Acts: A Tripartite Plan-Based Model of Dialogue*. Ph.D. thesis, University of Delaware, June.

Lambert, Lynn and Sandra Carberry. 1991. A Tripartite Plan-based Model of Dialogue. In *Proceedings of the 29th Annual Meeting of the ACL*, pages 47–54.

Lambert, Lynn and Sandra Carberry. 1992. Modeling Negotiation Subdialogues. In *Proceedings of the 30th Annual Meeting of the ACL*, pages 193–200.

Lewis, D. 1979. Scorekeeping in a language game. *Journal of Philosophical Logic*, pages 339–359.

Litman, Diane and James Allen. 1987. A Plan Recognition Model for Subdialogues in Conversation. *Cognitive Science*, 11:163–200.

Litman, Diane and Julia Hirschberg. 1990. Disambiguating Cue Phrases in Text and Speech. In *Proceedings of the 13th International Conference on Computational Linguistics*, pages 251–256.

Lochbaum, Karen. 1991. An Algorithm for Plan Recognition in Collaborative Discourse. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, pages 33–38.

Lochbaum, Karen, Barbara Grosz, and Candace Sidner. 1990. Models of Plans to Support Communication: An Initial Report. In *Proceedings of the Eighth National Conference on Artificial Intelligence*, pages 485–490.

Lochbaum, Karen Elizabeth. 1994. *Using Collaborative Plans to Model the Intentional Structure of Discourse*. Ph.D. thesis, Harvard University, Cambridge, Massachusetts. Tech Report: TR-25-94.

Marcu, Daniel. 1997. The rhetorical parsing of natural language text. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pages 96–103.

McGuire, Rod, Lawrence Birnbaum, and Margot Flowers. 1981. Opportunistic Processing in Arguments. In *Proceedings of the 1981 International Joint Conference on Artificial Intelligence*, pages 58–60.

McKeown, Kathleen R. 1983. Focus Constraints on Language Generation. In *Proceedings of the Third National Conference on Artificial Intelligence*, pages 582–587.

McRoy, Susan and Graeme Hirst. 1995. The repair of speech act misunderstandings by abductive inference. *Computational Linguistics*, pages 435–478.

Perrault, Raymond. 1990. An Application of Default Logic to Speech Act Theory. In Philip Cohen, Jerry Morgan, and Martha Pollack, editors, *Intentions in Communication*. MIT Press, Cambridge, Massachusetts, pages 161–185.

Perrault, Raymond and James Allen. 1980. A Plan-Based Analysis of Indirect Speech Acts. *American Journal of Computational Linguistics*, 6(3-4):167–182.

Polanyi, Livia. 1986. The Linguistic Discourse Model: Towards a Formal Theory of Discourse Structure. Technical Report 6409, Bolt Beranek and Newman Laboratories Inc., Cambridge, Massachusetts.

Pollack, Martha. 1990. Plans as Complex Mental Attitudes. In Philip R. Cohen, Jerry Morgan, and Martha E. Pollack, editors, *Intentions in Communication*. MIT Press, pages 77–104.

Quilici, Alexander. 1991. *The Correction Machine: A Computer Model of Recognizing and Producing Belief Justifications in Argumentative Dialogs*. Ph.D. thesis, Department of Computer Science, University of California at Los Angeles, Los Angeles, California.

Ramshaw, Lance A. 1989. A Metaplan Model for Problem-Solving Discourse. In *Proceedings of the Fourth Conference of the European Chapter of the Association for Computational Linguistics*, pages 35–42.

Reichman, Rachel. 1978. Conversational Coherency. *Cognitive Science*, 2:283–327.

Reichman, Rachel. 1981. Modeling Informal Debates. In *Proceedings of the 1981 International Joint Conference on Artificial Intelligence*, pages 19–24.

Reichman, Rachel. 1985. *Getting Computers to Talk Like You and Me*. MIT Press, Cambridge, Massachusetts.

Reithinger, Norbert and Elisabeth Maier. 1995. Utilizing statistical dialogue act processing in verbmobil. In *Proc. of the Annual Meeting of the Association for Computational Linguistics*, pages 116–121.

Rosé, Carolyn Penstein, Barbara Di Eugenio, Lori Levin, and Carol Van Ess-Dykema. 1995. Discourse processing of dialogues with multiple threads. In *Proceedings of the 33rd Meeting of the Association for Computational Linguistics*, pages 31–38.

Schegloff, Emanuel. and Harvey Sachs. 1973. Opening up closings. *Semiotica*, 8:289–327.

Schiffrin, Deborah. 1987. *Discourse Markers*. Cambridge University Press, Cambridge, England.

Searle, John. 1990. Collective Intentions and Actions. In Philip Cohen, Jerry Morgan, and Martha Pollack, editors, *Intentions in Communication*. MIT Press, Cambridge, Massachusetts, pages 401–416.

Searle, John R. 1970. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge University Press, London, England.

Thomason, Richmond. 1990. Accommodation, meaning, and implicature: Interdisciplinary foundations for pragmatics. In Philip Cohen, Jerry Morgan, and Martha Pollack, editors, *Intentions in Communication*. MIT Press, Cambridge, Massachusetts, pages 325–363.

Transcripts, Columbia University. 1985. Transcripts derived from audiotape conversations made at columbia university, new york, new york. Provided by Kathleen McKeown.

Transcripts, Harry Gross. 1982. Transcripts derived from tapes of the radio talk show harry gross: Speaking of your money. Provided by the Dept. of Computer Science at the University of Pennsylvania.

Transcripts, SRI. 1992. Transcripts derived from audiotape conversations made at sri international, menlo park, california. Prepared by Jacqueline Kowtko under the direction of Patti Price.

Traum, David. 1994. *A Computational Theory of Grounding in Natural Language Conversation*. Ph.D. thesis, University of Rochester.

Traum, David and Elizabeth Hinkelman. 1992. Conversation acts in task-oriented spoken dialogue. *Computational Intelligence*.

van Beek, Peter and Robin Cohen. 1986. Towards User Specific Explanations from Expert Systems. In *Proceedings of the Sixth Canadian Conference on Artificial Intelligence*, pages 194–198.

Walker, Marilyn. 1991. Redundancy in Collaborative Dialogue. In *Working Notes for the AAAI Fall Symposium: Discourse Structure in Natural Language Understanding and Generation*, pages 124–129.

Walker, Marilyn. 1992. Redundancy in Collaborative Dialogue. In *Proceedings of the Fifteenth International Conference on Computational Linguistics*, pages 345–351.

Walker, Marilyn. 1996. Inferring acceptance and rejection in dialog by default rules of inference. *Language and Speech*, 39(2-3):265–304.

Walker, Marilyn and Steve Whittaker. 1990. Mixed Initiative in Dialogue: An Investigation into Discourse Segmentation. In *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics*, pages 70–78.

Wilensky, Robert. 1981. Meta-Planning: Representing and Using Knowledge About Planning in Problem Solving and Natural Language Understanding. *Cognitive Science*, 5:197–233.

Young, Mark Anthony. 1987. The Design and Implementation of an Evidence Oracle for the Understanding of Arguments. Technical report, University of Waterloo, Waterloo, Ontario, Canada.