



Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

**Technical
Memo**
TM-91-09

**On the Semantics of Protocols Among
Distributed Intelligent Agents**

Munindar P. Singh

August 1991

**Deutsches Forschungszentrum für Künstliche Intelligenz
GmbH**

Postfach 20 80
D-6750 Kaiserslautern
Tel.: (+49 631) 205-3211/13
Fax: (+49 631) 205-3210

Stuhlsatzenhausweg 3
D-6600 Saarbrücken 11
Tel.: (+49 681) 302-5252
Fax: (+49 681) 302-5341

Deutsches Forschungszentrum für Künstliche Intelligenz

The German Research Center for Artificial Intelligence (Deutsches Forschungszentrum für Künstliche Intelligenz, DFKI) with sites in Kaiserslautern und Saarbrücken is a non-profit organization which was founded in 1988 by the shareholder companies ADV/Orga, AEG, IBM, Insiders, Fraunhofer Gesellschaft, GMD, Krupp-Atlas, Mannesmann-Kienzle, Philips, Siemens and Siemens-Nixdorf. Research projects conducted at the DFKI are funded by the German Ministry for Research and Technology, by the shareholder companies, or by other industrial contracts.

The DFKI conducts application-oriented basic research in the field of artificial intelligence and other related subfields of computer science. The overall goal is to construct *systems with technical knowledge and common sense* which - by using AI methods - implement a problem solution for a selected application area. Currently, there are the following research areas at the DFKI:

- Intelligent Engineering Systems
- Intelligent User Interfaces
- Intelligent Communication Networks
- Intelligent Cooperative Systems.

The DFKI strives at making its research results available to the scientific community. There exist many contacts to domestic and foreign research institutions, both in academy and industry. The DFKI hosts technology transfer workshops for shareholders and other interested groups in order to inform about the current state of research.

From its beginning, the DFKI has provided an attractive working environment for AI researchers from Germany and from all over the world. The goal is to have a staff of about 100 researchers at the end of the building-up phase.

Prof. Dr. Gerhard Barth
Director

On the Semantics of Protocols Among Distributed Intelligent Agents

Munindar P. Singh

DFKI-TM-91-09

This work has been supported by a grant from The Federal Ministry for Research and Technology (FKZ ITW-8903 0).

© Deutsches Forschungszentrum für Künstliche Intelligenz 1991

This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of Deutsches Forschungszentrum für Künstliche Intelligenz, Kaiserslautern, Federal Republic of Germany; an acknowledgement of the authors and individual contributors to the work; all applicable portions of this copyright notice. Copying, reproducing, or republishing for any other purpose shall require a licence with payment of fee to Deutsches Forschungszentrum für Künstliche Intelligenz.

On the Semantics of Protocols Among Distributed Intelligent Agents

Munindar P. Singh*

Center for Cognitive Science and
(and Dept of Computer Sciences)
University of Texas
Austin, TX 78712
USA

DFKI
Postfach 2080
D-6750 Kaiserslautern
GERMANY

`msingh@cs.utexas.edu`

*This research was supported by the National Science Foundation (through grant # IRI-8945845 to the Center for Cognitive Science, University of Texas at Austin) and by the DFKI (German Research Center for Artificial Intelligence), Kaiserslautern, Germany.

Abstract

The continuing expansion of distributed intelligent systems makes new demands on theories of communication in Computer Science. It is customary to describe the individual nodes or *agents* in an intelligent system in terms of higher level concepts like intentions, know-how and beliefs. However, current theories of the communication among such agents provide no form of a formal or rigorous semantics for the messages exchanged at a corresponding level of abstraction—they either concern themselves with implementational details or address what is, for artificial systems, an irrelevant aspect of the problem. A recent theory of communication that gives the *objective* model-theoretic semantics for speech acts is applied to this problem. This allows the important properties of protocols to be formalized abstractly, i.e., at the level of the application, not of the implementation. Further constraints on “good” designs can also be stated, which simplify the requirements imposed on the member agents. The resulting theory not only provides some insights into designing distributed intelligent systems, but also helps in their validation. As an example, it is applied to a logical reconstruction of the classical Contract Net protocol.

1 Introduction

The trend towards the development of increasingly intelligent systems is matched only by the trend towards the distribution of computing. Distributed Artificial Intelligence (DAI) lies at the intersection of these trends. Besides the well-known reasons for the usefulness of distributed systems, the continued development of DAI systems is attractive for the following reasons. DAI permits intelligent systems to be developed independently of each other and to be reused as components of new systems, i.e., as member *agents* in multiagent systems. This modularization is useful when expertise is distributed, as in medical diagnosis. It also adds to the robustness of the designed system by simplifying the acquisition and validation of knowledge relevant to different aspects of the domain. Moreover, it simplifies design for applications such as manufacturing planning and air-traffic control by allowing an intelligent agent to be located at the site where the data are available and where decisions have to be taken.

A major bottleneck in the design of DAI systems is the design of the protocols of interaction among their member agents. Unfortunately, while individual agents are usually described in terms of their knowledge, intentions and capabilities (i.e., high-level concepts), extant approaches to understanding the interactions between them are not sufficiently advanced. Even fairly recent DAI research, which provides primitives for communication among agent has tended to be concerned with the workings of the TCP/IP and similar protocols, i.e., it has not been possible to abstract out entirely aspects of communication roughly at or below the so-called Transport Layer of the classical ISO/OSI standard (e.g., see [Arni and others, 1990]). Even more to the point, current theories do not provide any kind of a formal or rigorous semantics for the messages exchanged in a DAI system.

This lack of a rigorous theory of the interactions among agents forces the system designer to think in terms of what are, from the point of view of DAI, merely details of the underlying architecture—these details are important, but are simply out of place in the context of DAI. The resulting mixing up of concerns often results in the behavior of the designed system depending crucially on details of the operating system and the network hardware. At the same time, the design of the individual agents is based on knowledge about the domain of application that they have at different stages of their computations. Thus there is no principled way to relate the interactions *among*

the agents to the knowledge *within* each of them. The designer must design some acceptable modes of interaction and relate them as best as possible to the knowledge of the agents. Not only is this a tedious task, it also has to be redone from the start if the system is ever re-implemented. And no help is provided when systems implemented in different ways are to be integrated. In short, the problems with extant technology are that

1. It requires that the interactions among agents be designed from a scratch each time.
2. The semantics of these interactions is embedded in the procedures, some of which involve network and operating system code. This makes the validation and modification of systems, even otherwise not trivial, even more difficult.
3. Systems designed independently cannot be easily integrated.
4. Graceful updation or redesign of a system is virtually impossible: one cannot easily replace an existing agent with a new one.

Taken together, these limitations subvert many of the main motivations for developing DAI. The goal of this paper is to present a theory of the interaction among agents and a formal semantics for their interactions. Our key methodological assumptions are the following. We take it for granted that intelligent agents can be best described (for design or analysis) with concepts such as intentions, know-how or beliefs. This is quite a standard assumption in AI [McCarthy, 1979]. We consider DAI systems from without, i.e., as designers and analyzers. We do *not* directly take the point of view of the different agents who compose the system. Thus we attribute beliefs and intentions to agents, and describe their communications as we see fit from an "external" viewpoint rather than how they might actually be represented in the agents. This is useful since this leaves the exact design of the agents an open issue to be settled later in the design process, provided they meet the minimal requirements imposed.

Recently much work has been done on the design of protocols based on a notion of "knowledge" [Halpern and Moses, 1987]. However, papers on this theme consider the knowledge that the processes have of the process of communication itself, e.g., about whether certain messages have been delivered

to the intended recipient or not. Also, these protocols are designed for lower level data transmission. The work reported here is significantly different in that it emphasizes and studies the semantics of the messages exchanged, not the process of exchanging them.

The rest of the paper is organized as follows. In §2, we broadly classify the kinds of communicative interaction that occur most often in DAI systems, briefly describe *Speech Act Theory* and relate it to those interactions. In §3, we describe a recent formal theory of the objective semantics of the major kinds of speech acts. In §4, we show how this theory can be applied to the understanding of protocols in DAI systems. In §5, we present a detailed example of the logical reconstruction of the *Contract Net*, a celebrated protocol in DAI, which we also describe within.

2 Kinds of Interactions among Agents

The behavior of a DAI system depends not just on its component agents, but also on how they interact. In the more interesting cases, the agents would also intelligently decide how to interact with other agents by considering their current situation at that time.

2.1 Protocols

Therefore, in a DAI system of sufficient complexity, each agent would not only need to be able to do the tasks that arise locally, but would also need to interact effectively with other agents. We take *protocols* to be the specifications of these interactions. Agents participate in different protocols by appropriately interacting with each other, e.g., by responding to messages, performing actions in their given domain, or updating their local states. Protocols can thus be taken as specifying the *policies* that the agents would follow with regard to their interactions with other agents; e.g., these policies would determine the conditions under which a request would be acceded to or a permission issued or a statement believed. These policies could be fixed to some extent at the time of design, but would involve significant components that depended on the agents' current situation and thus could be computed only during execution; e.g., a request might be acceded to only if it does not lead to overload. Protocols, when seen in this way, are a nice way to

enforce modularity in the design of a DAI system by separating the interface between agents from their internal design. These protocols are meant to be rather high-level; in the classical seven-layer ISO/OSI framework, they would lie in the application layer. Some of these protocols may, in practice, precede “real” applications-level communication by facilitating the setting up of another protocol. This distinction is not crucial for our purposes.

Several kinds of formalizations may be attempted for protocols. One kind would concern the deliberation processes of the agents as they decide how to respond to a message. These processes are highly nonmonotonic and can be accurately understood only with theories of belief and intention revision, which are still not sufficiently well-developed (e.g., see [Perrault, 1987]). Another formalization concerns the objective conditions of satisfaction for different kinds of messages. This is the one attempted here. Not only is this useful from the point of view of design, it also helps clarify our intuitions about the process of deliberation involved since ideally the agents should act so as to “satisfy” the messages communicated in their system. We return to this point in §6.

2.2 Speech Act Theory

Speech Act Theory deals with natural language utterances. Initially, it was developed to deal with utterances, e.g., “I declare you man and wife,” that are not easily classified as being true or false, but rather are actions. Later it was extended to deal with all utterances, with the primary understanding that all utterances are *actions* of some sort or the other [Austin, 1962; Bach and Harnish, 1979; Searle, 1969]. A speech act is associated with at least three distinct actions: (1) a *locution*, i.e., the corresponding physical utterance, (2) an *illocution*, i.e., the conveying of the speaker’s intent to the hearer and (3) any number of *perlocutions*, i.e., actions that occur as a result of the illocution. For example, “shut the door” is a locution, which might be the illocution of a command to shut the door, and might lead to the perlocution of the listener getting up to shut the door. A speech act *per se* is identified with its associated illocution.

Speech acts may be classified into a small number of interesting classes, including *assertives*, *directives*, *commissives*, *permissives* and *prohibitives*. Briefly, assertives are statements of fact; directives are commands, requests or advice; commissives (e.g., promises) commit the speaker to a course of action; per-

missives issue permissions; and prohibitives take them away [Singh, 1991c]. These classes are said to have different *illocutionary forces*: they can be combined with the same *proposition* to yield different illocutions; e.g., “the door is shut” is an assertive and “shut the door” a directive, both of which apply to the same proposition, namely, that the door is shut—the assertive says that this proposition *is* true; the directive asks that it be *made* true [Searle, 1969].

2.2.1 Speech Act Theory in DAI

Speech Act Theory has also been found useful in DAI as a foundation for communication among agents. We agree with this view. There are two kinds of applications of Speech Act Theory in DAI. The first, and by far the more common one, uses it to motivate different *message types* for interactions among agents. The idea is that since agents can perform different kinds of speech acts, the language used for communication must allow different types of messages [Huhns *et al.*, 1990; Thomas *et al.*, 1990]. This is quite standard, and something we shall do ourselves. However, these proposals are informal—they rely on ones understanding of the labels used to understand the meanings of the different message types. The true semantics is embedded in the procedures that manipulate different messages.

The second kind of application of Speech Act Theory in DAI yields more sophisticated theories, which treat illocutions as linguistic actions and aim to describe the interactions of agents in terms of what they say to each other. These theories attempt to generalize linguistic theories of communication designed for human communication to the domain of DAI [Cohen and Levesque, 1988]. As a result, they tend to be somewhat top-heavy; e.g., they require that each of the agents involved have beliefs about the others' beliefs about their beliefs, and so on *ad infinitum*. It is known that such *mutual beliefs* are not achievable in practical systems [Fischer and Immerman, 1986; Halpern and Moses, 1987]. But more to the point, these theories suffer from being based on traditional formalizations of speech acts [Allen and Perrault, 1980]. Traditional formalizations are primarily concerned with identifying different kinds of illocutions. Thus these theories give the conditions under which saying “can you pass the salt?” is not a question, but rather a request; it is then an *indirect* speech act. An example of a condition for requests might be that the speaker and hearer mutually believe that the speaker has certain

intentions and beliefs. The phenomenon of indirect speech acts is, no doubt, of great importance in understanding natural language. But it is of simply no use in any conceivable DAI system: DAI systems can function quite well with just an artificial language that can be simply designed to be free of the ambiguities that these theories have been created to detect.

In a DAI scenario, we can have agents specify explicitly whether they intend their communication to be a request or a promise or an assertion or whatever. Thus the interesting part of the semantics of speech acts, as they may be applied in DAI, concerns what they cause to be done rather than whether they are interpreted to be of one kind or another. At least as a first approximation, we can assume that the illocutionary force of a message transmitted be just the one that is obvious from its syntax. Thus we will not consider indirect speech acts, whose primary role in human language seems to be to permit communication that in the direct form might be culturally unacceptable.

3 Formal Semantics for Communication

The formal model of this theory posits a set of possible worlds. As diagrammed in Figure 1, each possible world is in one of several states, and may develop in any of several ways depending on the agents' actions and, possibly, other events; e.g., the state of the world may change from t_0 to t_1 or t_2 if the given agent does action a , depending on what else happens at that time. Each of the different ways in which a world may develop is called a *scenario* and is equivalent to a possible course of events.

Using the idea described at the end of §2.2, we can consider messages as having a simple abstract syntax. A message, m , is a pair $\langle i, p \rangle$, where i identifies the illocutionary force, and p the proposition. Here i is an atomic symbol from the set {directive, commissive, permissive, prohibitive, assertive}; and p is a logical formula. Let 'comm' be a predicate that applies to two agents, and a message. 'Comm(x, y, m)' is true at a time-point if message m is uttered to agent y by agent x then. Let says-to(y, m) be the (only) action that agent x can perform to make comm(x, y, m) true.

This allows us to ignore details of message transmission and to focus on the *objective semantics* of speech acts. The idea of an objective semantics for speech acts has been introduced and defended in previous work [Singh,

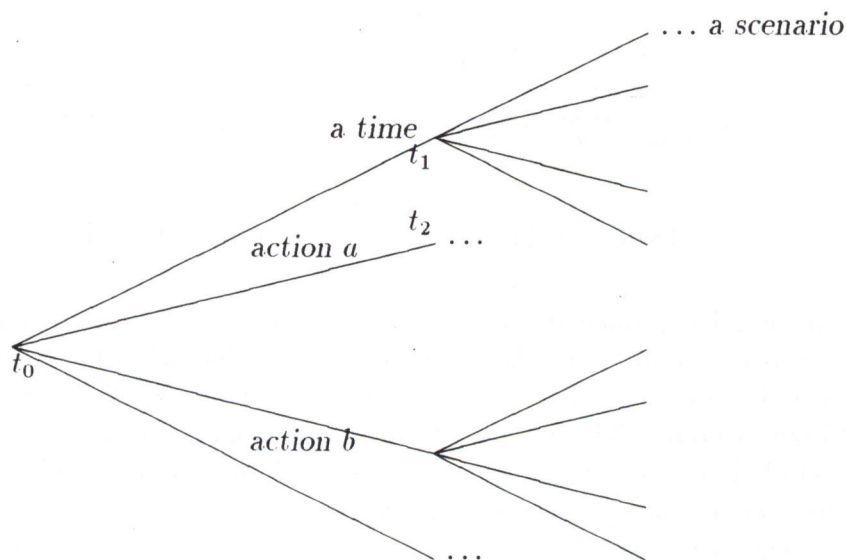


Figure 1: A World with a Branching History

1991c]. It considers, not the conditions under which a particular kind of speech act may be said to have occurred, but rather the conditions under which it may be said to have been *satisfied* objectively. A transmitted message may not always be satisfiable. In order to be able to talk of the satisfaction of messages explicitly, we introduce an operator **WSAT** that applies on formulas of the form ‘comm(x, y, m)’ and states that the corresponding message is *whole-heartedly* satisfied. Conditions of truth may be stated for **WSAT** applied to any kind of message, relative to a scenario and a time [Singh, 1991c].

The major classes of speech acts have been formalized in this way. As an example, a directive uttered by one agent to another is said to be satisfied along any course of events in which it becomes true, but in such a way that the listener intended it to become true and knew how to make it true; e.g., the directive “shut the door” would be satisfied if the door gets shut eventually, and until it is shut, the listener continuously intends to shut it and knows how to shut it (see Figure 2). The mere shutting of the door is

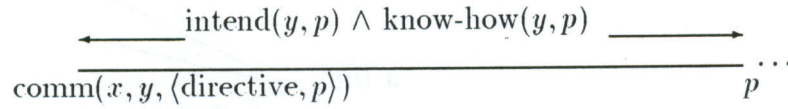


Figure 2: The Satisfaction Condition for Directives

not sufficient, since it could have happened by accident. The concepts of “intention” and “know-how” as used in this definition have themselves been formalized in the same model of action and time [Singh, 1990; Singh, 1991a; Singh, 1991b]. The details of those formalizations are too complex to be included here; however, that they are available is reason to be reassured that the crucial concepts are not undefined.

The formal language of this paper, \mathcal{L} , is CTL* (a propositional branching time logic [Emerson, 1989]) augmented with predicates for intention and know-how, and the operator WSAT. A formula can be any of the following: an atomic formula (ψ), a conjunction of formulae ($p \wedge q$), a negation of a formula ($\neg p$), an until-expression (pUq) or a predicate applied to some arguments, or a path-quantifier followed by a formula. A path-quantifier is one of A and E. A denotes “in *all* scenarios at the present time,” and $Ep \equiv \neg A\neg p$. Fp denotes “ p holds sometimes in the future on this scenario” and abbreviates “trueUp.” Gp denotes “ p always holds in the future on this scenario” and abbreviates “ $\neg F\neg p$.” Pp denotes “ p holds somewhere in the past.” Implication ($p \rightarrow q$) and disjunctions of formulae ($p \vee q$) are defined as the usual abbreviations.

The semantics of formulae in \mathcal{L} are given relative to a model as defined above and a world and time in it. $M \models_{w,t} p$ expresses “ M satisfies p at w, t .” $M \models_{S,t} p$ expresses “ M satisfies p at time t on scenario S ,” and is needed for some formulae. The satisfaction conditions for the propositional part of the language are standard and are not included here to save space. For reasons of space, only some of the possible kinds of speech acts are used in this paper. The formal semantics is given only for these (for a clarification of the semantics, please consult Figure 2 for directives).

1. $M \models_{w,t} Ap$ iff $(\forall S : S \in \mathbf{S}_{w,t} \rightarrow M \models_{S,t} p)$

2. $M \models_{S,t} pUq$ iff $(\exists t' : M \models_{S,t'} q \wedge (\forall t'' : t \leq t'' \leq t' \rightarrow M \models_{S,t''} p))$
3. $M \models_{S,t} Pp$ iff $(\exists t' : t' < t \wedge M \models_{S,t'} p)$
4. $M \models_{S,t} \text{WSAT}(\text{comm}(x, y, \langle \text{assertive}, p \rangle))$ iff $M \models_{S,t} p$
5. $M \models_{S,t} p$ iff $M \models_{w,t} p$, if p is not of the form qUr or $\langle a \rangle q$, and w is the (unique) world such that $S \in \mathbf{S}_{w,t}$
6. $M \models_{S,t} \text{WSAT}(\text{comm}(x, y, \langle \text{directive}, p \rangle))$ iff $(\exists t' \in S : t' \geq t \wedge M \models_{S,t'} p \wedge (\forall t'' : t \leq t'' < t' \rightarrow M \models_{S,t''} K_{how}(y, p) \wedge \text{intends}(y, p)))$
7. $M \models_{S,t} \text{WSAT}(\text{comm}(x, y, \langle \text{commissive}, p \rangle))$ iff $(\exists t' \in S : t' \geq t \wedge M \models_{S,t'} p \wedge (\forall t'' : t \leq t'' < t' \rightarrow M \models_{S,t''} K_{how}(x, p) \wedge \text{intends}(x, p)))$

4 Applying the Theory

The ways in which a theory of the semantics of speech acts, such as the one used here, may be applied in DAI are perhaps obvious. Such a theory can lead to a clearer understanding of the issues involved in the functioning of DAI systems and can be used in both their design and analysis. The formal model it supplies can be used to verify that a design has the desired properties. When a given system does not work as expected, this may be traced to a failure in meeting the semantics of some message. A designer may use the semantics by restricting the design to be such that only correct scenarios may be actualized. Thus the agents must act so that all messages exchanged in certain conditions be satisfied as time passes. For example, in cooperative systems, which are the majority of those likely to be designed by anyone, all requests that are somehow "reasonable" ought to be acceded to, all assertions ought to be true and all promises ought to be kept.

We would like that the design of a DAI system be such that only those scenarios be potentially actualized in it that are in some sense "good" or correct. An obvious requirement for correctness in our framework is that all the messages that arise on a given scenario be **WSAT** on it. In other words, the design should be constrained such that only those messages occur in it whose satisfaction can be guaranteed by it. There are two ways that a designer might go about enforcing these constraints on the design. One is to increase the capabilities of the agents appropriately, e.g., to increase the know-how

of the agents involved so that directives are more easily satisfied, to improve their perceptual and reasoning abilities so that their assertives may be true, or to limit what they may intend in different conditions so that their directives and commissives are achievable. The other approach is to treat messages, e.g., commissives, as setting up commitments that are later enforced, and limiting directives so that they occur only when a corresponding commitment has been made.

Once these design decisions have been made they can be stated declaratively in our formal language. One can then simply use standard methods in creating or testing designs. Such methods, which have already been developed for standard temporal logics include checking the satisfiability of sets of formulas (for us, constraints on the design) and for checking whether a given design satisfies a set of constraints (this is called *model checking*). These methods are described in [Emerson, 1989]. For the particular logic of this paper, such automated methods are not yet available. We now give some examples of formalizations of design constraints. It is by no means suggested that all these constraints make sense in all situations—they are stated below merely to exhibit the power of our theory. In the next section, we discuss an extended example that shows how constraints such as these may be used in DAI.

1. Intending Ones Directives:

The proposition of a directive should be intended by its issuer.

$$\text{comm}(x, y, \langle \text{directive}, p \rangle) \rightarrow \text{intends}(x, p)$$

2. Preference for Local Computation:

If an agent knows how to achieve a proposition by itself, it should not issue it as a directive.

$$K_{\text{how}}(x, p) \rightarrow \neg \text{comm}(x, y, \langle \text{directive}, p \rangle)$$

3. Weak Consistency for Directives:

A directive issued by an agent should not clash with the agent's own intentions; i.e., at least in some scenarios, the speaker's intentions and his directives should be compatible. This differs significantly from constraint 1. Constraint 1 says that the issuer intends the given directive;

this constraint says that all of the issuer's intentions are consistent with the directive.

$$\text{intends}(x, q) \wedge \text{comm}(x, y, \langle \text{directive}, p \rangle) \rightarrow \\ E[\text{WSAT} \text{comm}(x, y, \langle \text{directive}, p \rangle) \wedge Fq]$$

4. No Loss of Know-how for Issuers of Directives:

A directive issued by an agent should not clash with the issuer's own intentions and its satisfaction should not reduce the issuer's ability to achieve its intentions. That is, on all scenarios on which the directive is satisfied, the speaker knows-how to achieve its intentions.

$$\text{intends}(x, q) \wedge \text{comm}(x, y, \langle \text{directive}, p \rangle) \rightarrow \\ A[\text{WSAT} \text{comm}(x, y, \langle \text{directive}, p \rangle) \rightarrow K_{\text{how}}(x, q)]$$

5. Weak Consistency for Prohibitives:

A prohibitive is issued by an agent only if the agent itself does not intend that it be violated. That is, the agent who prohibits another from letting a certain condition occur should not itself try to make it happen. This is a minimal level of cooperation or rationality one expects from the issuers of prohibitions.

$$\text{comm}(x, y, \langle \text{prohibitive}, p \rangle) \rightarrow \\ \neg \text{intends}(x, p)$$

6. Prior Commitment:

A directive should be issued only after a conditional promise is given by the intended receiver that it would obey it. This solves for the issuer the problem of issuing only those directives that would be satisfied, provided the condition that promises are kept is enforced by the design. However, this condition is easier to enforce in a multiagent system, since it depends to a large extent on one agent (the issuer of the promise), rather than on several.

$$\text{comm}(x, y, \langle \text{directive}, p \rangle) \rightarrow \\ P[\text{comm}(y, x, \langle \text{commissive}, \text{comm}(x, y, \langle \text{directive}, p \rangle) \rightarrow Fp \rangle)]$$

5 The Contract Net: An Example

The Contract Net is among the most well-known and significant protocols in DAI [Davis and Smith, 1983]. While there are several variations possible, in its most basic form it may be described as in Figure 3. We are given a system with several agents. One of them has a task that it has to perform. It cannot do the task entirely locally and splits it into a number of subtasks. Let us consider one of the subtasks that cannot be performed locally. The agent now takes on the role of the *manager*. It sends out a *call for bids* to a subset of the other agents, describing the relevant subtask. Of the other agents, the ones who can and are willing to perform the advertised subtask respond by sending a *bid* to the manager. The manager evaluates the bids received, and selects one of them. It then sends a message *assigning* the subtask to that agent, who then becomes the *contractor*. The contractor performs the assigned task, possibly invoking other agents in the process. Finally, it communicates the *result* of performing the assigned task to the manager. The manager collects the results of all the subtasks of its original task and thus computes its result. If that task was assigned to it by some other agent, it then sends the result to it.

The key steps in the contract net from our point of view are the following: (1) the call for bids, (2) the bids, (3) the assignment of the task and (4) the result of the task. The processes of deciding whether to bid on a task and for evaluating the bids when they arrive can be safely abstracted out. These and other steps are local to each agent and involve knowledge of the domain in which the contract net is being used. We assume here that these processes, however designed and implemented, are available and are correct.

One can see almost instantaneously that the message with the result of the task should be classified as an assertive, because, in effect, it states that "the result is such and such." The message making the task assignment is a directive, since it asks the contractor to "do the task!" The message making the bid is a commissive, since it has the force of a conditional promise: "if asked to do the task, I will do it." Finally, the call for bids may itself be treated as a directive, because it has the effect of a request: "please speak up, if you will do this task."

This leads directly to an analysis in which these messages are nested, with the first one to occur being the outermost. The initial call for bids has the force of the following schematic message. If this message is satisfied by the

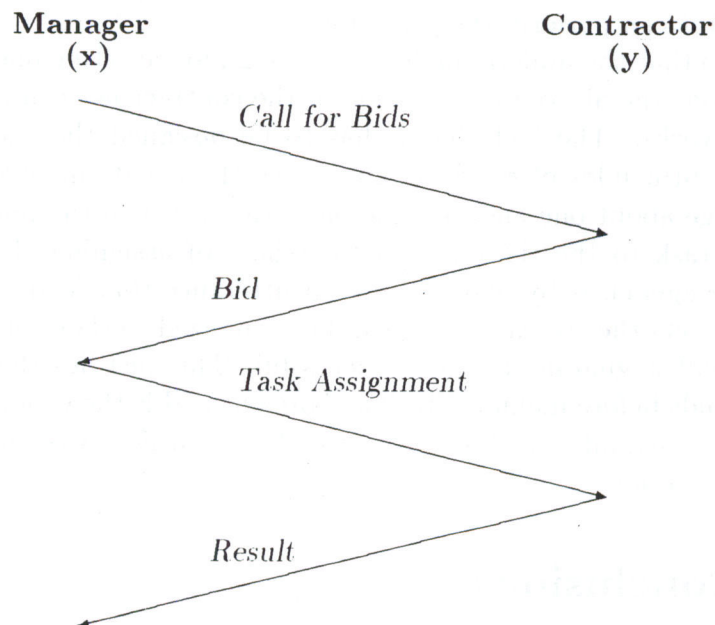


Figure 3: Messages Exchanged in the Contract Net

system, then we know it is performing correctly.

- $\langle \text{directive, comm}(y, x, \langle \text{commissive, comm}(x, y, \langle \text{directive, comm}(y, x, \langle \text{assertive, result}(T) \rangle \rangle \rangle) \rightarrow \text{F comm}(y, x, \langle \text{assertive, result}(T) \rangle \rangle \rangle \rangle)$

In other words, the call for bids is a directive asking the hearer to commit to sending the manager the result of the task, if the manager asks it to send it the result. The assertive with the result of the task is satisfied only if the contractor produces the right result, and does so while intending to and having the required know-how. The contractor must commit to producing the result, if assigned the task (the task can be assigned by sending a simpler message than in the above formalization by taking advantage of the context of communication, but it would logically have the same force as above). Thus

the task assignment directive is satisfied if the contractor produces the result when asked to. The call for bids is satisfied if the contractor makes the bid, provided it can perform the given task.

Given that the underlying heuristics, e.g., for selecting one of the bidders, are correct, the above formalization of the contract net can be used to show that it works. The fact that it has to be designed the way it is depends on some principles of good design. Since the agents involved have limited knowledge about one another, the only way in which the manager can send a given task to the right contractor (short of assigning the task to every available agent), is by first making an utterance that leads to an utterance that restricts the scenarios that can be actualized to those on which the task assignment is guaranteed to be successful. This justifies the sending of the call for bids before making a task assignment and is the canonical motivation for the constraint called *Prior Commitment*, which was introduced in the previous section.

6 Conclusions

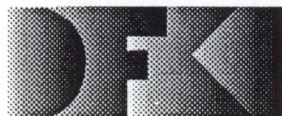
Though it is different from previous work on communication in DAI, our approach is compatible with, and complementary to, it. The main difference is in that we stress the objective semantics of messages as their most important aspect for DAI. Indeed, if in some system the language of communication cannot be constrained as we have assumed, it might be beneficial to use the traditional theories in determining the truth of $\text{comm}(x, y, m)$, i.e., in computing the illocutionary force of m . Our theory could then be applied at this stage.

We have considered only a few major classes of messages. As more refined categories of messages are considered, we will be able to determine their objective semantics with greater precision. We believe that the theory presented in this paper is a first, but important, step in developing a semantics for communication in DAI systems that would yield a rigorous foundation for their design and validation. Eventually formal methods, well-known in temporal logic as used in the validation and design of standard distributed systems may be extended to apply to distributed intelligent systems as well.

References

- [Allen and Perrault, 1980] Allen, James F. and Perrault, C. Raymond 1980. Analyzing intention in utterances. *Artificial Intelligence* 15:143–178.
- [Arni and others, 1990] Arni, Natraj and others, 1990. Overview of RAD: A hybrid and distributed reasoning tool. Technical Report ACT-RA-098-90, Microelectronics and Computer Technology Corporation, Artificial Intelligence Laboratory, Austin, TX.
- [Austin, 1962] Austin, John L. 1962. *How to do Things with Words*. Clarendon, Oxford, UK.
- [Bach and Harnish, 1979] Bach, Kent and Harnish, Robert M. 1979. *Linguistic Communication and Speech Acts*. MIT Press, Cambridge, MA.
- [Cohen and Levesque, 1988] Cohen, Philip R. and Levesque, Hector J. 1988. Rational interaction as the basis for communication. Technical Report 433, SRI International, Menlo Park, CA.
- [Davis and Smith, 1983] Davis, Randall and Smith, Reid G. 1983. Negotiation as a metaphor for distributed problem solving. *Artificial Intelligence* 20:63–109. Reprinted in *Readings in Distributed Artificial Intelligence*, A. H. Bond and L. Gasser, eds., Morgan Kaufmann, 1988.
- [Emerson, 1989] Emerson, E. A. 1989. Temporal and modal logic. In Leeuwen, J. van, editor 1989, *Handbook of Theoretical Computer Science*. North-Holland Publishing Company, Amsterdam, The Netherlands.
- [Fischer and Immerman, 1986] Fischer, Michael J. and Immerman, Neil 1986. Foundations of knowledge for distributed systems. In Halpern, Joseph Y., editor 1986, *Theoretical Aspects of Reasoning About Knowledge*. 171–185.
- [Halpern and Moses, 1987] Halpern, Joseph Y. and Moses, Yoram O. 1987. Knowledge and common knowledge in a distributed environment (revised version). Technical Report RJ 4421, IBM.
- [Huhns *et al.*, 1990] Huhns, Michael N.; Bridgeland, David; and Arni, Natraj 1990. A DAI communication aide. Technical Report ACT-RA-317-90,

- Microelectronics and Computer Technology Corporation, Artificial Intelligence Laboratory, Austin, TX.
- [McCarthy, 1979] McCarthy, John 1979. Ascribing mental qualities to machines. In Ringle, Martin, editor 1979, *Philosophical Perspectives in Artificial Intelligence*. Harvester Press. Page nos. from a revised version, issued as a report in 1987.
- [Perrault, 1987] Perrault, Raymond 1987. An application of default logic to speech act theory. Technical Report 90, Center for the Study of Language and Information, Stanford, CA.
- [Searle, 1969] Searle, John R. 1969. *Speech Acts*. Cambridge University Press, Cambridge, UK.
- [Singh, 1990] Singh, Munindar P. 1990. Group intentions. In *10th Workshop on Distributed Artificial Intelligence*.
- [Singh, 1991a] Singh, Munindar P. 1991a. Group ability and structure. In Demazeau, Y. and Müller, J.-P., editors 1991a, *Decentralized Artificial Intelligence, Volume 2*. Elsevier Science Publishers B.V. / North-Holland, Amsterdam, Holland.
- [Singh, 1991b] Singh, Munindar P. 1991b. A logic of situated know-how. In *National Conference on Artificial Intelligence (AAAI)*.
- [Singh, 1991c] Singh, Munindar P. 1991c. Towards a formal theory of communication for multiagent systems. In *International Joint Conference on Artificial Intelligence*.
- [Thomas *et al.*, 1990] Thomas, Becky; Shoham, Yoav; and Schwartz. Anton 1990. Modalities in agent-oriented programming. Computer Science Department, Stanford University.



**Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH**

**DFKI
-Bibliothek-
PF 2080
6750 Kaiserslautern
FRG**

DFKI Publikationen

Die folgenden DFKI Veröffentlichungen oder die aktuelle Liste von erhältlichen Publikationen können bezogen werden von der oben angegebenen Adresse.

Die Berichte werden, wenn nicht anders gekennzeichnet, kostenlos abgegeben.

DFKI Publications

The following DFKI publications or the list of currently available publications can be ordered from the above address.

The reports are distributed free of charge except if otherwise indicated.

DFKI Research Reports

RR-90-01

Franz Baader: Terminological Cycles in KL-ONE-based Knowledge Representation Languages
33 pages

RR-90-02

Hans-Jürgen Bürckert: A Resolution Principle for Clauses with Constraints
25 pages

RR-90-03

Andreas Dengel, Nelson M. Mattos: Integration of Document Representation, Processing and Management
18 pages

RR-90-04

Bernhard Hollunder, Werner Nutt: Subsumption Algorithms for Concept Languages
34 pages

RR-90-05

Franz Baader: A Formal Definition for the Expressive Power of Knowledge Representation Languages
22 pages

RR-90-06

Bernhard Hollunder: Hybrid Inferences in KL-ONE-based Knowledge Representation Systems
21 pages

RR-90-07

Elisabeth André, Thomas Rist: Wissensbasierte Informationspräsentation:
Zwei Beiträge zum Fachgespräch Graphik und KI:

1. Ein planbasierter Ansatz zur Synthese illustrierter Dokumente
2. Wissensbasierte Perspektivenwahl für die automatische Erzeugung von 3D-Objektdarstellungen

24 pages

RR-90-08

Andreas Dengel: A Step Towards Understanding Paper Documents
25 pages

RR-90-09

Susanne Biundo: Plan Generation Using a Method of Deductive Program Synthesis
17 pages

RR-90-10

Franz Baader, Hans-Jürgen Bürckert, Bernhard Hollunder, Werner Nutt, Jörg H. Siekmann: Concept Logics
26 pages

RR-90-11

Elisabeth André, Thomas Rist: Towards a Plan-Based Synthesis of Illustrated Documents
14 pages

RR-90-12

Harold Boley: Declarative Operations on Nets
43 pages

RR-90-13

Franz Baader: Augmenting Concept Languages by Transitive Closure of Roles: An Alternative to Terminological Cycles
40 pages

RR-90-14

Franz Schmalhofer, Otto Kühn, Gabriele Schmidt: Integrated Knowledge Acquisition from Text, Previously Solved Cases, and Expert Memories
20 pages

RR-90-15

Harald Trost: The Application of Two-level Morphology to Non-concatenative German Morphology
13 pages

RR-90-16

Franz Baader, Werner Nutt: Adding Homomorphisms to Commutative/Monoidal Theories, or: How Algebra Can Help in Equational Unification
25 pages

RR-90-17

Stephan Busemann: Generalisierte Phasenstrukturgrammatiken und ihre Verwendung zur maschinellen Sprachverarbeitung
114 Seiten

RR-91-01

Franz Baader, Hans-Jürgen Bürckert, Bernhard Nebel, Werner Nutt, Gert Smolka: On the Expressivity of Feature Logics with Negation, Functional Uncertainty, and Sort Equations
20 pages

RR-91-02

Francesco Donini, Bernhard Hollunder, Maurizio Lenzerini, Alberto Marchetti Spaccamela, Daniele Nardi, Werner Nutt: The Complexity of Existential Quantification in Concept Languages
22 pages

RR-91-03

B.Hollunder, Franz Baader: Qualifying Number Restrictions in Concept Languages
34 pages

RR-91-04

Harald Trost: X2MORF: A Morphological Component Based on Augmented Two-Level Morphology
19 pages

RR-91-05

Wolfgang Wahlster, Elisabeth André, Winfried Graf, Thomas Rist: Designing Illustrated Texts: How Language Production is Influenced by Graphics Generation.
17 pages

RR-91-06

Elisabeth André, Thomas Rist: Synthesizing Illustrated Documents A Plan-Based Approach
11 pages

RR-91-07

Günter Neumann, Wolfgang Finkler: A Head-Driven Approach to Incremental and Parallel Generation of Syntactic Structures
13 pages

RR-91-08

Wolfgang Wahlster, Elisabeth André, Som Bandyopadhyay, Winfried Graf, Thomas Rist: WIP: The Coordinated Generation of Multimodal Presentations from a Common Representation
23 pages

RR-91-09

Hans-Jürgen Bürckert, Jürgen Müller, Achim Schupeta: RATMAN and its Relation to Other Multi-Agent Testbeds
31 pages

RR-91-10

Franz Baader, Philipp Hanschke: A Scheme for Integrating Concrete Domains into Concept Languages
31 pages

RR-91-11

Bernhard Nebel: Belief Revision and Default Reasoning: Syntax-Based Approaches
37 pages

RR-91-12

J.Mark Gawron, John Nerbonne, Stanley Peters: The Absorption Principle and E-Type Anaphora
33 pages

RR-91-13

Gert Smolka: Residuation and Guarded Rules for Constraint Logic Programming
17 pages

RR-91-14

Peter Breuer, Jürgen Müller: A Two Level Representation for Spatial Relations, Part I
27 pages

RR-91-15

Bernhard Nebel, Gert Smolka: Attributive Description Formalisms ... and the Rest of the World
20 pages

RR-91-16

Stephan Busemann: Using Pattern-Action Rules for the Generation of GPSG Structures from Separate Semantic Representations
18 pages

RR-91-17

Andreas Dengel & Nelson M. Mattos: The Use of Abstraction Concepts for Representing and Structuring Documents
17 pages

RR-91-19

Munindar P. Singh: On the Commitments and Precommitments of Limited Agents
15 pages

RR-91-20

Christoph Klauck, Ansgar Bernardi, Ralf Legleitner: FEAT-Rep: Representing Features in CAD/CAM
48 pages

RR-91-22

Andreas Dengel: Self-Adapting Structuring and Representation of Space
27 pages

RR-91-23

Michael Richter, Ansgar Bernardi, Christoph Klauck, Ralf Legleitner: Akquisition und Repräsentation von technischem Wissen für Planungsaufgaben im Bereich der Fertigungstechnik
24 Seiten

RR-91-24

Jochen Heinsohn: A Hybrid Approach for Modeling Uncertainty in Terminological Logics
22 pages

RR-91-25

Karin Harbusch, Wolfgang Finkler, Anne Schauder: Incremental Syntax Generation with Tree Adjoining Grammars
16 pages

RR-91-26

M. Bauer, S. Biundo, D. Dengler, M. Hecking, J. Koehler, G. Merziger: Integrated Plan Generation and Recognition - A Logic-Based Approach -
17 pages

RR-91-27

A. Bernardi, H. Boley, Ph. Hanschke, K. Hinkelmann, Ch. Klauck, O. Kühn, R. Legleitner, M. Meyer, M. M. Richter, F. Schmalhofer, G. Schmidt, W. Sommer: ARC-TEC: Acquisition, Representation and Compilation of Technical Knowledge
18 pages

RR-91-30

Dan Flickinger, John Nerbonne: Inheritance and Complementation: A Case Study of Easy Adjectives and Related Nouns
39 pages

DFKI Technical Memos
TM-89-01

Susan Holbach-Weber: Connectionist Models and Figurative Speech
27 pages

TM-90-01

Som Bandyopadhyay: Towards an Understanding of Coherence in Multimodal Discourse
18 pages

TM-90-02

Jay C. Weber: The Myth of Domain-Independent Persistence
18 pages

TM-90-03

Franz Baader, Bernhard Hollunder: KRIS: Knowledge Representation and Inference System -System Description-
15 pages

TM-90-04

Franz Baader, Hans-Jürgen Bürckert, Jochen Heinsohn, Bernhard Hollunder, Jürgen Müller, Bernhard Nebel, Werner Nutt, Hans-Jürgen Profitlich: Terminological Knowledge Representation: A Proposal for a Terminological Logic
7 pages

TM-91-01

Jana Köhler: Approaches to the Reuse of Plan Schemata in Planning Formalisms
52 pages

TM-91-02

Knut Hinkelmann: Bidirectional Reasoning of Horn Clause Programs: Transformation and Compilation
20 pages

TM-91-03

Otto Kühn, Marc Linster, Gabriele Schmidt: Clamping, COKAM, KADS, and OMOS: The Construction and Operationalization of a KADS Conceptual Model
20 pages

TM-91-04

Harold Boley: A sampler of Relational/Functional Definitions
12 pages

TM-91-05

Jay C. Weber, Andreas Dengel, Rainer Bleisinger: Theoretical Consideration of Goal Recognition Aspects for Understanding Information in Business Letters
10 pages

TM-91-08

Munindar P. Singh: Social and Psychological Commitments in Multiagent Systems
11 pages

TM-91-09

Munindar P. Singh: On the Semantics of Protocols Among Distributed Intelligent Agents
18 pages

TM-91-10

Béla Buschauer, Peter Poller, Anne Schauder, Karin Harbusch: Tree Adjoining Grammars mit Unifikation
149 pages

TM-91-11

Peter Wazinski: Generating Spatial Descriptions for Cross-modal References
21 pages

DFKI Documents**D-89-01**

Michael H. Malburg, Rainer Bleisinger:
HYPERBIS: ein betriebliches Hypermedia-Informationssystem
43 Seiten

D-90-01

DFKI Wissenschaftlich-Technischer Jahresbericht 1989
45 pages

D-90-02

Georg Seul: Logisches Programmieren mit Feature-Typen
107 Seiten

D-90-03

Ansgar Bernardi, Christoph Klauck, Ralf Legleitner: Abschlußbericht des Arbeitspaketes PROD
36 Seiten

D-90-04

Ansgar Bernardi, Christoph Klauck, Ralf Legleitner: STEP: Überblick über eine zukünftige Schnittstelle zum Produktdatenaustausch
69 Seiten

D-90-05

Ansgar Bernardi, Christoph Klauck, Ralf Legleitner: Formalismus zur Repräsentation von Geo-metrie- und Technologieinformationen als Teil eines Wissensbasierten Produktmodells
66 Seiten

D-90-06

Andreas Becker: The Window Tool Kit
66 Seiten

D-91-01

Werner Stein, Michael Sintek: Relfun/X - An Experimental Prolog Implementation of Relfun
48 pages

D-91-03

Harold Boley, Klaus Elsbernd, Hans-Günther Hein, Thomas Krause: RFM Manual: Compiling RELFUN into the Relational/Functional Machine
43 pages

D-91-04

DFKI Wissenschaftlich-Technischer Jahresbericht 1990
93 Seiten

D-91-06

Gerd Kamp: Entwurf, vergleichende Beschreibung und Integration eines Arbeitsplanerstellungssystems für Drehteile
130 Seiten

D-91-07

Ansgar Bernardi, Christoph Klauck, Ralf Legleitner
TEC-REP: Repräsentation von Geometrie- und Technologieinformationen
70 Seiten

D-91-08

Thomas Krause: Globale Datenflußanalyse und horizontale Compilation der relational-funktionalen Sprache RELFUN
137 pages

D-91-09

David Powers and Lary Reeker (Eds):
Proceedings MLNLO '91 - Machine Learning of Natural Language and Ontology
211 pages

Note: This document is available only for a nominal charge of 25 DM (or 15 US-\$).

D-91-10

Donald R. Steiner, Jürgen Müller (Eds.)
MAAMAW '91: Pre-Proceedings of the 3rd European Workshop on „Modeling Autonomous Agents and Multi-Agent Worlds“
246 pages

Note: This document is available only for a nominal charge of 25 DM (or 15 US-\$).

D-91-11

Thilo C. Horstmann: Distributed Truth Maintenance
61 pages

D-91-12

Bernd Bachmann:
HieraCon - a Knowledge Representation System with Typed Hierarchies and Constraints
75 pages

D-91-13

International Workshop on Terminological Logics
Organizers: Bernhard Nebel, Christof Peltason, Kai von Luck
131 pages

D-91-14

Erich Achilles, Bernhard Hollunder, Armin Laux, Jörg-Peter Mohren: KRIS: Knowledge Representation and Inference System - Benutzerhandbuch -
28 Seiten

