An Artificial Discourse Language for Collaborative Negotiation

Candace L. Sidner
Lotus Development Corp.
One Rogers St.
Cambridge, MA 02139
email: csidner@lotus.com

Abstract

Collaborations to accomplish common goals necessitate negotiation to share and reach agreement on the beliefs that agents hold as part of the collaboration. Negotiation in communication can be simulated by a series of exchanges in which agents propose, reject, counterpropose or seek supporting information for beliefs they wish to be held mutually. In an artificial language of negotiation, messages display the state of the agents' beliefs. Dialogues consisting of such messages clarify the means by which agents come to agree or fail to agree on mutual beliefs and individual intentions.

Introduction

In human problem solving, agents often recognize that they share goals in common. To achieve their common goals, they plan and act jointly. These activities are collaborative processes. Collaboration requires *negotiation*, that is, the interactive process of attempting to agree on the goals, actions and beliefs that comprise the planning and acting decisions of the collaboration.

This paper reports on an artificial language and associated machinery for modelling discourses in which agents discuss their collaborative activities (Grosz & Sidner 1990; Lochbaum, Grosz, & Sidner 1990; Grosz & Kraus 1993; Lochbaum 1993). The need for this language results from an attempt to understand how agents might come to hold the beliefs and intentions of the SharedPlan model (Grosz & Sidner 1990; Grosz & Kraus 1993).

A Sample Human Negotiation

To begin to address the questions concerning beliefs, and the means by which beliefs become shared, consider the portion of a dialogue transcription shown in the dialogue D1 shown in Figure 1.

The dialogue the transcription captures was spoken between M and K, who were participating in a biweekly meeting; M was the group leader and K a member of the group. The text has been editted to improve its intelligibility by removing "ums," "ahs," pauses and extranaeous phrases.

M and K demonstrate the most typical characteristics of negotiation in discourse: proposal and acceptance or proposal and rejection sequences. While this phenomenon has been subject to some previous study (Winograd 86; Weihmayer & Brandau 1990) my focus concerns the nature of the beliefs at each step of the communication. In the sample, K offers for mutual belief her belief that she should discuss graphics tools with John. M, who might be understood to be agreeing by saying, "OK," is in fact simply listening to K's proposal. When she has heard the whole message from K, she makes clear that she does not believe K's proposed action is best. It is vital to bear in mind that these agents are focused on the common goal they have agreed to (automating K's job), and thus they are able to infer that K's proposal is meant to be about actions that will *contribute* (Grosz & Sidner 1990; Lochbaum, Grosz, & Sidner 1990; Grosz & Kraus 1993) to the common goal.

An Artificial Language of Negotiation

English, like any human language, provides agents with a variety of ways to express beliefs and intentions. To abstract from these, I have devised an artificial language, which is given below. The language is defined as a series of messages from one agent to another, the content of which includes a proposition. For each message type, after it is received, certain beliefs or intentions can be taken as true regarding the proposition. The state of beliefs or intentions following a message abstracts away from individual agents in the negotitation and provides the state of communication held by all the agents.

In addition to the language, I define some additional machinery for "interpreting" the language: a set of stacks for Open beliefs and Rejected beliefs, which capture part of the state of the discourse. This machinery informally captures a portion of the attentional state in the sense of (Grosz & Sidner 1986), but a complete rendering of beliefs and intentions in terms of that theory requires further research.

I also assume an automated belief revision system to track all the mutual beliefs that come to be held.

M: Throughput, error meeting, European conjestion. We started talking at the error meeting about figuring out a good tool to deal with I mean we've seen the same kind of problem come up over and over again. And there may be some way to automate what you do a little bit.

K: One thing I've got to do [M: what?] is I've got to talk to John about his graphics stuff. Evidently people have been telling me he has a thing that makes nodes, circles and draws lines. Mean that would help even just as a start.

M: OK

K: but I have, I need to do that, and see what he he's got written. All right?

M: I'm not sure it's worth using fancy graphics on the LISP machine or stuff for this kind of thing.

K: I'm not even thinking about fancy. I mean I would just, like I said, if I had a little xerox template that showed the nodes and I could fill in my own allergism [sic], just quickly draw it, that would be a help. [M: mm] You know.

M: Well, you'd also want Typically what we draw is something like this [K:yeh], right? [K:yeh] And I mean ideally I think what you'd like is a map, and then if you mark the lines that have [K: yeh] retransmissions, and then you also process the outage report and you mark the lines [K:right] that have outages [K:yup], during the same time that these ones had [K:yep] retransmissions. [K:that's] right?

Figure 1: Dialogue D1

Among the many TMSs, that of Galliers' (Galliers 1992) is most relevant here for its use of the notion of *more cohence*. Belief revision is critical because agents sometimes change their minds about proposals they have agreed on, and the decision to believe a new proposal may hinge on already held beliefs.

The Language Definition

In the definitions that follow, `BEL` is short for believe, `INT` for intend, and `MB` for mutual belief.

PFA (ProposeForAccept) agt1 belief agt2: Agt1 expresses belief to agt2. After receiving a message of this type from agt1, the state of communication is:

```
(BEL agt1 belief)
(INT agt1 (Achieve agt1 (BEL agt2 belief)))
(BEL agt1 (Communicated agt1 belief agt2)).
```

AR (AcknowledgeReceipt) agt1 belief agt2: Agt1 sends this message to agt2 to indicate that a previous message from agt2 about belief has been heard. This does *not* mean that agt1 believes belief. After receiving the message the communication state is:

```
(MB agt1 agt2 (BEL agt2 belief))
(MB agt1 agt2
  (INT agt2 (Achieve agt2 (BEL agt1 belief))))
(MB agt1 agt2 (Communicated agt2 belief agt1)).
```

Furthermore, `(Open belief)` occurs to put belief on Open stack.

RJ (Reject) agt1 belief agt2: Upon receipt of this message, agt2 can conclude that agt1 does not believe belief, which has been offered as a proposal. The effect is that belief is no longer an Open proposal. Following receipt of this message, the state of communication is:

```
(Not (BEL agt1 belief)
(BEL agt1
  (Communicated agt1
    (Not (BEL agt1 belief)) agt2)).
```

Furthermore, `(DeleteOpen belief)` removes belief from Open stack, and `(Rejected agt1 belief)` puts belief in Rejected stack.

ARJ (AcknowledgeReject) agt1 belief agt2: This is the counterpart of AR for rejections. It establishes the mutual belief of the conclusions from RJ:

```
(MB agt1 agt2 (Not (BEL agt2 belief)))
(MB agt1 agt2
   (Communicated agt2
      (Not (BEL agt2 belief)) agt1)).
```

AP (AcceptProposal) agt1 belief agt2: Upon receipt of this message from agt1, agt1 and agt2 now hold belief as a mutual belief:

```
(MB agt1 agt2 belief).
```

Also `(DeleteOpen belief)` occurs and belief is tracked by the belief revision system.

CO (Counter) agt1 belief1 agt2 belief2: Agt1 has reason to doubt belief1. Without rejecting belief1, agt1 offers belief2 to agt2. The state of the communication is just that which is obtained by sending the following messages:

```
(PFA agt1 belief2 agt2)
(PFA agt1 (Supports (Not belief1) belief2) agt2)
```

RP (RetractProposal) agt1 agt2 belief: Other researchers (Weihmayer & Brandau 1990) report the need for retraction of belief. This message is sent when agt1 no longer believes a belief proposed previously.

```
(Not (BEL agt1 belief))
(Not (INT agt1
   (Achieve agt1 (BEL agt2 belief))))
(BEL agt1
   (Communicated agt1
      (Not (BEL agt1 belief)) agt2)).
```

If belief has not been accepted before the retraction, it is on the Open stack, and must be deleted.

ARP (AcknowledgeRetractedProposal) agt1 agt2 belief: Acknowledgement is similar in kind to other acknowledgements. Note that agt2 was the source of the retraction that this message acknowledges.

```
(MB agt1 agt2 (Not (BEL agt2 belief)))
(MB agt1 agt2
   (Not
      (INT agt2
         (Achieve agt2 (BEL agt1 belief))))
(MB agt1 agt2
   (Communicated agt2
      (Not (BEL agt2 belief)) agt1)).
```

AOP (AcceptOthersProp) agt1 belief1 agt2 belief2: This message is used when agt1 realizes that belief1 is worthy of belief. Agt1 sends this message to indicate that belief2 is now being retracted, and belief1 is being accepted. All the results of the following messages hold for this message:

```
(RP agt1 agt2 belief2)
(AP agt1 agt2 belief1).
```

PR (ProposeReplace) agt1 belief1 agt2 belief2: This message is shorthand for two messages:

```
(RJ agt1 belief2 agt2)
(PFA agt1 belief1 agt2).
```

PA (ProposeAct) agt1 agt2 action context: This message is a schema for

```
(PFA agt1 (Should-do agt2 action context) agt2)
```

where context is optional and the action can vary over a set that includes `Identify`, `Provide-Support`, and other actions. `Should-Do` is an optative expression over an action to be performed by the agent. `Should-Do` is not as strong as `INT`; see section for further discussion. This message is needed to correspond to questions and commands as in the following examples:

Why X?
```
(PFA agt1
   (Should-Do agt2
      (Provide-Support X context)) agt2);
```

What is X?
(similarly for where is X, when is X, and who is X)
```
(PFA agt1
   (Should-Do agt2 (Identify X context)) agt2);
```

Can you X?
```
(PA agt1 agt2
   (Should-Do agt2
      (Tellif agt2 (Able agt2 X context))));
```

Did John come?
```
(PA agt1 agt2
   (Should-Do agt2
      (Tellif agt2 '(john did come))));
```

Listen to this!
```
(PA agt1 agt2
   (Should-Do agt2 '(listen to this))).
```

Discussion

This language makes an important assumption, the mutual belief assumption, about the nature of communication, namely that following certain messages (e.g. AP, AR), mutual belief obtains among the agents of the collaboration. This assumption rests on the lack of intentional deception and misinformation in collaborative activity. For individual agents, the strongest statement that can be made is that the agent believes there is mutual belief. However, the language uses full mutual belief because if each agent can believe there is mutual belief and there is no deception, then full mutual belief follows. For synchronous communciations in collaboration, (e.g. face-to-face conversations, phone calls and the like), the mutual belief assumption is reasonable. For asynchronous cases (e.g. written or spoken email or letters), or cases where interaction is not possible (e.g. speeches to a large audience or written publications), the assumption of mutual belief can lead to difficulties (Halpern & Moses 1984) because, for example, one of the negotiators may never receive the acceptance message, or it cannot be sent.

The negotiation language does not constrain the order in which agents choose to send messages. Agents therefore can have incoherent discourses, or reach certain false conclusions if they send certain messages before others. For example, when agent A sends `(PFA A B x)` to agent B and then sends `(AP A B x)`, the state of communication includes that `(MB A B x)`. This pair of messages is odd because agent A is accepting his own proposal. The communication state misrepresents the real state of affairs in which agent B has yet to say anything about x. Operational constraints on sending of messages can eliminate such false conclusions.

Among the messages of this language, AR (AcknowledgeReceipt) serves a special role worth noting here. The linguistics and psychology literature[1] raises many issues regarding utterances that are misunderstood by the hearer and must be repaired before their content is understood. The AR message is a place holder for the simulation of this behavior. In particular, when a proposal is not understood, rather than send an AR message, some new message could be sent that would indicate for example, where the previous message was garbled or that the whole message was lost. The sample dialogues in this paper do not illustrate repairs and the like.

Sample Simulated Dialogues

To illustrate the language, I will present some sample constructed discourses. Discourses are about some domain of affairs, and in natural conversation this domain can be quite rich and diverse; the job automation dialogues illustrate this claim clearly. To illustrate the negotiation language, the domain is a very simple one, namely actions, denoted by A, B, etc. that contribute to a Goal, denoted only as G. Keeping the domain of conversation straightforward facilitates closer observation of the properties of the language of negotiation.

A Simple Negotiation The first example of negotiation in the artificial language involves two agents R and C. These agents have a partial SharedPlan for accomplishing G, a goal, and no shared recipes for the performance of G[2]. They mutually believe that each knows a way to do A and D, two actions. C believes A Enables D, while R believes D generates G(Balkanski 1990). To see how they come to decide the recipe for G, they could have the dialogue given in Figure 2. An

[1] A paper by (Clark & Shaefer 1987) contains discussion of this literature.

[2] See (Grosz & Kraus 1993) for elaboration of partial SharedPlans.

```
1 (PFA C (Should-Do R&C A) R)     C: Let's do A.

1'(AR R C (Should-Do R&C A))      R: uh-huh.
  --Additional messages of this type
    are not included in the conversation--

2 (PR R (Should-Do R&C D) C       R: No, let's
 (Should-Do R&C A))             do D.

3 (RJ C (Should-Do R&C D) R)      C: No.

4 (PFA R (Should-Do C             R: Why do A?
   (Provide-Support C
     (Should-Do R&C A)
     (Recipe G))) C)

5 (AP C                           C: ok.
     (Should-Do C
       (Provide-Support  C
   (Should-Do R&C A)
   (Recipe G))))

6 (PFA C                          C: A enables D.
   (Enables A D) R)

7 (AP R                           R: ok.
   (Enables A D) C C)

8 (PFA C                          C: Let's do A.
   (Should-Do R&C A) R)

9 (AP R                           R: ok.
   (Should-Do R&C A) C C)

10 (PFA R (Generates D G) C)      R: D generates G.

11 (AP C (Generates D G) R)       C: ok.

12 (PFA R                         R: Let's do D.
   (Should-Do R&C D) C)

13 (AP C (Should-Do R&C D) R)     C: ok.
```

Figure 2: Negotiating to achieve G

English gloss for each message is given at the right.

This conversation is not as fluent as human ones. First, it includes extra "uh-huhs" to tell the sending agent that the message was received. In the figure, most of these have been deleted for ease of understanding, but a full conversation includes one after every PFA and RJ. Second, the language demands additional messages when undertaking action. For example, at line 5, an extra "ok" occurs to tell R that C is accepting R's previous proposal (that is, C will answer the question). The artificial conversation also demands that agents say more than "no" or "ok" to a proposal; the content of the proposal is repeated in each case (the English gloss does not include this repetition).

Figure 3 illustrates the states of mutual belief, private beliefs, the Open stack and the Rejected stack after line 2 of the dialogue. For purposes of contrast, the figure illustrates only the acknowledgement (ARJ) of the RJ that is part of PR as having occurred. The ARJ for RJ leads to the mutual belief that

`(Not (BEL R (Should-Do R&C A)))`.

owever, the acknowledgement AR for the proposal `(Should-Do R&C D)` has not occurred, so the three private beliefs concerning doing D are not yet mutually believed, and the Open stack does not contain a proposal under consideration.

While clearly simpler than most human conversations, this sample dialogue illustrates how agents can establish mutual beliefs concerning a common goal without simply believing everything that is said. The basic sequence of proposing new beliefs (such as `(Should-Do R&C A)`), rejecting and eventually accepting them is demonstrated.

The sample dialogue raises an intriguing question: Why do R and C accept some proposals and reject other ones? In creating this dialogue, I stipulated that the agents R and C would accept any belief of another agent that could be seen to contribute to the goal G. Thus agent R is able to accept the proposal that A enables D because R knows already that D generates G, and thus the proposal has information that contributes to the goal G. Use of an artificial language of negotiation by artificial agents requires that the agents be constructed with enough intelligence to decide what beliefs to accept when proffered by other agents.

Likewise, the nature of strategies pursued by the agents in the sample conversation is paramount for constructing artificial agents. In the sample, C appears to use the strategy of "propose to do whatever you know about" (a rather foolhardy strategy). A more sensible strategy would be to question R about what R believes. Strategies for focusing attention, explored by Walker (Walker 1992) also result in more efficient conversations. Another fruitful area of research would be to develop strategies along the lines of the deals in (Zlotkin & Rosenschein 1990).

Mutual beliefs of R and C:

```
(BEL C (Should-Do R&C A))
(INT C (Achieve C (BEL R (Should-Do R&C A))))
(Communicated C (Should-Do R&C A) R))
(Not (BEL R (Should-Do R&C A)))
(Commmicated R (Not (BEL R (Should-Do R&C A))))
```

Beliefs held individually by each of R and C:

```
(BEL R (Should-Do R&C D))
(INT R (Achieve R (BEL C (Should-Do R&C D))))
(BEL R (Communicated R (Should-Do R&C D) C))
```

Open stack is empty.
Rejected stack: `(Rejected R (Should-Do R&C A))`

Figure 3: Belief states after line 2 of Figure 2

1 **C**: what way do you know to do Z?

2 **R**: *Ok.* A followed by B is part of it.

3 **C**: *Ok.*

4 **R**: let's do A.

5 **C**: *No*, F then B and D followed by E generates Z.

6 **R**: why do F? [in F;B]

7 **C**: *Ok.* The result of F is W.

8 **R**: *Yeh but* A is easier than F, *so* A;B then D;E generates Z.

9 **C**: *Yeh*, A's easier, *so* A's enabling conditions are different
from F's *so* A can't be in the recipe.

10 **R**: *But* they are the same. I've done A;B.

11 **C**: *Ok.* A and F have same results. *And* A;B then D;E generates Z.

12 **R**: *So* let's do A.

Figure 4: Proposals and counterproposals, English gloss

```
1 (PFA C (Should-Do R (Identify R (Recipe Z))) R)

2 (AP R (Should-Do R (Identify R (Recipe Z))) C)
  (PFA R (Contributes (A;B) Z) )

3 (AP C (Contributes (A;B) Z) R)

4 (PFA R (Should-Do R&C A) C)

5 (PR C (Generates (((F;B) & D); E) Z) R
      (Should-Do R&C A))

6 (PFA R (Should-Do C
    (Provide-Support C (Contributes (F;B) Z))) C)
      C)

7 (AP C (Should-Do C
    (Provide-Support C (Contributes (F;B) Z))) R)
  (PFA C (Equal (Result F) W) R)

8 (AP R  (Equal (Result F) W) C)
  (CO R (Generates (((F;B) & D); E) Z) C
      (And (EasierThan A F)
           (Generates (((A;B) & D);E) Z)))

9 (AP C (EasierThan A F) R)
  (CO C (Generates (((A;B) & D);E) Z) R
      (Not (Equal (Result A) (Result F))))

10 (CO R (Not (Equal (Result A) (Result F)))
        C  (Equal (Result A)(Result F)))
   (PFA R (Done R A;B) C)

11 (AP C (Done R A;B) R)
   (AOP C  (Equal (Result A) (Result F)) R
       (Not (Equal (Result A) (Result F))))
   (AOP C ((((A;B) & D);E) Generates Z) R
         ((((F;B) & D); E) Generates Z))

12 (PFA R (Should-Do R&C A) C)
```

Figure 5: Artificial language version of Figure 4

A Counterproposal Dialogue The artificial language presented here can be used to simulate much more complex conversations. A gloss of one such conversation is illustrated in Figure 4; the artificial language version is shown in Figure 5. Like the previous sample, the domain of conversation is actions, denoted by letters, and relations among them that achieve a goal. For this example, assume that C and R have a partial SharedPlan to do Z; that R believes that A followed by B contributes to doing Z; that C believes a recipe of the form F followed by B with D, then followed by E generates Z; that C believes R knows some way to achieve Z (but does not know what it is) and that R and C mutually believe the W enables B.

This conversation makes use of the CO message to allow two different proposals (at line 8) to be active at one time; maintaining more than one open proposal is a common feature of human discourses and negotiations. In the artificial language CO allows each agent to make beliefs available for mutual belief but without having to communicate that the other agent's beliefs are not believed.

This dialogue also demonstrates a a typical human behavior, dubbed "call for the question." Calls of the question allow the questioner to delay accepting or rejecting a proposed belief until the questioner can collect more information about whether the belief is reasonable to believe. In the dialogue of Figure 5, at line 6, R asks a question in order to determine whether action F is correct in the proposal at line 5.

Two other features of the sample conversation are noteworthy. First, in line 11 C names all the proposals being accepted. In normal English discourse only the main proposal would be mentioned; the others are usually assumed as inferrable. Second, cue phrases (cf. (Grosz & Sidner 1986)), which are not part of the negotiation language, are presented in italics. While not necessary in the language, they may play a role in natural language to indicate the state of negotiations, because human speakers fail to repeat just what beliefs they are accepting, rejecting or countering.

For reasons of brevity, the M and K dialogue in Figure 1 cannot be illustrated in this paper, but is given in (Sidner 1993).

Related Research

As so far defined, this language does not take a stand on matter of commitments (Shoam 1990) or promises (Winograd 86). Agents must resolve when these actions are relevant by determining which types of mutual beliefs signal commitment. For example, once an agent agrees that he Should-Do an action, the two agents might decide that such agreement means the agreeing agent is committed to (or even signals intent to do) the action, and hence that the acceptance is a promise.

Cohen and Levesque (Cohen & Levesque 1990) have proposed an alternative formulation to (Grosz & Sidner 1990) for collaboration using their modal language of intention. Their definition of joint goal (JPG) for two agents is specified in terms of mutual belief and a mutual goal between the agents. The negotiation language is compatible for use with this account of collaboration.

Research on argumentation (Kraus & Sycara 93) in non-collaborative interaction (but where cooperation among agents is required) takes for granted the need for requests, statements, and threats. The artificial language proposed here is less compatible with these communiations because in that framework the agents can lie about their beliefs.

Future Directions

How can we test the completeness of the artificial language for capturing human conversations? Are there other features of dialogue in addition to cue phrases that are outside the language? Current investigations (Sidner 1994) are exploring the translation of dialogues into the negotiation language to determine potentially missing features and to characterize the interpretation of multi-functional phrases, such as "okay." The negotiation language is also being explored as a communication language between agents and users in product applications.

Acknowledgements

The author thanks the anonymous reviewers of the AAAI-92 workshop on Cooperation among Heterogeneous Intelligent Agents for their thoughtful remarks on an earlier draft of this paper, and Cliff Kahn, Bonnie Webber and Chuck Rich for their comments.

References

Balkanski, C. T. 1990. Modelling act-type relations in collaborative activity. Technical Report TR-23-90, Harvard University.

Clark, H. H., and Shaefer, E. 1987. Collaborating on contributions to conversations. *Language and Cognitive Processes* 11:1–23.

Cohen, P. R., and Levesque, H. 1990. On acting together. In *Proceedings of the Eighth National Conference on Artificial Intelligence*, 94–99. Menlo Park, CA: AAAI.

Galliers, J. R. 1992. Autonomous belief revision and communication. In Gaedenfors, P., ed., *Belief Revision.* Cambridge University Press. 220–246.

Grosz, B., and Kraus, S. 1993. Collaborative plans for group activities. In *Proceedings of IJCAI-13*, 367–373.

Grosz, B., and Sidner, C. 1986. Attention, intentions, and the structure of discourse. *Computational Linguistics* 12(3):175–204.

Grosz, B., and Sidner, C. 1990. Plans for discourse. In Cohen, P.; Morgan, J.; and Pollack, M., eds., *Intentions in Communication.* MIT Press.

Halpern, J., and Moses, Y. 1984. Knowledge and common knowledge in a distributed environment. In *Proceedings of the Third ACM Conference on the Principles of Distributed Computing.* ACM.

Kraus, S., M. N., and Sycara, K. 93. Reaching agreements through argumentation: A logical approach. In *Proceedings of the 12th International Workshop on Distributed Artificial Intelligence.*

Lochbaum, K. E.; Grosz, B. J.; and Sidner, C. L. 1990. Models of plans to support communication: An initial report. In *Proceedings of AAAI-90*, 485–490. Menlo Park, CA: AAAI Press/MIT Press.

Lochbaum, K. E. 1993. A collaborative planning approach to discourse understanding. Technical Report TR-20-93, Aiken Computational Laboratory, Harvard University, Cambridge, MA.

Shoam, Y. 1990. Agent oriented programming. Technical Report STAN-CS-90-1335, Computer Science Dept., Stanford University, Stanford, CA.

Sidner, C. L. 1993. The role of negotiation in collaborative activity. In Terveen, L., ed., *Human-Computer Collaboration: Reconciling Theory, Synthesizing Practice, Papers from the 1993 Fall Symposium Series, AAAI Technical Report FS-93-05.* Menlo Park, CA: AAAI Press.

Sidner, C. 1994. Negotiation in collaborative activity: A discourse analysis. *Knowledge-Based Systems.* forthcoming.

Walker, M. A. 1992. Redundancy in collaborative dialogue. In Boitet, C., ed., *Proceedings of the 14th Int. Conf. on Computational Linguistics*, 345–351. Assoc. for Computational Linguistics.

Weihmayer, R., and Brandau, R. 1990. Cooperative distributed problem solving for communication network management. *Computer Communications* 13(9).

Winograd, T. 86. A language/action perspective on the design of cooperative work. In *Proceedings of CSCW-86*, 203–220.

Zlotkin, G., and Rosenschein, J. S. 1990. Neogitation and conflict resolution in non-cooperative domains. In *Proceedings of the Eighth National Conference on Artificial Intelligence*, 100–105. Menlo Park, CA: AAAI.