ASCRIBING PLANS TO AGENTS
Preliminary Report

Kurt Konolige and Martha E. Pollack"
Artificial Intelligence Center *and*
("enter for the Study of Language and Information
SRI International
Menlo Park, California 94025/USA

Abstract

Intelligent agents who are situated in nuiltiagent domains must reason about one anotIters' actions and plans. Following the tradition of earlier work in AI, we present a model of plan recognition as belief and intention ascription, an inherently defeasible reasoning process. However, we encode this process using a direct argumentation system. Within this system, we can make explicit statements about why one candidate ascription should be preferred over another. And we can avoid the overly strong assumption that the actor's plan is correct from the perspective of the observer—an assumption that was necessary in previous formalizations of plan recognition.

1 Introduction

Intelligent agents who are situated in muJtiagent, domains must reason about one anothers' actions and plans. Many AT models of plan recognition have viewed it as a process of belief and intention ascription [Allen, 1983, Litman and Allen, 1987, Schmidt *el ai,* 1978, Sidner, 1985]. Kautz has recently criticized this body of research for its failure to provide a formal analysis of the defeasible reasoning inherent in plan recognition. In particular, he notes that in the existing work "[o]nly a space of *possible* inferences is outlined, and little or nothing is said about why one should infer one conclusion over another, or what one should conclude if the situation is truly ambiguous" [Kautz, In Press]. Kautz himself pro vides an elegant formalization of plan recognition, stated in terms of circumscription. However, his account relies upon strong assumptions; that the agent performing the plan recognition (the *observer)* has complete knowledge of the domain, and that the agent whose plan is being inferred (the *actor)* has a correct plan. Pollack, in research on plan recognition in discourse understanding, has shown that these assumptions are too strong for any realistic, useful model of plan recognition [Pollack, 1986, Pollack, In press].

*This research was supported by the Office of Naval Research under Contract No, N00014--85 C 0251, by a contract with the Nippon Telegraph and Telephone Corporation, and by a gift from the System Development Foundation.

It is not obvious how to remove the strong assumptions from Kautz's model.[1] Consequently, we have been developing an alternative formalization of plan recognition, one in which it is not necessary to assume that the actor's plan is correct from the perspective of the observer, but in which it is possible to specify in a precise way why certain conclusions should be preferred over others. This paper consitutes a preliminary report on this approach.

In our view, plan recognition, like model-based recognition tasks in perceptual domains, involves the interaction of two types of information: local cues derived from the data, and the coherence of local information in a global structure. In plan recognition, the data comprise (1) the actions of the actor, which, in the cases we will consider, are restricted to being utterances; and (2) previously held mutual beliefs. From this data, the observer can initially derive local cues about some of the actor's beliefs and intentions. During plan recognition the observer ascribes to the actor additional beliefs and intentions (or what we will call "plan fragments"). Typically, the ascription is process is local, in the sense that, at each point in the process, the observer uses a. small subset of already ascribed beliefs and plan fragments as the basis of further ascription. The ascription of beliefs and the ascription of plan fragments interact with one another. For example, if the observer *O* can ascribe to the actor *A* a belief that another agent *B* has some object *Ohj* that *A* wants, then the observer is justified in ascribing to *A* a plan fragment that involves asking *B* for *Ohj.* But if there is no need to ascribe this plan fragment to A, then *O* will typically not ascribe the belief either.

Both belief and intention ascription are defeasible. Moreover, there will often be conflicts among the possible ascriptions. We maintain that much of the complexify and subtlety of the plan recognition process comes from ways in which these conflicts are adjudicated. We give some important general principles of conflict resolution, but do not assert that these form a complete catalog: only more experience with plan recognition systems in actual use will reveal deficiencies here.

[1] Kautz suggests the introduction of an "error" plan that will be inferred whenever one of the assumptions is violated. But, in general, this is insufficient, since agents need to be able not only to reason *that* a plan is incorrect, but also to reason about what makes it so.

What role does global coherence play in the plan recognition process? It. is clear that local cues must, give rise to a globally coherent plan in order for such a plan to be attributed, but the relative strength of local vs. global information has yet to be determined. We conjecture, however, that plan recognition is essentially a "bottom-up" recognition process, with global coherence used mostly as an evaluative measure, to eliminate ambiguous plan fragments that emerge from local cues. In this paper we concentrate on the formal definition of local ascriptions, along with their interactions. For this HI r pose we use the direct argumentation system ARCH Konolige, 1989]. One of the more useful features of this system is that it permits explicit statements about the resolution of conflicts between candidate ascriptions.

2 Working Assumptions

To ground our discussion, we will draw our examples in this paper from the "Robot Delivery Domain" (henceforth, *HDD),* in wliich a Flakey, a mobile robot, is presumed to roam the halls of an office, delivering objects to people on request. To behave intelligently, Flakey must attempt to recognize the plans of the people who make requests of him: in our examples, he is the observer, and the requesting agent is the actor. While we neither assume that Flakey has complete knowledge of the domain, nor that the people directing him always have correct plans, there arc certain other simplifications that we make in order to focus our work.

First, we simplify the form of some of the intentions that Flakey reasons about. The plans he infers may include actions done by the requesting agent, as well as actions done by Flakey himself. Strictly speaking, one agent cannot have an intention that has as its object an action of another agent—although one may have an intention to cause some other agent to perform an action [Castaneda, 1975]. We will finesse this distinction in this paper.

Second, we bound the number of nested beliefs that Flakey needs to consider. In our examples, Flakey can reason about his own beliefs, and about the actor's beliefs about, the world or about other agents. However, we rule out the need for Flakey to reason explicitly about, the actor's beliefs about Flakey's own beliefs. This amounts to an assumption that Flakey believes that the actor's beliefs about his (Flakey's) own beliefs are correct. This assumption is not essential to our account, but it simplifies the subsequent discussion.

Third, we avoid the "Dudley Doright" problem [McDermott, 1982], by assuming that there is a distinction between what agents expect to achieve as a result of their intended actions, and what they believe is or will be true independent of their actions. We use the term *achieve* to represent the former, and *believe* to represent the latter. If an agent either will achieve some proposition or believes it will be true, then we say that he *expects* it.

Finally, while recognizing that the temporal aspects of a plan can be the subject of complex reasoning in their own right, we simplify the structure of time by adopting the situation calculus view of single atomic actions occuring between discrete situations, with propositions being true at. a situation.

3 The Structure of Plans

Because plan ascription involves reasoning about the beliefs and intentions of agents, we define a language £ containing operators which express these concepts:

INT(a,o) agent *a* intends plan fragment o

BFL(<7, p) agent *a* believes proposition *p*

A(_TI(a,p) agent a will achieve proposition *p*

EXP(ri,p) agent *a* expects proposition *p* to be true

These, together with truth (T), constitute the operators of *£.* £ also contains constants denoting the agents in the domain: *F* denotes Flakey, and *11* denotes Harry. Also, *11* denotes a (particular) report and O denot.es Harry's of lice; the initial situation is S_0, followed by *S* and .SV

In our examples, Flakey must infer Harry's plans, which may include actions done by Flakey. flakey thus needs to represent his own beliefs about the world, as well as his beliefs about Harry's intentions and beliefs. In turn, Harry's beliefs may themselves concern the behefs and intentions of Flakey. In a more general setting we would need to represent all of these constructs; here, however, we bound the number of nested beliefs that Flakey needs to reason about, as noted above. Flakey represents the truth of the proposition *p* as T(;>), and he represents Harry's belief in *p* as BFL(i/,p). Similarly, we bound the number of nested intentions: Flakey need only reason about Harry's intentions (usually that Flakey perform some act), which we represent as 1NT(//,Q).

Finally, we need constructions for propositional expressions (pexp) and plan-fragment, expressions (pian-exp), which can serve as arguments to the operators of £. Propositional expressions are formed from a property name and parameters, which usually include parameters for the situation (or time) at which the property holds. For our examples, we need only two properties:

ar(a,/,/) agent *a* is at location / at time /

has(a, o,t) agent, *a* possesses object *o* at time /

There are two kinds of plan-fragment expressions. The simpler kind are similar to propositional expressions: they are formed from an action name and parameters, which, again, usually include parameters for the situation at wliich the action commences. Plan-fragment expressions also include a parameter for the agent performing the action. We can call this simple kind of plan-fragment expression an action expression *(actexp).* In the *HDD* we need the following actions:

gct(a,o,t) agent *a* gets object *o* at time *t*

niovc(a, d,t) agent *a* moves to location *d* at time *t*

tra.ns(a,o,d,t) agent *a* transports object *o* to location *d* at time /

bring(a,o,b,t) agent *a* brings object *o* to agent *b* time /

We allow implicit coercion of actexp's so that they can serve as propositions. Thus we can can say that Flakey

believes that an action specified by the expression α occurred (or will occur) by using $T(\alpha)$; that the agent, a believes this by $BEL(a, \alpha)$. Of course, we can also say that Flakev belives that a intends to perform a by $INT(a, \alpha)$.

More complex plan-fragment expressions can be built using the operators *BY* and *TO:*

B}'(;ictcxp, actcxp, pcxp) the complex plan fragment, consisting of doing the second *nctexp,* by doing the first *nctcxp* while the *pcxp* is true

TO(cictcxp, pcxp) the complex plan fragment., consisting of making the *pcxp* true by doing the a*ctexp*

When we write $INT'(a, BY(\alpha, \beta, p))$, we mean that agent *a* expects p and intends to do B moreover, *a* intends to do β by doing α. When we write INT(a, $TO(\alpha, p)$), we mean that a intends α, as a way of achieving p. As before, we allow for implicit coercion from p/anexp\s to pewp's. This enables us to specify that an agent, a believes that a given relationship holds among actions: for example, BEL(a, $BY(\alpha, \beta, p)$) denotes a's belief that if p is true, the occurrence of o guarantees the occurrence of β. A similar coercion applies to *TO,* so, for example, we represent Flakey's belief that when an agent gets an object, he then has the object, as follows:

$$T(TO(get(a, o, S_0), has(a, o, S_1))).$$

Axioms such as this will be introduced as needed in Section 5.

4 An Argumentation System for Plan Ascription

In this section we give a brief overview of the defeasible argumentation system AUGII [Konolige, 1989]. ARGII is a formal system, in the sense that its elements are formal objects, and the processes that manipulate them could be implemented on a computer. If is similar in many respects to so-called justification-based Truth Maintenance Systems [Doyle, 1979], but differs in the diversity of argumentation allowed, and the fact, that arguments for a proposition and its negation may coexist without, contradiction. It also differs from formal nonmonotonic logic approaches, such as circumscription [McCarthy, 1984] or default logie [Reiter, 1980], in that it makes arguments the direct subject matter of the system. Finally, it differs from other direct, argumentation systems (for example, those of [Horty *et al.,* 1988, Loui, 1987, Poole, 1985]) in that it has an explicit notion of argument *support* independent of belief, and allows a flexible specification of domain-dependent conditions of adjudication among arguments.

The purpose of argumentation is to formulate connections between propositions, so that an agent can come to plausible conclusions based on initial data. Formally, an argument is a relation between a set of propositions (the *premises* of the argument), and another set of propositions (the *conclusion* of the argument). For example, an argument that Flakey's beliefs are also Harry's beliefs could be stated as:

$$T(p) \xrightarrow{beiaSC} BEL(H, p) \qquad (2)$$

Arguments are normally specified by schemata, in which we allow free variables to stand for arbitrary terms. Here *p* is a schema variable for an arbitrary proposifional expression. We will generally use lowercase roman letters to indicate schema variables. Classes of arguments in ARCH are given names. All arguments of a schema must belong to the same class: the schema above defines part of the be/asr (belief ascription) class of arguments.

In the process of argumentation, we start with an initial set of facts (which we call a *world)* describing the situation in question. We then use argument schemata to construct plausible arguments based on the initial facts. The process of deciding which arguments are valid ones from a given world makes direct argumentation systems interesting and complex. In ARCH, the concepts by which we express the validity of the conclusions of arguments are *support, conflict, dejeat* and *acceptance.*

Support. The conclusion of an argument is supported by a world if the premises of the argument are supported, provided the argument is not defeated (see below). All initial facts of a world are supported. To eliminate circularity in the support relation, we demand that, no argument be used in the support of its own premises. Note thai a proposition and its negation may both be supported, if there are valid arguments for both of them.

Conflict. Two propositions are said to conflict, if they are in some way opposites. Generally, a proposition and its netgation confliet,. For a particular domain we will often specify explicitly which propositions conflict., without necessarily deriving the negations. A proposition that does not conflict with any supported proposition in a world is called *uncontested.*

Aeooptaiico. If a proposition is supported by an argument, whose premises are accepted, and it is uncontested, then it, is accepted. By definition, a proposition and its negation can never be simultaneously accepted. We consider accepted propositions to be valid conclusions about a world. By fiat, all initial facts of a. world are accepted, and cannot be contested.

Defeat. An argument is defeated in a world if the conditions of its defeat hold in that, world. We will define such conditions below; generally, they are used to specify which of two arguments supporting conflicting propositions is to prevail. Defeat is what makes arguments defeasible, and is one of the most complicated and interesting parts of defining a domain.

Unfortunately, these definitions do not guarantee that for any world there will be a single consistent assignment, of support, and acceptance. For example, it is possible for the arguments to two conflicting propositions to be mutually defeating, so that we could consistently assign acceptance to either of the propositions. In this case, there is a genuine ambiguity in the acceptance process, and although ARCH allows us to conclude either one, we prefer to remain skeptical of both. This problem is similar to that of deciding among multiple competing extensions in typical nonmonotonic formalisms [Horty *et a/.,* 1988].

5 Ascribing Beliefs and Plan Fragments

In our model of plan recognition, the initial world consists of the beliefs and intentions that can be directly "read off" the actor's utterances, along with any previously held mutual beliefs.[1] The argumentation system is then used to discover the actor's intended plan. It does this by applying arguments to discover plan fragments that can be ascribed to the agent, some of which will eventually be labeled *accepted* by the argumentation system. We now present some specific arguments for local ascription, and illustrate their use with some examples from the *HDD.*

5.1 Belief Ascription Arguments

The problem of deciding what beliefs to ascribe to agents is a complex one, and beyond the scope of this paper. Rather, we will use the simple defeasible rule that agents are likely to know the true facts about the general effeets and relations of actions, as well as the particular facts that are true of a given sit nation:

$$T(P) \xrightarrow{bdtusc} \mathbf{BEL}(a, p), \quad (3)$$

where a is an agent, and p is a *pexj).*

Since the reasoning is being done from Flakev's point, of view, this essentially ascribes to other agents all of Plakeys beliefs about, the domain. Belief ascription is defeasible, so specific information about, other agents' beliefs can override this rule.

The other rules of belief ascription that we need encode the fact that either believinf or achieving p is a. wav of expecting p:

$$\begin{aligned} \mathrm{BEL}(a,p) &\xrightarrow{belasc} \mathrm{EXP}(a,p) \\ \mathrm{ACH}(a,p) &\xrightarrow{belasc} \mathrm{EXP}(a,p) \end{aligned} \quad \textbf{(4)}$$

5.2 Plan-Fragment Ascription Arguments

These arguments are used to ascribe *BY* or *TO* [dan fragment,s to the actor, based upon what, the observer already believes lie intends. We start with an argument, that ascribes a *TO* fragment,:

$$\begin{aligned} &\mathrm{BEL}(a, TO(\alpha,p)), \mathrm{INT}(a,\alpha), \mathrm{ACH}(a,p) \xrightarrow{to1} \\ &\mathrm{INT}(a, TO(\alpha,p)) \end{aligned} \quad (5)$$

This schema says that, if an agent, a believes that p is an effect, of performing α, and he intends to do α and to achieve ;>, then it is plausible that his reason for doing α is to achieve p. Instances of this rule are used to coalesce plan fragments. A similar rule coalesces fragments involving the *BY* relation:

$$\begin{aligned} &\mathrm{BEL}(a, BY(\alpha,\beta,p)), \mathrm{INT}(a,\alpha), \mathrm{INT}(a,\beta), \mathrm{EXP}(a,p) \xrightarrow{by1} \\ &\mathrm{INT}(a, BY(\alpha,\beta,p)) \end{aligned} \quad \textbf{(0)}$$

[2]This then implies that the agents in our domain com municate with one another in a formal language that precludes indirection. In a model of plan recognition in *natural*-language discourse, the initial facts would he that the utterances themselves were observed, and one of the tasks for plan recognition would be the derivation of the intentions and beliefs encoded in those utterances.

Note that, in the *BY* relation, the enabling condition p is expected: it can be either already believed to hold, or achieved by some other plan fragment,.

EXAMPLE 5.1 Harry requests that Flakey get the report so that Flakev will have it.[3] The initial facts consist of the following intentions and beliefs derived directly from Harry's request:

$$\begin{aligned} &\mathrm{INT}(H, get(F,R,S_0)) \\ &\mathrm{ACH}(H, has(F,R,S_1)) \end{aligned} \quad \textbf{(7)}$$

as well as Flakev's *a priori* beliefs about, domain plan relations, one of which is relevant, here:

$$\mathrm{T}(TO(get(F,R,S_0), has(F,R,S_1))) \; . \quad (8)$$

From this world, the *belasc* schema is used to generate support for $\mathrm{BEL}(H, TO(get(F,R,S_0), has(F,R,S_1)))$. The *to1* schema can then be used to support $\mathrm{INT}(H, TO(get(F,R,S_0), has(F,R,S_1)))$. Since there is no conflict, these propositions are also accepted.

Additional arguments extend ascribed plan fragments. The following schema provides support, for an intended *TO* relation when its precipitating act.ion is observed:

$$\begin{aligned} &\mathrm{BEL}(a, TO(\alpha,p)), \mathrm{INT}(a,\alpha) \xrightarrow{to2} \\ &\mathrm{INT}(a, TO(\alpha,p)), \mathrm{ACH}(a,p) \end{aligned} \quad (9)$$

This rule is intuitivelv weaker than the *tol* rule because it, uses less evidence. Weaker still would be an argumerit in which an intended *TO* relation was inferred simply from the expectation of its effect,s. (This would correspond to a traditional "backward chaining rule[1]", while the *to'2* schema corresponds to a "forward-chaining rule.") The reason this would be such a weak rule is that there are usually very many ways of achieving a proposition, and context-dependent information is required to focus on the likelv intended wav. We will return to this problem below (Section 5.*i).

There are also fragment, extension arguments for the *BY* rehat.ion:

$$\begin{aligned} &\mathrm{BEL}(a, BY(\alpha,\beta,p)), \mathrm{INT}(a,\alpha), \mathrm{EXP}(a,p) \xrightarrow{by2} \\ &\mathrm{INT}(a, BY(\alpha,\beta,p)), \mathrm{INT}(a,\beta) \\ &\mathrm{BEL}(a, BY(\alpha,\beta,p)), \mathrm{INT}(a,\beta), \mathrm{EXP}(a,p) \xrightarrow{by3} \\ &\mathrm{INT}(a, BY(\alpha,\beta,p)), \mathrm{INT}(a,\alpha) \\ &\mathrm{BEL}(a, BY(\alpha,\beta,p)), \mathrm{INT}(a,\alpha), \mathrm{INT}(a,\beta) \xrightarrow{by4} \\ &\mathrm{INT}(a, BY(\alpha,\beta,p)), \mathrm{EXP}(a,p) \end{aligned} \quad (10)$$

~ll may perhaps seem odd that Harry would be so explicit in specifying his request. To some extent this is true, l)iit wc intend this very simple example to illustrate the basic elements of belief ascription and recognition of the relation among known intended actions. Also, it is worth noting that study of human conversation shows that in fact people provide a great deal of explicit information about their plans when they are communicating [Pollack, 1986]. In addition, one might imagine a variant in which Harry says "Get the report so that you'll have it, when Sue calls for it," a quite natural-sounding request.

We are generally willing to infer an intended BY relation 1 rom knowledge of any two of its components. With more contextual information, it may also be possible to infer tins from one component,.

5.3 Cont.ext-Dojxnideiit Ascription

Consider the following example:

EXAMI'LF. 5.2 Harry requests that Flakey get the report so that he (Harry) can have it, i.e., the intial world includes:

$$\begin{array}{l}\mathrm{INT}(H, get(F, R, S_0)) \\ \mathrm{ACH}(H, has(H, R, S)\end{array} \quad (11)$$

The obvious implication, in the context of Flakey as a delivery service, is that Flakey should bring the report to Harry. However, there is no way of generating INT(/i , *bring(F,* ft, //, *S'))* from ACH(fl, *IIHS(H , 11, S))* --- the *lo2* argument works in the other direction. As we have argued, there could be many ways of achieving a proposition, and in order to infer which of these is the correct one (that is, which one Harry has in mind) it is necessary to use contextual information. In any given domain, there are certain "normal" or "typical" ways to do things, which are more likely than their alternatives. In the ftD/) a request by an agent to Flakey is usually an attempt to get Flakey to deliver something. In this case, we would have the following context-dependent rule:

$$\begin{array}{l}\mathrm{BEL}(a, TO(bring(F, o, b, s), has(b, o, s'))), \\ \mathrm{ACH}(a, has(b, o, s')) \xrightarrow{to3} \\ \mathrm{INT}(a, TO(bring(F, o, b, s), has(b, o, s'))), \\ \mathrm{INT}(a, bring(F, o, b, s))\end{array} \quad (12)$$

We do not have a theory of how Flakey would arrive at a set. of such rules; but supposedly it would involve knowledge of the frequency with which certain types of requests are made, the utility and cost of various alternative actions, and so on.

6 Adjudicating Local Ascriptions

One common result, of having many local ascription rules is that their conclusions will often conflict. Often, it is possible to adjudicate these conflicts on the basis of local information, thus sharply limiting ambiguity in ascription. In this section we examine two types of local adjudication. First, however, we define conditions of conflict in the *RDD.*

6.1 Conflicting Propositions

For the examples in this paper we will need only two kinds of conflict. If an agent expects to achieve *p* by performing some action, he generally will not believe that *p* will become true if he does not act. Similarly, if he believes *p* will be true regardless of his actions, he will generally not plan to achieve *p.* So we have:

$$\mathrm{BEL}(a, p) \ and \ \mathrm{ACH}(a, p) \ conflict. \quad (13)$$

A second kind of conflict arises when two intended actions would occur at the same time and have the same effect. Thus Flakey bringing Harry the report, and Harry getting it himself (in the same time interval) are conflicting intended actions.

G.2 Initial Fact Defeat

Our first defeat rule acknowledges the importance of initial facts:

Initial Fact Defeat An initial fact of a world defeats any argument supporting a conflicting proposition.

FXAMPLK 0.1 Consider a slight, variation of Example 5.1. Harry makes the same request as before, but this time Flakey already has the report. Consequently, to the initial world given above we add the statement:

$$\mathrm{T}(has(F, R, S_1)). \quad (14)$$

By belief ascription, there should be support for $\mathrm{BEL}(H, has(F, R, S_1))$. However, this conflicts with $\mathrm{ACH}(H, has(F, R, S_1))$. Because the latter is an initial fact—it was derived directly from Harry's request—it defeats the belief ascription argument. Harry's beliefs are hence judged to be different from Flakey's.

Fxample 0.1 provides a very simple example of an actor with an incorrect plan. Here, the problem with the plan is that it relies on an incorrect situational belief. In other cases, actors' plans may rely upon incorrect beliefs about the relations between actions (represented with BY"or *TO* statements). To handle these cases, we can introduce additional plan-fragment ascription rules, modeled on those developed by Pollack [Pollack, 1986], along with defeat principles that adjudicate bet,ween competing plan-fragment ascription rules. For example, arguments that a,scrihe belief in incorrect relations are generallv defeated by those that ascribe correct belief.

6.3 Purposeful Action Defeat

EXAMPLE 0.2 This time all that Harry requests is that Flakey get the report,: he does not assert why he wants him to do that. And Flakey already has the report. So the initial world is:

$$\begin{array}{l}\mathrm{INT}(H, get(F, R, S_0)) \\ \mathrm{T}(TO(get(F, R, S_0), has(F, R, S_1))) \\ \mathrm{T}(has(F, R, S_1)).\end{array} \quad (15)$$

As in Example 0.1, belief ascription leads to support for BEL(ft,/ms(F, ft, *S]*)). But it is also the case that the *to2* rule results in support for ACH(HJas(F, ft,S_1j)), and as before these two facts conflict. The natural conclusion is that Harry wants to achieve Flakey's having the report: he does not realize that Flakey already has it. The argument using the *to2* schema defeats the belief ascription argument.

We thus propose the following defeat principle:

Purposeful Action Defeat If $x \xrightarrow{to2} \mathrm{ACH}(a, p)$ is an argument whose premises *x* are supported, and so is $y \xrightarrow{belasc} \mathrm{BEL}(a, p)$, then the belief ascription argument is defeated.

The name of this defeat principle reflects the fact that it encodes a presumption that agents engage in purposeful actions: they do not typically intend actions whose effects they already believe to be true.

In the *HDD* the Purposeful Action Defeat, principle results in natural conclusions. However, in more complex domains, this rule would have to be complicated. To see why, imagine a case in which an intended action α could he used to achieve more than one proposition, if the observer believes that one of these, say p, is already true, while another, say q, is not, he may be justified in inferring that the actor intends to do α in order to achieve q. He does not then need to ascribe to the actor a mistaken belief that p is not true. Balancing such considerations requires the use of experiential, or context-dependent knowledge: such reasoning should be used only if q is a likelv intended result of α. We now turn our attention to another way in which context-dependent, knowledge can affect plan recognition.

7 Evaluative Methods

As we indicated earlier, local ascription is only part of the process of plan recognition. Local information that can plausibly be ascribed to an actor must be evaluated to determine whether it forms a globally coherent structure.

7.1 Conflicting Action Defeat

In addition to the kinds of local defeat rules presented above, one can distinguish among alternative plan-fragment ascriptions by using the context of the other ascribed fragment,s. That is, if there are competing alternative actions, and one of them is part of a coherent set of ascribed plan fragments while the other is not, then we prefer the former. To illustrate this principle, recall Example 11, in which Flakey needs to derive INT(r7,/>n"jig(F,R,//,S')) from ACH(H, /ias(H, K, *S)).* Now consider what happens if he makes use of the following reasonable argument, schema:

$$\mathrm{BEL}(a, TO(\alpha, has(b, o, s))), \mathrm{ACH}(a, has(b, o, s')) \xrightarrow{to4} \mathrm{INT}(a, TO(\alpha, has(b, o, s'))), \mathrm{INT}(a, \alpha) \ . \quad (16)$$

While still context-dependent, this rule is more general than *to3,* since it supports ascribing an intent-ion to do any action which leads to an agent having an object.

Flakey can use *to4* to infer support for both INT(//,hWng(F,R,R,S_i)) and TNT(H,get(W, R,S_i)); these intentions conflict, because they occur at the same time and lead to llarry having the report. (Recall, from Section 6.1, that in the *HDD,* two intended actions conflict if they would occur at the same time and have the same effect.) This conflict means that Flakey will not be able to accept either intention, although both are supported.

How can we distinguish the correct inference, that Flakey should bring Harry the report? As we noted above, this intention differs from Harry's intention to get the card himself in that it is connected by plan fragments to another initial fact: Harry's intention that Flakey get the report. Let us define the *support set* of a proposition as those initial fact INT and ACH predicates which are used as premises in some argument chain that supports the proposition. Then we can state the preference lor the more-connected intention as the following defeat principle.

Conflicting Action Defeat If α and β are conflicting actions, and INT(a, α) and T N(a, β) a v e arguments with supported premises, and the support set of the former is a proper superset of the support set of the latter, then any argument to INT(a, β) is defeated.

7.2 Limits of Global Evaluation

Global evaluation may it,self be a complicated process, involving many different coherency rules; Conflicting Action Defeat is only one example. Moreover, global evaluation may fail: the observer may not be able to view the locally ascribed beliefs and intentions as a globally coherent plan. Consider one more example:

EXAMPLE 7.1 Harry requests that Flakey get the report so that he will win a bet he made with Sue. (It will suflice for our purposes to represent this last proposition as *winbct(ll, S_i).)* The initial world includes 1NT(H, *gct(* F,R.,S_o)) and ACll (H , *winet(ll, S_i)*). Flakey Inay believe that Harry intends for there to be some connection between these two facts: he could come to this belief through the use of arguments derived from the Cooperative Principle of Conversation [Grice, 1975], which we do not present in this paper. However, Flakey is unable to determine precisely what connection Harry intends: in a very real sense, Flakey is unable to recognize Harry's plan. He may need to ask him what he has in mind.

In this example, Flakey is not able to find a single coherent plan that explains Harry's request.[4] He cannot relate the asserted intended action and the asserted goal; in particular, he cannot infer INT(H, *TO(F,* R,So), *winbet(H,.S_i)*)). When the locally ascribed beliefs and intentions do not form a globally coherent, structure, further interaction between observer and actor may be necessary.

8 Conclusion

We have presented a direct argumentation model of plan recognition which we are currently developing. We believe that plan recognition is especially well suited to this approach, for several reasons. First, knowing enough to engage in plan recognition means not only knowing what kinds of arguments there are for belief and intention ascription, but also knowing about the relative strength of these arguments. The defeat, principles described in this

[4] In our account, such a plan would consist of a set of intended *BY* and *TO* relations that connected all the asserted intentions and goals, and that also connected those to a likely domain goal. Deciding what constitutes a likely domain goal—i.e., deciding when to terminate the plan recognition process—is, in general, a difficult problem. Existing systems have either assumed that there is a very small set of goals [Allen, 1983, Kautz, In Press, Lit man and Allen, 1987] or else that the actor makes his goal explicit [Pollack, 1986j.

paper are examples of this. Direct argumentation systems provide a natural way of representing such knowledge. Second, in direct argumentation systems, the arguments used to arrive at any conclusion are readily available for inspection—and cooperative interaction requires that agents reason about the correctness of the plans they ascribe to others. Finally, direct argumentation systems are incremental, in the sense that one can add a single argument at a time. This means that when resource limits are encountered, the argumentation process can stop, and return the best conclusion so far derived. This contrasts with indirect systems for defeasible reasoning, such as circumscription, which rely on some kind of global minimization [Konolige, 1988]. Although we have not focused on this aspect of plan recognition in this paper, the ability to cope with resource limitations during reasoning is crucial for agents situated in dynamic, multiagent environments—the very agents most likely to make plans and reason about one anothers' plans [Bratman *et ai,* 1988].

Acknowledgements

We would like to thank Ray Perrault for his comments on this work.

References

[Allen, 1983] J. F. Allen. Recognizing intentions from natural language utterances. In M. Brady and R C. Berwick, eds., *Computational Models of Discourse.* MIT Press, Cambridge, Ma., 1983.

[Bratman *et al.,* 1988] M. E. Bratman, D. J. Israel, and M. E. Pollack. Plans and resource-bounded practical reasoning. *Computational Intelligence,* 4(4), 1988.

[Castaneda, 1975] H.-N. Castaneda. *Thinking and Doing.* Reidel, Dordrecht, Holland, 1975.

[Doyle, 1979] J. Doyle. A truth maintenance system. *Artificial Intelligence,* 12(3), 1979.

[Grice, 1975] H.P. Grice. Logic and conversation. In P. Cole and J. Morgan, eds., *Syntax and Semantics Vol. 3: Speech Acts.* Academic Press, New York, 1975.

[Horty *et al,* 1988] J.F. Horty, R.H. Toh mason, and D.S. Touretzky. A sckeptical theory of inheritance in non-monotonic semantic nets. In *Proceedings of the American Association for Artificial Intelligence,* Seattle, Wa., 1988.

[Kautz, In Press] H. A. Kautz. A circumscriptive theory of plan recognition. In P. R. Cohen, J. Morgan, and M. E. Pollack, editors, *Intentions in Communication.* MIT Press, Cambridge, Ma., In Press.

[Konolige, 1988] K. Konolige. Hierarchic autoepistemic theories for nonmonotonic reasoning. In *Proceedings of the American Association for Artificial Intelligence,* Seattle, Wa., 1988.

[Konolige, 1989] K. Konolige. Defeasible argumentation in reasoning about events. In *Proceedings of the International Symposium on Machine Intelligence and Systems,* Torino, Italy, 1988.

[hitman and Allen, 1987] D. .1. Litman and .1. F. Allen. A plan recognition model for subdialogues in conversations. *Cognitive Sen Tier,* 11(2), 1987.

[Loui, 1987] R. P. Loui. Defeat among arguments: A system of defeasible inference. *Computational Intelligence,* 3(2), 1987.

[McCarthy, 1984] .1. McCarthy. Applications of circumscription to formalizing common sense knowledge. In *Proceedings of the American Association for Artificial Intelligence Workshop on Nonmonotonic Reasoning,* New Paltz, NY., 1984.

[McDermott, 1982] D. McDennott. A temporal logic for reasoning about processes and plans. *Cognitive Science,* 6(2), 1982.

[Pollack, 1986] M. E. Pollack. A model of plan inference that distinguishes between the beliefs of actors and observers. In *Proceedings of the Association for Computational Linguistics,* New York, 1986.

[Pollack, In press] M. E. Pollack. Plans as complex mental attitudes. In P. R. Cohen, .1. Morgan, and M. E. Pollack, editors, *Intentions in Communication.* MIT Press, Cambridge, Ma., In press.

[Poole, 1985] D. Poole. On the comparison of theories: preferring the most specific explanation. In *Proceedings of the Inter-national Joint Conference on Artificial Intelligence,* Los Angeles, 1985.

[Reiter, 1980] R. Reiter. A logic for default reasoning. *Artificial Intelligence,* 13(2), 1980.

[Schmidt *et al,* 1978] C.F. Schmidt, N.S. Sridharan, and J.L. Coodson. The plan recognition problem: an intersection of artificial intelligence and psychology. *Artificial Inteliigence,* 10(1), 1978.

[Sidner, 1985] C. L. Sidner. Plan parsing for intended response recognition in discourse. *Computational Intelligence,* 1(1), 1985.