Logic-Based Approaches to Intention Recognition

Fariba Sadri

Department of Computing, Imperial College London, UK

fs@doc.ic.ac.uk

Abstract

In this paper we discuss intention recognition in general, and the use of logic-based formalisms, and deduction and abduction in particular. We consider the relationship between causal theories used for planning and the knowledge representation and reasoning used for intention recognition. We look at the challenges and the issues, and we explore eight case studies.

1 Introduction

Intention recognition, also called ***goal recognition***, is the task of recognizing the intentions of an agent by analyzing some or all of their actions and/or analyzing the changes in the state (environment) resulting from their actions. ***Plan recognition*** is closely related to intention recognition, and extends it to recognizing the plan (i.e. the sequence of actions, including future actions) the observed agent is following in order to achieve his intention. Throughout this paper we will use *goal* and *intention* interchangeably.

Work on **intention recognition** has been going on for about 30 years. Examples of early work are attributed to Schmidt et al. [1978], Wilensky [1983], and Kautz and Allen [1986]. Much of the early work has been in the context of language and story understanding and automatic response generation, for example in Unix help facilities. However, new applications, such as assisted living and ambient intelligence, increasingly sophisticated computer games, intrusion and terrorism detection, and the military have brought new and exciting challenges to the field. For example assisted living applications require recognizing the intentions of residents in domestic environments in order to anticipate and assist with their needs. Applications in computer systems intrusion or terrorism detection require recognizing the intentions of the would-be-attackers in order to prevent them. Military applications need to recognize the intentions of the enemy maneuvers in order to plan counter-measures and react appropriately. Programs that make moral decisions (e.g. Pereira and Saptawijaya 2009) need to reason about intentions, in particular to decide whether untoward consequences of some actions were intended by the agent that performed them or were merely unintended side-effects.

Logic has been a powerful tool in **intention recognition**. Since the early days, for example in the work of Charniak and McDermott [1985], **abduction** has been used as the underlying reasoning mechanism in providing hypotheses about intentions. Also, conceptually, **intention recognition** is directly related to planning, and logic has been the basis of many **causal theories** describing the relationship between actions and effects.

Intention recognition is a rich and challenging field. Often multiple competing hypotheses are possible regarding the intentions of an observed agent. The choice between these hypotheses is one challenge, but there are many others. One, for example, is that circumstances, including the adversarial nature of the observed agent, may afford only partial observability of the actions. Geib and Goldman [2001] make a contribution in this respect, as do Sindlar et al [2008], described in Section 4 in this article. Furthermore, would-be-intruders and would-be attackers may even deliberatively execute misleading actions.

Another challenge is the case where the acting agent may have multiple intentions and may interleave the execution of his plans of actions for achieving them, or the case where the actor is concurrently trying alternative plans for achieving the same intention. Intention recognition becomes more difficult when we attempt to interpret the actions of cognitively impaired individuals who may be executing actions in error and confusion, for example in the case of Alzheimer patients (Roy et. al. 2007). Similar complications arise and are magnified when attempting to analyze the actions and intentions of multiple (co-operating) agents (e.g. Sukthankar and Sycara 2008).

In this article we will explore the field of intention recognition, and in particular we will focus on single agent cases, as opposed to multi-agents, and on logic-based approaches. We will explore the logical basis of intention recognition, and provide an analysis of eight case studies. The case studies are chosen from the literature for their variety of methodologies and applications. We will not consider many probabilistic approaches to intention recognition, except in two of the case studies, Pereira and Anh [2009b] and Demolombe and Frenandez [2006], both of which combine logic with probabilities.

In Section 2 we look at the background and issues involved in intention recognition. In Section 3 we look at the possible relationships between logic-based causal theories and knowledge representation and reasoning for intention recognition. In Section 4 we describe and analyze the eight case studies. Finally, in Section 5, we conclude with a further discussion of the challenges.

2 Background and Issues

2.1 Applications

Much of the early work on intention recognition has been in the context of language and story understanding and automatic response generation, for example in Unix help facilities. Other early application areas include interfaces for computer-aided design [for example Goodman and Litman 1992] and collaborative problem-solving [Lesh, Rich, Sidner 1999]. More recent applications have been in diverse areas. We mention some of these below.

Assisted technologies, in general, and in the care of the elderly at home, in particular, are popular application areas for intention recognition. Giroux et al. [2008] address intention recognition for the purpose of providing assistance for cognitively impaired individuals. Pereira and Anh [2009b] and Geib and Goldman [2005], for example, focus on intention recognition in the care of the elderly. Roy et al. [2007] address complications in intention recognition when tracking the behaviour of Alzheimer patients. In the case of such patients it cannot always be assumed that their actions are based on an organized rational plan. So if the next action they execute is not what is expected it can be due to an interleaving of two plans for two different goals or it can be due to error.

Canberry and Elzer [2007] consider a more unusual, but related, application, namely the recognition of intention behind (bar chart) graphics, in order to convey the “messages” of bar charts to sight-impaired individuals. The application involves recognizing the intention of the designer of a bar chart, by analyzing an XML representation of the chart that provides information about the heights, colours and labels of the bars.

Another application area is interactive storytelling. LOGTELL [Karlsson et al. 2007], for example, is a logic-based tool for interactive storytelling. During story creation, the user can intervene by inserting events chosen from a pre-specified list. Plan recognition is then used to find plans from a plan library that subsume these events. The user chooses from amongst them and the story unfolds accordingly.

Other major application areas include computer system intrusion detection [Geib and Goldman 2001], terrorism intrusion detection [Jarvis et al. 2004], and computer games, such as real-time strategy games [Cheng and Thawonmas 2004]. The military and the civil policing also provide major applications and challenges to the field. Mao and Gratch [2004], for example, address intention recognition from observing military movements. Suzić and Svenson [2006] consider the application of riot control in urban environments. Observations include movements of groups of people. From these and contextual information, such as location (how close they are to which buildings) and resources (what weapons they have) their intention is hypothesized.

2.2 Classification

Cohen, Perrault, Allen [1981] classify intention recognition as either *intended* or ***keyhole***. In the *intended* case the agent which is being observed wants his intentions to be identified and intentionally gives signals to be sensed by the other (observing) agent. This would apply, for example, in the case of language understanding where the speaker wants to convey his intentions. In the ***keyhole*** case the agent which is being observed either does not intend for his intentions to be identified, or does not care; he is focused on his own activities, which may provide only partial observability to the other agent. This might be the case with help systems that provide unsolicited guidance, for example in an ambient intelligence system at home.

A third class, identified by Geib et al [2001], is *adversarial*, where the actor is hostile to his actions being observed, for example where the actions are aimed at intrusion in a network system. In fact we can take the classification further, to *diversionary*, where the actor is in fact attempting to conceal his intentions by performing misleading actions. Much of the work on intention recognition concentrates on the first two classes, namely *intended* and *keyhole*. Pereira and Anh [2009a] is a recent attempt to deal with diversionary intentions.

2.3 Components, Formalisms, Methodologies

Typically there are at least three components in an intention recognition system: (1) a set of intentions from which the system chooses, (2) some form of knowledge about how actions and plans achieve goals, and (3) a sequence of observed actions executed by the agent whose intention is being recognized. A possible additional component may be a set of hypotheses currently held about the agent’s intention. For example, in a home setting, if it is late at night the recognition system may hypothesize that the resident normally prepares a hot drink or prepares a hot water bottle for bed.

There is often an assumption that the actions the observed agent executes are aimed at achieving a goal, i.e. are part of the execution of a plan formed by the agent. Of course, in difficult cases, for example in the case of Alzheimer patients, or students learning a new skill, the actions may be erroneous. There is very little work, if any, that is directed at recognizing *reactive* behaviour, namely recognizing that actions may be done in reaction to external events and stimuli, and not part of a *proactive* plan.

The intention recognition problem has been cast in different formalisms and methodologies. Prominent amongst these are **logic**-based formalisms, case-based approaches, and probabilistic approaches. Accordingly, component (2) of the system, i.e. the knowledge about the relationship between actions and goals, may be, for example, in the form of **logic**-based specifications of macro-actions [Demolombe and Fernandez 2006], or in the form of cases [for example Cox and Kerkez 2006], or in the form of plan libraries specified as Hierarchical Task Networks (HTNs) [Geib and Goldman 2005]. Geib and Steedman [2003] cast intention recognition as a problem in parsing, much as in natural language processing. Accordingly, they map Hierarchical Task Networks into context-free grammars, and the parsing is used to group together individual observations into structures that are meaningful according to the grammars.

A common assumption is that the observer agent has (full) knowledge of the planning rules (sometimes called behaviour rules) of the acting agent. For intention recognition, the observer agent may use the same representation of the planning rules, or more commonly, some transformed form of those rules which are not useful for planning, but are useful specifically for intention/plan recognition. This transformation may be, for example, in the form of HTNs augmented with probabilities and/or utilities, or the only-if direction of if-then planning rules. This will be discussed further in Section 3.

Depending on how the knowledge is cast in component (2), component (1), i.e. the set of all possible intentions at the disposal of the system, may be explicitly represented or may remain implicit. If it is explicit it will be represented as a list of possible intentions, or a set of logical facts naming each possible intention. If it remains implicit then it can be identified as, for example, the topmost operator in the HTN used in component (2), or as (ground instances of) of the predicates in the conclusion of certain sets of rules used in (2).

Component (3) of an intention recognition system is the sequence of observed actions. The assumption that the acting agent's actions can be observed unambiguously is a strong one which may be justified only in virtual environments such as simulations or games. To provide the intention recognition system with this component, *activity recognition* may be used in conjunction with intention recognition.

2.4 Activity recognition

Activity recognition, typically, uses data from cameras and RFID (Radio Frequency Identification) readers and tags to track movements of humans and to identify the objects they handle. In the home environment, for example, RFID tags may be attached to household objects. Park and Kautz [2008a,b], for example, describe a prototype human activity recognition system built in the Laboratory for Assisted Cognitive Environments (LACE). The system uses a combination of distributed multi-view visions system and RFID readers and tags to provide information about activities, at both coarse (e.g. walking around) and fine (e.g. opened cupboard door, took out cereal box) levels.

Mihailidis, Fernie, Barbenel [2001] focus on recognizing very specific activities such as hand washing, similarly to Barger et al. [2002] who focus on meal preparation. PROACT [Philipose et al. 2005] employs a Dynamic Bayesian Network representing daily activities such as making tea. The user wears special gloves that can read the RFID tags of objects such as cups. Making tea is modeled as a three stage process, with high probabilities of using the kettle in the first stage and the box of tea-bags in the second stage, and medium probability of using milk, sugar or lemon in the third stage. PROACT then uses information about the objects being used and time elapsed between their usages to hypothesise possible activities. Sanchez, Tentori, Favela [2008] describe work on activity recognition in a hospital. Here information about the context, such as location, time, and the roles of the people present, and RFID-tagged artifacts is used to determine what an actor is doing, for example the activities of a ward nurse and doctor during the morning hours, handling reports and case files, may be interpreted as patient care or clinical case assessment.

Whatever the formalism and methodology, the result or output of intention recognition is a hypothesis or a set of hypotheses about the intention of the observed agent. The output of plan recognition, in addition includes hypotheses about the plan of the observed agent, including the rest of his actions yet to come. The process of generating such results may be iterative whereby hypotheses are generated, predictions are made and tested, and hypotheses are refined.

2.5 Pruning the space of hypotheses

A substantial issue in intention (and plan) recognition is the problem of narrowing down the space of possible hypotheses. Various techniques have been used to accomplish this. Early approaches impose minimality or simplicity constraints, for example via circumscription [Kautz and Allen 1986]. Here circumscription is used, in effect, to characterize the assumption that all the actions observed are being executed towards a minimum number of (top-level) intentions.

Appelt and Pollock [1992] use weighted **abduction** where weights are attached to conditions of the rules used for intention recognition. The cost of proving a conclusion G is the sum of the costs of proving the conditions of a rule whose conclusion matches G. The cost of proving each condition depends on whether it is assumed via abduction, or is true or is proved using other rules. The weights are intended to capture domain-specific information. For example, consider the two rules below:

$$building(X, public) \wedge door\text{-}open(X)^{0.1} \rightarrow may\text{-}enter(X)$$

$$building(X, private) \wedge door\text{-}open(X)^{0.9} \rightarrow may\text{-}enter(X)$$

They both state that you may enter a building if its door is open, but the cost of assuming that the door of the building is open is much higher if it is a private building than if it is a public one. Put another way, intuitively, more evidence is required to believe that the door of a private building is open than to believe that the door of a public building is open.

More recently, Jarvis, Lunt and Myers [2005] couple a form of abductive reasoning with domain information regarding frequency of certain actions (for example *high* for renting a car and *low* for filing a crime report). Then in the application of terrorism intention recognition they use this information to impose a maximum on the frequency of observed actions that are to be used in the recognition system. Thus, given a set of observed actions the system focuses on a subset of these that have a maximum threshold frequency, in effect ignoring the more “common” actions and focusing on the more “rare” or “unusual” ones. Another approach is to make use of ordering constraints. For example, Avrahami-Zilberbrand and Kaminka [2005], reject hypotheses of plans which have some matching observed actions but also have actions that should have been executed earlier but have not been observed.

Associating weights with conditions of rules as in Appelt and Pollock [1992] and *a priori* frequencies to actions as in Jarvis, Lunt and Myers [2005] may be thought of as forms of probabilistic reasoning. Other more explicit forms of probabilistic reasoning to prune the search space have also been used [e.g. Geib and Goldman 2001, 2005, Geib and Steedman, 2003]. The probabilistic approaches may be based on Baysian reasoning or the hidden Markov model [Bui 2003]. Another approach is situation-sensitive Causal Bayes Nets [Pereira and Anh 2009b] where logic programming clauses are used to specify probabilities of intentions given information about the current state, including time of day and temperature. For example, modifying the authors’ notation, the following rules

$$pa\_rule(lw(T), (9,10)) \leftarrow time(T), schedule(T, football)$$

$$pa\_rule(lw(T), (1,10)) \leftarrow time(T), (T>23 \vee T<5)$$

$$pa\_rule(lw(T), (3,10)) \leftarrow temp(T, TM), TM>30$$

state that the probability of (the observed agent) liking to watch TV (*lw*) is 90% when football is on, 10% when it is between 23:00 and 5:00 hours, and 30% if the temperature is higher than 30 degrees.

Mao and Gratch [2004] combine probabilities with utilities in plan recognition to choose from amongst competing hypotheses. The domain they consider is that of military maneuvers. Intentions are given pre-specified associated utilities as well as probabilities. The utilities are from the point of view of the observed agent. Thus if the observed actions lead to two equally probable hypotheses they are ranked according to their utilities, preferring the hypothesis which has higher utility, i.e. the one believed to be more profitable for the actor. Avrahami-Zilberbrand and Kamonka [2007], on the other hand, exploit utility from the point of view of the observer, for example ordering the hypothesized intentions according to how dangerous they may be to the observing agent. In their work this is particularly useful when there is uncertainty about the observations. For example, in CCTV monitoring at an airport, someone is observed putting down their luggage and it is uncertain if they have picked it up and taken it with them or not. The hypothesis that they have left the luggage with criminal intent is preferred as it is more dangerous from the point of view of the observers.

Once a hypothesis is chosen from amongst the possible ones, how it is used depends on the application of intention recognition. For example, two contrasting applications are identifying terrorist activity and providing assistance at home. In the first [e.g. Jarvis et al. 2004], the objective is to prevent the terrorists achieving their intentions by first identifying the intentions. In the second, for example the care of the elderly at home in an ambient intelligence setting, the objective is to help and guide the elder towards achieving his intention, by first identifying the intention. Notice that these two applications correspond to the adversarial (and possibly diversionary) and **keyhole** (and possibly intended) classes of problems, respectively.

To our knowledge, there is no work studying the relative effectiveness of the different approaches to intention recognition. Mayfield [2000] proposes three criteria for evaluating the effectiveness of the outcome of plan recognition systems, *applicability*, *grounding* and *completeness*. *Applicability* refers to how useful the explanation generated by the plan recognition system is to the program (or person) which is to use it in terms of its content, granularity and level of detail. *Grounding* refers to how well the plan recognition system takes account of all that is known about the actor and context (apart from actions that are observed). *Completeness* refers to how well the explanation that is produced covers all of the observations. But the three notions remain informal and anecdotal in Mayfield's work and his focus is on dialogue understanding particularly in the context of Unix help facility.

3 Logic-Based Approaches to Intention Recognition

3.1 Abductive Approaches

Abduction is a prominent methodology used in intention recognition and forms the basis of several of the papers reviewed in the case studies in Section 4. **Abduction** [Peirce 1958] is a form of defeasible reasoning, often used to provide explanations for observations. For example given a logic rule

$$\textit{room-is-hot} \leftarrow \textit{heating-is-on}$$

deduction allows deriving *room-is-hot* from the knowledge that *heating-is-on*, and abduction allows abducing *heating-is-on* to explain the observation of *room-is-hot*. In the abductive framework [e.g. Kakas, et al. for abductive logic programming], in general, given a background theory T, and an observation (or goal) Q, an abductive answer to Q is a set Δ, such that Δ consists of special pre-specified abducible atoms, $T \cup \Delta \models Q$ and $T \cup \Delta$ is consistent. In addition, in particular in abductive logic programming, an extra requirement may be that $T \cup \Delta$ satisfies a given set of integrity constraints. For example an integrity constraint

$$\textit{heating-is-on} \wedge \neg \textit{boiler-working} \Rightarrow \textit{false}$$

will result in disregarding *heating-is-on* as an explanation for *room-is-hot* if it is believed that the boiler is not working.

Charniak and McDermott [1985] were possibly the first to suggest that intention/plan recognition could be framed as an abductive problem. Their focus was on intention recognition, or *motivation analysis*, as they called it, in the context of story comprehension. They identified intention recognition as the reverse of planning. In the latter, given a task, reasoning is employed to determine what actions would achieve it. In the former, given an action, reasoning is employed to determine what tasks it could help achieve, either directly, or in conjunction with other possible actions. This reversal of the reasoning employed for planning gives the flavour of abductive reasoning, but their actual formalization did not strictly conform to the abductive framework, as described above. For the purpose of intention recognition plan schemas were compiled in the form *todo(G, A)* denoting that action A achieves goal G. Thus observing an instance of A, the same instance of G could be hypothesized as a possible intention.

Abduction can provide multiple hypotheses explaining an observation. Charniak and McDermott [1985] suggested a number of criteria for choosing between multiple hypotheses. One is to prefer a hypothesis that uses the most specific characteristics of the observed action. For example, if we observe that Tom picks up a newspaper, there may be two possible explanations, he intends to read it or he intends to swat a fly with it. According to the specific characteristics criteria, the first is preferred because it uses the characteristic of the newspaper as a readable object, whereas the second uses the characteristic of the newspaper just as an object. Another criterion suggested is to prefer a hypothesis which requires fewer additional assumptions (similar to the *global* criteria mentioned below). For example the explanation of swatting a fly requires an additional assumption that there is a fly, and may thus be less preferred to the explanation of reading if that requires no additional assumptions.

More recently, two broad types of criteria, *global* and *local*, are often used for ranking, and thus choosing from amongst the explanations. The global criteria may, for example, prefer explanations that are minimal in some sense, for example syntactically in terms of the number of facts. The local criteria, on the other hand, may associate some form of evaluation metric with each rule in the background theory, and provide an evaluation metric for a set of hypotheses by combining the metrics of the rules from which the hypotheses originated.

In intention recognition the background theory, T, is a characterization of the relationships between actions and intentions, the observations, Q, are the actions of the observed agent, and the explanations, Δ, are hypotheses about the agent's intentions. In general, as explained in Section 2, whatever form of reasoning is employed, whether abductive, deductive, probabilistic or a mixture, intention recognition requires some knowledge of how actions achieve goals. As observed by Charniak and McDermott, such a theory is conceptually closely related to **causal theories** used for planning.

3.2 Causal Theories for Planning and Theories for Intention Recognition

A common premise of intention recognition is that the observed agent is rational (even though forgetful and chaotic in some cases), and is pursuing a course of actions he believes will help him achieve a goal. This course of action must be the result of reasoning with some causal theory for planning. A further assumption is that the observer agent has some knowledge of this causal theory, although he may not use that same theory for intention recognition. The theory that he uses for intention recognition may have some direct logical relationship to the observed agent's causal theory, or may have some loose and informal relationship to it. The theory and the representation that the observer agent uses lends itself naturally to the form of reasoning that the agent needs to employ.

Logic-based **causal theories** essentially include rules of the form:

> *Goal* ← *Conjunction of actions and preconditions of actions and other properties and temporal constraints.*

Causal theories are often general purpose and can be used for different applications, such as planning and prediction. Here, we are only interested in their use for planning. In particular there are two approaches, planning from first principles and planning from second principles or plan libraries. Both can be formalized, for example in the **situation calculus** [Reiter 2001] or the **event calculus** [Kowalski and Sergot 1986], or some other formalism of actions and effects. So for example an abductive theory of the **event calculus** for first principle planning may include the following rules:

$$holds\text{-}at(P,T) \leftarrow happens(E, T1) \wedge initiates(E,P) \wedge T1<T \wedge persists(T1,P,T)$$

$persists(T1,P,T) \leftarrow not\ clipped(T1,P,T)$

$clipped(T1,P,T) \leftarrow happens(E, T2) \wedge terminates(E,P) \wedge not\ out(T2,T1,T)$

$out(T2,T1,T) \leftarrow T=T2$ $\quad$ $out(T2,T1,T) \leftarrow T<T2$ $\quad$ $out(T2,T1,T) \leftarrow T2<T1$

Integrity constraint: $\quad$ $happens(A, T), precondition(A,P) \Rightarrow holds(P, T).$

The rules state that a property *P* holds at a time *T* if an event *E* happens earlier which initiates *P* and *P* persists (at least) from the occurrence of *E* until *T*. *P* persists between two times if it is not clipped in that interval. *P* is clipped in an interval if an event *E* happens that terminates *P* and it cannot be shown that *E* occurred outside that interval. Here *not* can be thought of negation as failure. The integrity constraint states that an action can be done only if its preconditions hold.

Domain dependent information is used to specify rules defining *initiates* and *terminates*, and *preconditions* of actions, for example:

initiates(unlock-door(R), gain-entry(R))

terminates(lock-door(R), gain-entry(R))

precondition(unlock-door(R), have-key(R))

precondition(unlock-door(R), in-front-of(R))

These state that unlocking the door to a room initiates gaining entry to that room, and has preconditions having the key to and being in front of that room, and locking the door terminates gaining entry. In this abductive framework the set of abducible predicates will consist of *happens* and the ordering relations, = and <.

A theory for planning from second principles, based on the above theory, and further information about preconditions and effects of actions, may have rules such as:

$$gain\text{-}entry(laboratory, T+3) \leftarrow goto(reception, T) \wedge pick\text{-}up\text{-}key(laboratory, T+1) \wedge goto(laboratory, T+2) \wedge unlock\text{-}door(laboratory, T+3).$$ [1]

Thus a **logic**-based plan library is typically of the form:

$$G \leftarrow A_{1;} \ldots; A_n$$

where ";" denotes (conjunction and) sequencing.

Very little work uses causal theories from first principles for intention recognition (one exception is Quaresma and Lopes [1995] reviewed in Section 4), although an obvious major advantage of it would be to increase the chances of recognizing the intention behind unusual and unpredicted clusters of actions,

[1] Here we are ignoring persistence and we are assuming that no action occurs between any two times T and T+1.

and also lending itself well to recognizing short and medium term intentions as well as the ultimate intention.

Much of intention recognition work assumes a plan library, at least used by the observed agent. If the observer agent uses the same representation, i.e.

$$G \leftarrow A_{1;}\ \ldots;\ A_n$$

then the reasoning employed for intention recognition can be deductive, reasoning from the observed actions to the goal they establish, i.e. given instances of $A_{1;}\ \ldots;\ A_n$ deduce the appropriate instance of G. To make the approach more flexible this can be combined with probabilities, in the sense of increasing the probability of (an instance of) G being the intention as an increasing number of the (appropriate instances of the) actions A_i are observed. This is essentially the basis of the work of Demolombe and Fernandez [2006], described in Section 4. The reasoning may also be a combination of deductive and abductive, reasoning deductively from the occurrence of some actions $c_{i;}\ c_{j;}\ldots;c_k$ matching actions $A_{i;}\ A_{j;}\ldots;\ A_k$, with a most general unifier σ, to deduce a residue (resolvent)

$$G\sigma \leftarrow (A_{1;}\ \ldots;\ A_{i-1;}\ A_{i+1};\ \ldots.;\ A_{j-1};\ A_{j+1};\ \ldots.;\ A_{k-1};\ A_{k+1};\ldots.;\ A_n)\sigma$$

and then abducing (possibly a ground instance $\sigma\phi$ of) the remaining actions $(A_{1;}\ \ldots,\ A_{i-1,}\ A_{i+1},\ \ldots.,\ A_{j-1},\ A_{j+1},\ \ldots.,\ A_{k-1},\ A_{k+1},\ldots.,\ A_n)\sigma$ (and any conditions, such as the ordering), and thus hypothesizing the goal $G\sigma$ (or $G\sigma\phi$). This is essentially the approach used by Quaresma and Lopes [1995], also described in Section 4.

Another approach is for the observer agent to employ a theory of rules of the form

$$G \rightarrow A_{1;}\ \ldots;\ A_n$$

or more generally,

$$G \rightarrow (A_{11;}\ \ldots;\ A_{1n}) \vee (A_{21;}\ \ldots;\ A_{2m}) \vee \ldots. \vee (A_{p1;}\ \ldots;\ A_{pq})$$

which is the only-if half of a completed plan library. Then the reasoning employed is abductive, finding hypotheses G that would explain observations of actions A_{ij}. This is essentially the approach used by several of the studies described in Section 4, in particular Sindlar, Dastani et al. [2008], Myers [1997], Jarvis, Lunt, and Myers [2005], and Dragoni, Giorgini and Serafini [2002]. However, Sindlar, Dastani et al. [2008] use a meta-level representation of the form:

$$goal\ (G) \rightarrow plan(A_{11;}\ \ldots;\ A_{1n}) \qquad goal\ (G) \rightarrow plan(A_{21;}\ \ldots;\ A_{2m}) \qquad \ldots\ldots.$$

$$goal\ (G) \rightarrow plan(A_{p1;}\ \ldots;\ A_{pq}).$$

Myers [1997], Jarvis, Lunt, and Myers [2005] employ an HTN (similar to a production rule type) representation of the form:

$$G \rightarrow A_{11;}\ \ldots;\ A_{1n} \qquad G \rightarrow A_{21;}\ \ldots;\ A_{2m}\ \ \ldots\ldots \qquad G \rightarrow A_{p1;}\ \ldots;\ A_{pq}.$$

In Dragoni, Giorgini and Serafini [2002], which is perhaps one of the most complex approaches because of its use of bridge rules, a crucial part of the knowledge remains an implicit assumption, in particular the knowledge that links certain actions (in their case utterances of speech acts) and the intention behind them. Note that none of these representations is intended to be used for planning, thus although the declarative reading of the representations of Sindlar et al, Myers, and Jarvis, Lunt, and Myers are at some variant with the only-if part of completed planning rules, their usage, can, in effect, be interpreted as using abductive reasoning with such only-if representations.

In the next section we will look at a number of studies that essentially employ some form of logic-based approach for intention recognition. They cover diverse methodologies and applications. Most work described here falls in the ***keyhole*** or *intended* classification of intention recognition. Mulder and Voorbraak [2003] deal with enemy intention recognition, and thus conceptually fall in the *adversarial* classification.

We start with case studies (4.1, 4.2, 4.3) that use simpler forms of knowledge representation, primarily in the form of simple plan libraries, and consequently use simpler forms of reasoning. We then consider an approach based on BDI-agents (4.4), and move on to two approaches (4.5 and 4.6) that combine **logic** with probabilities. The second of these (4.6) also incorporates the notion of states as in the **situation calculus** [Reiter 2001]. We finish with two approaches (4.7 and 4.8), possibly the most complex of the eight, which incorporate some concept of theory of the mind, as bridge rules in the first, and epistemic operators in the second. This latter approach also incorporates an extension of the **event calculus** [Kowalski and Sergot 1986] as the background theory used for intention recognition.

4 Case Studies of Logic-Based Approaches

4.1 Mulder and Voorbraak [2003]

This paper addresses *tactical* intention recognition, which the authors define as the recognition of enemy plans. In general such a task will have various identifying characteristics, for example the specialized domain of the military, the tendency of the enemy to attempt to mislead the observers, or at the very least to try and avoid detection and prevent recognition of their plans and intentions. The paper makes a contribution related to the last of these features, by way of allowing observation of actions that do not convey all the information about the actions. In other words, and in logical terms, where other work below, and in the majority of the literature, assumes fully grounded observations, such as the action *land(jet101, airbase1)*, here the action may be non-ground and existentially quantified, for example $\exists X$ *land(X, airbase1)*, namely that it has been observed that something has landed at *airbase1*.

The work assumes fairly simple plan libraries, with rules of the form:

$$G \leftrightarrow A_1, \; ...., \; A_n,$$

where the A_i are actions, not defined by any other rules (i.e. they are simple rather than macro-actions). The "," denotes conjunction. The reasoning for intention recognition is abductive. So given a plan library P, and set of observations O, the reasoning seeks to find abductive explanations for all observations in O, i.e. it seeks to find sets of hypotheses Δ, made up of ground atoms in the predicates that appear in the goal side (the Gs) in the rules in P such that

$P \cup \Delta \nvdash O$.

The following example helps illustrate the contribution:

P: *attack-plan(airbase1)* $\leftrightarrow$ *land(X, airbase1), load(X, missiles, airbase1)*

aid-plan(airbase1) $\leftrightarrow$ *land(X, airbase1), load(X, aid-supplies, airbase1), cover-with-red-cross(X, airbase1)*

The first rule in *P* specifies that an attack is planned from *airbase1* if something lands at *airbase1* and is loaded with *missiles*. The second rule specifies that an aid flight is planned if what is loaded is *aid-supplies*, and the carrier is covered with the Red Cross symbol.

Now an observation *load(jet1, missiles, airbase1)* leads to one hypothesis, namely *attack-plan(airbase1)*. However, an observation $\exists Z$ *load(jet1, Z, airbase1)* leads to two potential hypotheses, namely *attack-plan(airbase1)* and *aid-plan(airbase1)*. A further observation $\exists X$ *cover-with-red-cross(X, airbase1)* still maintains two hypotheses, one consisting of *aid-plan(airbase1)* explaining both observations, and the other consisting of both *attack-plan(airbase1)* and *aid-plan(airbase1)*, each explaining one of the observations.

No proof procedures are suggested for the abduction, but it is worth noting that most, if not all, proof procedures for abductive logic programming will capture this reasoning (for example, the iff-proof procedure of Fung and Kowalski [1997] and the CIFF proof procedure of Mancarella et al. [2009]).

Note that with a slightly more elaborate representation of plan libraries it may well be possible to avoid reasoning with existentially quantified representations of observations. One such formalization could make use of binary representations of the known information about the observed actions, such as, for example for an event $e_1$of landing:

act(e_1, *land*) *destination*(e_1, *airbase1*),

ignoring what is not known, here the aircraft involved in e_1, and for an event e_2 of loading:

act(e_2, *load*) *base*(e_2, *airbase1*) *carrier*(e_2, *jet1*),

ignoring what is not known, here the cargo that is loaded. To accommodate such a representation the *attack-plan* rule, above, for example, can be written as:

attack-plan(airbase1) $\leftrightarrow$ *act(E1, land), destination(E1, airbase1), carrier(E1, X), act(E2, load) base(E2, airbase1), cargo(E2, missiles), carrier(E2, X)*,

ignoring any temporal constraints and persistence requirements between events *E1* and *E2*.

4.2 K. Myers [1997]

Myers' [1997] work is motivated by making planning technology more accessible and easier to use. Here the two parties, the observer and the actor are, respectively, a planning system and its user. The planning system observes the user's inputs and attempts to guess his intentions. Conventionally, in a planning system the user inputs a goal and the system provides a plan of actions that, if executed, would achieve the goal. In Myers' work the planning system is expected to do more than this. In effect, the planning is to be done co-operatively, where, as well as (optionally) inputting a goal, the user can participate in the planning process by providing the planning system with a partial plan which may consist of a (partial) list of tasks (actions or subgoals).

The system then attempts to identify from this partial plan any higher level goals the user may have, and then to complete the partial plan to achieve these higher level goals. For example let us consider the domain of travel planning. A traveler may provide a partial list of tasks, for example visiting the Swiss embassy and visiting a ski shop. A co-operative planner, in principle, may fill in the gaps, by deducing or guessing (abducing) that the user wishes to take a ski holiday trip to Switzerland, and can then complete the plan by including the actions of booking the flight and booking accommodation in Switzerland.

The top level goals come from a predefined set, either explicitly pre-defined by the programmer, or otherwise available to the system. The planner uses plan libraries, and in fact the same libraries, both for planning and for intention recognition. The plan libraries are based on the Hierarchical Task Network (HTN) model of planning [Erol, Hendler, Nau, 1994]. HTN defines operators for reducing goals to subgoals. For example:

$O1$: $B \rightarrow C, D$

$O2$: $C \rightarrow K, L, M$

$O3$: $C \rightarrow P, Z$

$O4$: $D \rightarrow W$

We comment later, at the end of this section, on the logic of HTNs and the above representation.

Here $O2$, for example, is an operator that reduces goal C to subgoals K, L, and M. So, given this HTN, for example if the user provides a goal C two plans are possible, one consisting of subgoals K, L, M, and the other of P, Z. On the other hand, if instead of providing a goal the user provides a partial plan consisting of P, the intention recognition system guesses that the intention is C, and ultimately B, and provides a plan P, Z, D, further refined to P, Z, W, compatibly with the user-given partial plan.

The planning is done conventionally, using the operators to reduce goals to subgoals. On the other hand, the determination of the user goals from the actions or subgoals they input is based on a form of abduction. For example, in the case above, B is an abductive explanation, according to Myers, for the user input subgoal P. This is determined using the notion of an *abductive chain* which in this case is:

$$P \rightarrow^{O3} C \rightarrow^{O1} B.$$

It is assumed that the operator definitions are non-recursive. If there are alternative top level goals possible via such abductive chains then a subset is chosen. The choice is not addressed in the paper. Once a set of goals is identified then the planning proceeds as in HTN with the constraint that the final plan should accommodate the user-given tasks. A brief analysis of the approach provides an exponential upper-bound for the process of constructing the abductive chains, but argues that this is dependent on the length of the chain and in practice this length is small, while the user-given partial plans can reduce the search space of the planning phase.

Note that the use of **abduction** here calls for a logical interpretation of the HTN representation. The operator decompositions are similar to goal-reduction rules in production systems. Kowalski and Sadri [2009] argue the difficulties of giving such rules model-theoretic semantics and provide an alternative framework that does provide such semantics. However, it seems relatively straightforward to put Myers' work in the context of the discussion of the different logic-based approaches in Section 3, and to relate the abduction done here to the formal notion of abduction defined there. This can be done by representing the operators in a logically meaningful way. For example, if we interpret operator definitions O1-O4, above as goal-reduction rules for goals *B*, *C* and *D*, we can have:

$$B \leftrightarrow C \wedge D$$

$$C \leftrightarrow (K \wedge L \wedge M) \vee (P \wedge Z)$$

$$D \leftrightarrow W.$$

Now the abductive reasoning required in the construction of the *abductive chains* is classical **abduction** using the rules above in the only-if direction ($\rightarrow$), as in Sindlar et al. [2008], described below. The planning can also be seen as classical planning using the *if* direction ($\leftarrow$) of the rules.

4.3 Jarvis, Lunt, and Myers [2004, 2005]

This work essentially uses the approach of Myers [1997] in the application of terrorist intention recognition. This more recent work uses an architecture called CAPRe (Computer-Aided Plan Recognition). Here the plan libraries are in the form of templates, and they contain more information than the earlier work [Myers 1997]. They are non-ground (i.e. contain non-ground parameters) and have additional information, such as ordering and preconditions of tasks, and *frequency* and *accuracy* of observations, described later.

An example, in a slightly simplified notation, is:

template Physical_Attack(?group, ?target)

purpose destroy(?group, ?target)

tasks 1. reconnaissance(?group, ?target)

2. prepare_attack(?group, ?target)

3. attack(?group, ?target);

Ordering 1-->3, 2-->3;

The template specifies a macro-action *Physical_Attack* of a target by a group. The purpose (effect) of the action is the destruction of the target by the group. The action is decomposed into three partially ordered actions, reconnaissance of the target and preparation of the attack, followed by the attack.

Ignoring the ordering and the variables this template corresponds to the HTN representation:

destroy $\rightarrow$ *Physical_Attack*

Physical_Attack $\rightarrow$ *reconnaissance, prepare_attack, attack*

in the earlier notation (Subsection 4.2). A more elaborate abstract template example, ignoring variables, may be:

template TaskName

purpose P

tasks $T_1, \ldots, T_n$

Ordering set of $T_i \dashrightarrow T_j$

Effects

E_1 *at* T_1, *frequency* F_1

....

E_n *at* T_n, *frequency* F_n;

The task *TaskName*, itself may be a subtask of another task, and so on. Ignoring the ordering and the *frequency* information the template above corresponds to the HTN representation:

$P \rightarrow TaskName$

$TaskName \rightarrow E_1, \ldots, E_n$

$E_1 \rightarrow T_1$

....

$E_n \rightarrow T_n$.

The *frequency* information represents the frequency of a given action occurring in *normal* circumstances. It is given values *high*, *medium* or *low*. For example car renting has a *high* frequency, whereas filing reports of missing persons has a *low* frequency. The templates may also include information about *accuracy* of normal observation, again given as *high*, *medium* or *low*. For example observing a missing person report has higher accuracy than recollecting a license plate. *Frequency* and *accuracy* can be thought of as qualitative measures used as a means of introducing probabilities in reasoning about choices.

The intention recognition is done in two phases. The first is mostly identical to the approach of Myers [1997], with so-called *task seedlings* generated from the templates in a way that is the similar to the generation of *abductive chains* from Hierarchical Task Networks. The variables are treated by unification, along the chain. The templates allow constructing chains starting from observation of actions T_i or effects E_i. Any additional *frequency* information can be used to specify maximum frequencies of observations for which abductive explanations are sought, thus, for example allowing the system to ignore commonly occurring (supposedly mundane) actions and changes in the environment. Similarly the *accuracy* information can also be used to impose a threshold on the observations that are taken into account.

The result of the first phase of intention recognition consists of explanations for each chosen observation. The second phase attempts to combine these to provide compatible explanations for clusters of observations. It does so by considering sets of increasing size of the explanations generated by the first phase, and removing from consideration incompatible sets. A set is incompatible if it has incompatible variable bindings, or incompatible orderings or other constraints within the templates.

As in the case of Myers [1997] the performance of the system is dependent on the length and number of the *task seedlings* (*abductive chains*). It is also dependent on the number of sets of explanations that have to be considered in the second phase. As might be expected, experimental results report a degradation in performance with increasing number of observations to be explained and increasing noise, i.e. activities that are unrelated to any attack plan.

It can be noted that the templates can be formalized in logic along the lines sketched in Subsection 4.2. For example the *Physical_Attack* template can be formalized as:

$$destroy(Group, Target, Time) \leftrightarrow physical\text{-}attack(Group, Target, Time)$$

$$physical\text{-}attack(Group, Target, Time) \leftrightarrow reconnaissance(Group, Target, Time1), prepare\text{-}attack(Group, Target, Time2), attack(Group, Target, Time), Time1<Time, Time2<Time$$

It would be interesting to explore if such a logic-based formalization, together with more recent abductive proof procedures with constraint handling, such as Mancarella et al. [2009] would have any impact on the performance of the system. Such an abductive proof procedure allows merging into one process the two phases of generation of explanations for each action and then finding compatible clusters. This merging may prevent generation of hypotheses and clusters that would in the end prove incompatible.

4.4 Sindlar, Dastani et al. [2008]

In this work the acting agent is assumed to be a BDI-type agent [Rao and Georgeff 1995], with the particular feature of interest being the planning rules that govern its behaviour. These rules are assumed to be of the form

$$G \leftarrow B \mid \pi$$

to be interpreted in logical terms as stating that goal G holds if B is believed and plan π is executed.

A plan may consist of sequences of actions and non-deterministic choice. Actions include test actions, that do not bring about any changes but simply check a condition. For example the planning rule below:

$$g \leftarrow b \mid a1; a2; (\emptyset 1?; ((\emptyset 2?; (a3; a4) + (\neg \emptyset 2?; a5))) + (\neg \emptyset 1?)); a6; a7$$

specifies that g is achieved if b is believed and actions $a1$ and $a2$ are executed and, if $\emptyset 1$ holds then if $\emptyset 2$ holds $a3$ and $a4$ are executed, otherwise $a5$ is executed, and then $a6$ and $a7$ are executed.

The observer agent may also be a BDI-type agent with a belief base, goals and behaviour rules. Those details are not relevant for the purposes of intention recognition. What is relevant and important here about the observer agent is that, in addition to any rules that govern its own (planning and action execution) behaviour, it has knowledge of the planning rules of the observed agent. This knowledge is in the form of rules of the form

$$goal(G) \wedge belief(B) \rightarrow plan(\pi) \qquad \text{RD}$$

The agent is assumed to have one such rule for every rule of the form

$$G \leftarrow B|\ \pi$$

in the observed agent's knowledge base. Rules of the form RD are not used for planning purposes, and are not particularly appropriate for such purposes, but they are used specifically for intention recognition. They are used abductively to provide explanations for observed actions in terms of the goals and beliefs of the observed agent. The abducible predicates are *goal* and *belief*. Thus although the observer agent is a BDI-type agent, the rules it uses for intention recognition have logical semantics, and they lend themselves to abductive reasoning.

The work is based on several assumptions: One is that the observed agents execute one plan at a time, so there are no interleaved plan executions. Another is that the observer is able to determine ahead of time whether they have *complete* observability or whether their observations will be *late* or *partial*. *Complete* observation means that all the non-test actions of the actor will be observed. *Late* observation means that the observer may miss some of the initial actions of the plan executed but from the first observation it will observe all the non-test actions executed henceforth. *Partial* observation means that the observer may miss some of the actions, anywhere in the plan. In all three cases all the actions that are observed are assumed to be observed in the correct sequence.

In each of the three cases a match is sought between the sequence, S, of actions that have been observed and the sequence in π in the conclusion of rules of the form RD. The test actions in π are ignored in this matching process. In the case of *complete* observability the match sought is between S and any initial segments of π (for example *a1;a2* or *a1;a2;a3;a4* or any other initial segment of *a1;a2; a3;a4;a6;a7* or *a1;a2; a5;a6;a7* or *a1;a2; a6;a7* in the above example). In the case of *late* observation the match sought is between S and any sub-segments of π (for example a1;a2 or a2;a3;a4 or a2;a5 or …). In the case of *partial* observation the match sought is between S and any *dilution* of any sub-segment of π, i.e. any sub-segment with any elements removed (for example a2;a4 or a1;a4;a7 or …).

When such a match is found the conditions $goal(G) \wedge belief(B)$ of the RD rule provide the necessary abductions and thus the hypothesis regarding the intention of the acting agent.[2] We can interpret what is done here in the general abductive framework outlined in Section 3 by considering the theory used for abduction to consist of the rules of the form

[2] In the paper both the planning rules of the observed agent and rules of the form RD of the observer agent are assumed to be ground.

$$goal(G) \wedge belief(B) \rightarrow (a_1;a_2;\ldots;a_n) \vee \ldots \vee (a'_1; a'_2; ..; a'_m)$$

where the a_i and a'_i are non-test actions, and ";" can be thought of, as is common, as representing conjunctions of actions and their sequential ordering. The three cases of observability can be thought of as providing additional integrity constraints. *Complete* and *late* observability both impose the constraint that there is no action in between any two actions observed where one immediately follows the other. *Complete* observability additionally imposes the constraint that there is no action before the first one observed. *Partial* observability imposes neither constraint.

Two quite intuitive propositions are proved. One states that the number of possible explanations (abductive hypotheses) generated in case of *complete* observability is less or the same as that generated by *late* observation which in turn is less or the same as *partial* observation. The other is that in each case the number of possible explanations decreases or at most stays the same as the number of observations increases.

4.5 Demolombe and Fernandez [2006]

This paper proposes a framework that combines probabilities with a **situation calculus**-like formalization of actions. The **situation calculus** [Reiter 2001] and its further extension GOLOG [Levesque et al 1997] are logical formalisms for specifying actions and their effects. They allow specifying macro-actions or, in the authors' terminology, procedures. For example, the procedure for dealing with a fire on board a plane may be represented as:

$$\textit{tackle-fire-on-board} =^{def} \textit{turn-fuel-off; turn-full-throttle; turn-mixture-off}$$

to express the sequence of actions *turn fuel off*, *turn full throttle* and *turn mixture off*. Note the similarity between this notation and the HTN concept and representation.

Each single atomic action maps one state to its successor, and correspondingly macro-actions map one state to another. The assumption of the paper is that the actor and the observer, essentially, have the same "knowledge" about such procedures. However, a central aim of the paper is to allow intention recognition of human actors who may interleave procedures. To this end for each procedure that is in the knowledge base of the actor a modified one is included in the knowledge base of the observer. The modification is done by explicitly adding arbitrary actions, and any constraints, to the procedure definition. For example the definition above is modified to the following:

$$\textit{tackle-fire-on-board} =^{def} \textit{turn-fuel-off; } (\sigma/\textit{turn-fuel-on}); \textit{ turn-full-throttle; } (\sigma/\textit{turn-fuel-on});$$
$$\textit{turn-mixture-off.} \qquad \text{OD}$$

This modified definition represents the same procedure for tackling fire on board, but explicitly allows any sequence of actions σ, except *turn-fuel-on* in between turning the fuel off and turning full throttle, and between turning full throttle and turning the mixture off. Here, in the *fire-on-board* procedure, the three actions *turn-fuel-off*, *turn-full-throttle, turn-mixture-off* are *explicit* actions, and the action *turn-fuel-on* is said to be *prohibited* in between any two of the *explicit* actions. The other actions in σ are said to be *tolerated*. The set of all defined procedures provides the set of all possible intentions.

Those familiar with formalisms such as the **situation calculus** or the **event calculus** [Kowalski and Sergot 1986] may relate the observer definition OD to a formalization that makes explicit the required persistence of properties between actions or states. An example of such formalization, letting the *Si* to represent states or time points, is:

happen(tackle-fire-on-board, S3) ← happen(turn-fuel-off, S1), happen(turn-full-throttle, S2), happen(turn-mixture-off, S3), S1<S2, S2<S3, persists(S1, fuel-off, , S3)

persists(S1, fuel-off, , S3)← ¬(happens(turn-fuel-on, S), S1<S, S<S3),[3]

where action *turn-fuel-off* is specified to have the effect *fuel-off*.

The paper assumes that there is full visibility of actions performed, or rather that the intention recognition is performed purely on the basis of the observed actions. In the initial state before any action is observed all possible intentions are given pre-specified probabilities. Then as actions are observed they are matched with the procedure definitions. The matching includes *explicit* actions in procedure definitions, such as *turn-fuel-off* in the example above, as well as any *prohibited* or *tolerated* actions.

With each match probabilities are updated for the possible intentions. The probability of an intention is increased, if the next expected *explicit* action is observed, it is decreased if a *tolerated* action is observed, and decreased to a greater extent if a *prohibited* action is observed. All the probability increments and decrements are by pre-specified amounts. The computation cost of evaluating the probabilities is said to be linear with respect to the number of observations for a given procedure.

The following is a modified version of an example in the paper. Consider three procedures

$P1 = a; (\sigma/g); b; c$

$P2 = d; \sigma; e$

$P3 = a; \sigma; f$

where, σ denotes a sequence of arbitrary actions. If in the initial state *s0* action *f* is observed then the probabilities of the agent having the intention *P1*, *P2* or *P3* in state *s0* are equal and low. So *P(Int(P1, s0))=P(Int(P2, s0))= P(Int(P3, s0)),* where *P(Int(Q,S))* denotes the probability of intention *Q* in state *S*. Then in the resulting state, *s1=do([f],s0)* in modified **situation calculus** notation, if action *a* is observed, the probabilities of *P1* and *P3* are increased equally in state *s1*. Now if an action *m* is observed in state *s2=do([f,a],s0)* there is still a match with *P1* and *P3,* but action *m* lowers the probability of both *P1* and P3, thus, for example *P(Int(P1, s0))< P(Int(P1, s1))* and *P(Int(P1, s2))< P(Int(P1, s1))*. If now an action

---

[3] Alternatively to conform to the notation we introduced in Section 3 we could write:
persists(T1,P,T) ← not clipped(T1,P,T)
clipped(T1,P,T) ←happens(E, T2) ∧ terminates(E,P) ∧ not out(T2,T1,T)
terminates(turn-fuel-on, ,fuel-off).

g is observed in state *s3*, where *s3= do([f,a, m],s0)*, it reduces the probability of *P1*, because *g* is a prohibited action for *P1* in sate s3. But the observation of *g* does not affect the probability of *P3*, thus *P(Int(P1, s3))< P(Int(P3, s3))*.

4.6 Pereira and Anh [2009b]

This paper describes an implemented logic programming framework incorporating *situation-sensitive Causal Bayes Nets* (CBNs) for intention recognition in the care of the elderly. The CBNs provide a graphical representation and are translated into a declarative language called P-log which represents the same information in logical terms. P-log combines logical and probabilistic reasoning, the logical part based on Answer Set Programming (ASP) [Hu et al. 2007] and the probabilistic part based on CBNs.

The CBNs consist of nodes representing causes, intentions, actions and effects of actions. Causes give rise to intentions, somewhat like reactive production rules (e.g. *if you are thirsty then you (intend to) drink*). Intentions give rise to actions, somewhat like goal reduction production rules (e.g. *if you intend to drink then you look for a drink*), as in Fig.1, where we have ignored effects of actions.

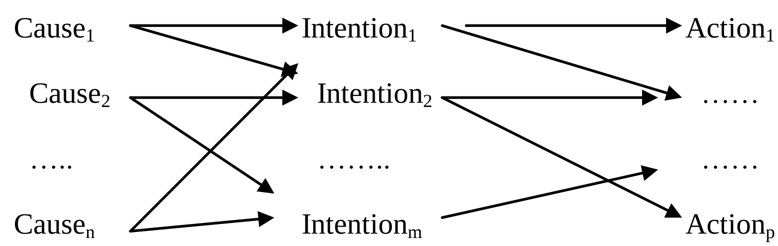

Fig. 1.An abstract CBN

In the running example in the paper there is one action, which the elder is observed to be doing, namely looking for something in the sitting room. The possible intentions from which the system can choose from are looking for a book or for a drink or for the TV remote or the light switch. The causes include that he is thirsty, likes to read, or likes to watch TV.

The paper makes no assumptions about the planning methodology or representation of the observed agent or about its relationship to the knowledge and representation used by the observer. The approach is entirely based on the knowledge representation the observer uses specifically for intention recognition. The observer's statistical knowledge is compiled as CBNs. The causes in the resulting CBNs are either attributes that can be observed, for example *the light is on* or *the TV is on*, or are attributes that can be surmised, for example (*the elder is) thirsty*. The causes have pre-specified probabilities, either unconditional probabilities represented as facts (in P-log), for example:

the probability of being thirsty is 50/100,

or situation-sensitive probabilities represented as rules, for example:

the probability of being thirsty is 70/100 ← *temperature is higher than 30.*

Similarly the probability distributions of intentions conditional on causes are given as rules, for example:

the probability of the intention to look for a drink is 9/10 ← *the light is on* ∧ *thirsty*, as are the probability distributions of actions conditional on intentions, for example:

> *the probability of looking for something is 99/100* $\leftarrow$ *the intention to look for a book is true* $\wedge$ *the intention to look for a drink is true* $\wedge$ *the intention to look for the remote is true*

> *the probability of looking for something is 30/100* $\leftarrow$ *the intention to look for a book is false* $\wedge$ *the intention to look for a drink is true* $\wedge$ *the intention to look for the remote is false*.

(Note the conjunction of possible intentions in the conditions of these rules, especially in the first rule, rather than the disjunction). All the probability distributions, conditional and unconditional, as well as the list of all possible intentions, causes and actions are pre-specified by the designer of the system.

Given this formalization, intended specifically for a P-log implementation, the probabilities of each of the possible intentions can be obtained conditional on observations regarding the status of the light and TV (on/off), temperature (all related to the causes), and the observation that the elder is looking for something.

It is difficult to frame this approach in the methodologies seen in Section 3. Whilst the reasoning from actions to possible intentions has the flavour of abduction, here the explanation is a conjunction of possible intentions rather than a disjunction. This is because of the use of Bayesian networks. Each possible intention is exclusive of the others, and thus the choice of one intention excludes the others.

Notice that there is some similarity between this work and that of Mulder and Voorbraak, described in Subsection 4.1. In both the observer does not have all the information about the action it observes (here he observes that the elder is looking for *something*, for example). Here, however, the intentions are designed to provide hypotheses as to what the missing information may be (the *something* is a book, or the T.V. remote, for example).

4.7 Dragoni, Giorgini and Serafini [2002]

This paper explores the use of abductive reasoning in intention recognition in the context of agent communication. Here the two parties are the hearer (*h*) and the speaker (*s*) of a speech act. The hearer attempts to recognise the beliefs and intentions of the speaker that led to the speaker producing the communication. To this end a *multi-context* system of beliefs and intentions is proposed where the contexts are connected via bridge rules. The contexts correspond to agents' beliefs and intentions, and nested beliefs and intentions about other agents' beliefs and intentions. For example B_iI_j is a context, representing the beliefs of agent *i* about the intentions of agent j. The formula $\varnothing$ in this context, denoted $B_iI_j:\varnothing$, states that agent *i* believes that agent *j* intends $\varnothing$. $B_iB_jI_i$ is another context, representing the beliefs of agent *i* about the beliefs of agent *j* about the intentions of agent *i*. The formula $\varnothing$ in this context, denoted $B_iB_jI_i:\varnothing$, states that agent *i* believes that agent *j* believes that *i* intends $\varnothing$.

There are three kinds of bridge rules: *reflection down*, *reflection up*, and *belief-to-intention* rules. *Reflection down* allows an agent *i* to reason with its image of the mental state of another agent *j* (for example eliminating B_j from $B_j\varnothing$). *Reflection up* allows agent *i* to lift up the result of such reasoning to ascribe it to its beliefs about agent *j*. For example:

> If B_i: B_jP and B_i: $B_j(P\rightarrow Q)$ then by *reflection down* B_iB_j: P and B_iB_j: $(P\rightarrow Q)$.

Then within the context B_iB_j, by modus ponens we can derive Q. Thus B_iB_j*: Q*, and by *reflection up* we obtain B_i*:* B_jQ.

The *belief-to-intention* rules connect an agent's intention to its belief, for example:

$$\frac{B_i\text{: raining}}{I_i\text{: bring-umbrella}} \quad \text{or} \quad \frac{B_i\text{: temp-higher-}20^\circ \wedge \text{conditioning-on}}{I_i\text{: stop-working}}$$

The first rule, above, states that if *i* believes it is raining then *i* will have the intention of bringing an umbrella. The second rule states that if *i* believes that the temperature is higher than 20° and the conditioning is on then *i* will have the intention to stop working. The *belief-to-intention* rules have the flavour of simple reactive production rules. In particular they capture the relationship between the beliefs and intentions of the hearer. On the other hand, what the hearer believes about the reactive behaviour rules of the speaker are formalized as ordinary rules. For example in the context B_h the rule

$$B_s(\text{temp-higher-}20^\circ \wedge \text{conditioning-on}) \rightarrow I_s(\text{stop-working})$$

represents what *h* believes about *s*'s intention when *s* believes the temperature is higher than 20° and the conditioning is on. The underlying language of the multi-context system is propositional logic, and the inference within contexts is based on natural deduction.

The work assumes a plan-based model of speech acts, in the sense that speech acts, as any other actions, have pre-conditions and post-conditions. Thus, an agent might utter certain speech acts as part of a plan to achieve an intention. The core assumption in the paper is thus that there is a causal relationship between an agent's mental state and his utterances.

Two speech acts are considered, *inform(s,h,*$\varnothing$*)* and *request(s,h,*$\varnothing$*)*, where *s* represents the speaker and *h*, the hearer. *inform(s,h,*$\varnothing$*)* represents *s* telling *h* that $\varnothing$ holds, and *request(s,h,*$\varnothing$*)* represents *s* asking *h* whether or not $\varnothing$ holds. A pre-condition and a post-condition of *inform(s,h,*$\varnothing$*)* are, respectively, that *s* believes $\varnothing$, and *h* believes that *s* believes $\varnothing$. Similarly, a pre-condition and a post-condition of *request(s,h,*$\varnothing$*)* are, respectively, that *s* believes neither $\varnothing$ nor $\neg\varnothing$, and *s* believes that *h* believes that *s* intends to believe $\varnothing$.

The hearing agent does not necessarily have any representation of the planning rules of the speaker, in contrast, say, with the work of Sardar, et al [2008]. The hearing agent may have some snippets of the behaviour rules of the speaker, such as the rule:

$$B_s(\text{ temp-higher-}20^\circ \wedge \text{conditioning-on}) \rightarrow I_s(\text{stop-working}),$$

but he does not have any explicit representation of any planning rules of the speaker that link the speaker's intentions to his utterances of speech acts. The absence from the knowledge of the hearer of any explicit representation of any link between the intentions of the speaker and the actions that the hearer can observe (i.e. utterances of speech acts) is one crucial difference between this work and all others in this study, and, in fact in the majority of work in the literature on intention recognition.

In the absence of any such explicit information, the hearing agent works with the assumption that the intention behind a speech act *inform(s,h,*$\varnothing$*)* is to bring about in *h* a mental state in which *h* believes or intends some formula ψ, and the intention behind a speech act *request(s,h,*$\varnothing$*)* is to bring about in *s* a mental state in which s believes or intends some formula ψ. Also in both cases $\varnothing$ is expected to be useful for the derivation of the belief or intention ψ. So the intention recognition task with respect to both utterances is to determine what the ψ is, and whether the intention is to believe ψ or to intend ψ.

So, to attempt to put this in the frameworks discussed in Section 3, conceptually, it is as if the hearer has rules of the form below with which it performs **abduction**:

If $I_s I_h \psi$ and "$\varnothing$ is relevant to ψ" then inform(s,h,$\varnothing$)

If $I_s B_h \psi$ and "$\varnothing$ is relevant to ψ" then inform(s,h,$\varnothing$)

If $I_s I_s \psi$ and "$\varnothing$ is relevant to ψ" then request(s,h,$\varnothing$)

If $I_s B_s \psi$ and "$\varnothing$ is relevant to ψ" then request(s,h,$\varnothing$).

When h receives a speech act *inform(s,h,$\varnothing$)*, to find out what such a ψ may be, and to determine if the intention is for him to believe or intend ψ, he reasons about the changes that $\varnothing$ brings to its beliefs and intentions. It can reason about the consequences of $\varnothing$ within any context available to it. For example, an appropriate context would be what h believes s believes about h's beliefs, i.e. the context $B_h B_s B_h$. The new consequences of $\varnothing$, i.e. the consequences that were not derivable before the addition of $\varnothing$, but are derivable afterwards, are candidate hypotheses for the intention of the speaker.

A similar process is adapted with the speech act *request(s,h,$\varnothing$)*. As an example suppose h receives a communication *request(s, h, conditioning-on)*, to be interpreted as s asking h if the conditioning is on. Then if h can prove within its image of the mental state of s that *temp-higher-20°* and the image also includes or allows the derivation of the rule

$$\textit{temp-higher-20}^\circ \wedge \textit{conditioning-on} \rightarrow \textit{stop-working},$$

then h can hypothesise that s's intention behind the speech act is to stop working.

It is interesting to contrast this approach with the abductive approach of Quaresma and Lopes [1995], described below, which also focuses on recognizing the intentions behind the same speech acts. One of the main differences between the two is that the plan-based model of the speech acts, in terms of their preconditions and effects are made explicit in Quaresma and Lopes. Another main difference is that Quaresma and Lopes avoid using bridge rules.

4.8 Quaresma and Lopes [1995]

This work uses an abductive version of the **event calculus** [Kowalski and Sergot 1986] and a modified version of the action language of Gelfond and Lifschitz [1992] for recognizing the intention behind speech acts. As with Dragoni et al., it focuses on the *request* and *inform* speech acts. This work differs from the majority of the papers on intention recognition in that it uses a theory of planning from first principles as the background theory for intention recognition. This background theory would presumably be used by the speaker, s, to reduce goals to subgoals, and thus plan, and is used by the hearer, h, with a mixture of deductive and abductive reasoning, for intention recognition.

The background theory is a modification of the **event calculus** which is a causal theory formalized in Horn clause logic augmented with negation as failure. Here are their core **event calculus** rules, which have minor differences with those we presented in Section 3:

$$holds\text{-}at(P,T) \leftarrow happens(E), initiates(E,P), succeeds(E), E<T, persists(E,P,T)$$

$$persists(E,P,T) \leftarrow not\ clipped(E,P,T)$$

$$clipped(E,P,T) \leftarrow happens(C), terminates(C,P), succeeds(C), not\ out(C,E,T)$$

$$out(C,E,T) \leftarrow T=C \qquad out(C,E,T) \leftarrow T<C \qquad out(C,E,T) \leftarrow C<E.$$

They state that a property *P* holds at time *T* if an earlier event *E* happens which initiates *P* and *E* succeeds and *P* persists (at least) from the occurrence of *E* until *T*. *P* persists between two times, (the time of) *E* and *T*, if it is not clipped in that interval. It is clipped in that interval if an event *C* happens and succeeds and it cannot be shown that *C* occurred outside that interval. Here *not* is negation as failure.

An event *E* succeeds if its preconditions hold at the time of its occurrence, i.e.:

$$\text{succeeds(E)} \leftarrow act(E, A), hold\text{-}at(P_1,E), \ldots.., holds\text{-}at(p_n,E)$$

where $P_1, \ldots, P_n$ and the preconditions of the action operator *A* of event *E*.

These core rules are augmented by the following rules that enable abductions:

$$F \leftarrow not\ \neg F \qquad \text{R1}$$

$$\neg F \leftarrow not\ F \qquad \text{R2}$$

where ¬ is classical negation, and *F* is an abducible predicate. These rules model abductive reasoning and state that if it is not possible to prove ¬*F* then *F* should hold, and vice versa. In the intention recognition framework the abducible predicates are *happens/1*, *act/2* and *</2*. The semantics assumed is that of Well Founded Semantics of Extended Logic Programs [Pereira et al. 1992].

Epistemic operators are used to describe the agents' mental state, for example:

int(a, actn) to specify that agent *a* wants action *actn* to be done

bel(a, p) to specify that agent *a* believes that *p* is true

ach(a, p) to specify that agent *a* believes *p* will be achieved as a consequence of the actions of some agent (itself or others).

Event-calculus-type rules describe the relationships between the components of the mental state (as ramifications), for example, the rule:

$$holds\_at(int(A,to(\alpha,P)), T) \leftarrow holds\_at(bel(A, to(\alpha,P), T) \wedge holds\_at(int(A, \alpha), T) \wedge holds\_at(ach(A,P), T) \qquad \text{R3}$$

where $to(\alpha,P)$ is a term representing the plan of performing α to make P true. The rule specifies that if an agent A believes that by doing α P will become true and A intends to do α, and A believes P will be achieved, then A intends to do α in order to make P true. Furthermore, such rules are augmented by an integrity constraint:

$$holds\_at(bel(A,P), T), holds\_at(ach(A,P), T) \Rightarrow false,$$

stating that at no time does the agent believe both that P is true and P will be achieved.

Speech acts are specified as actions within the **event calculus** framework, for example:

$$succeeds(E) \leftarrow act(E, inform(S, H,P)), hold\text{-}at(bel(S,P),E), holds\text{-}at(bel(S, int(S, inform(S,H,P))),E)$$

$$initiates(E, bel(H, bel(S,P))) \leftarrow act(E, inform(S, H,P))$$

state that a speech act event of S informing H some information P succeeds if S believes P and intends to inform H about it, and it initiates H believing that S believes P. Similarly for the *request* speech act:

$$succeeds(E) \leftarrow act(E, request(S, H, A)), hold\text{-}at(bel(S,cando(H, A)),E), holds\text{-}at(bel(S, int(S, request(S, H, A))),E) \qquad \text{R4}$$

$$initiates(E, bel(H, bel(S,int(S,A)))) \leftarrow act(E, request(S, H, A)) \qquad \text{R5}$$

stating that the preconditions of S requesting H to do an action A are that S believes H can do the action and S intends to make the request, and the action has the effect that H believes S believes it wants action A done.[4]

Further rules can provide a relationship between the speaker and hearer, for example trust:

$$holds\text{-}at(bel(H,P), T) \leftarrow holds\text{-}at(bel(H, bel(S,P)), T).$$

Now, in principle, the hearer, *h*, receiving, for example, a *request* speech act to do an action *A* will deductively entail the initiation of a belief in *h* that *s* intends *A* (R5). This, in turn, through deduction and also abductions of *</2*, *happens*, and *act* atoms, leads to hypotheses about what *s* intends to achieve (using rules including the core rules, R1, R2, R3, R4).

5 Conclusion and Challenges

In this paper we discussed logic-based approaches to intention recognition. We looked at different knowledge representation and reasoning mechanisms and we looked at the relationship between the two.

[4] Note that here, slightly differently from Dragoni et al., the *request* is aimed at requesting that an action be done. But, in practice, by requesting that an *inform* action be done, the effect is the same, i.e. to request some information.

We then described and analysed a number of concrete contributions in this field. The following table summarises some of the features of these contributions.

	Formalism	Reasoning	Application	Requires full observability?
Mulder & Voorbraak	Simple plan libraries	Abductive	Tactical/Military	No, allows existentially quantified variables in observations
Myers	HTN plan libraries	Abductive	Co-operative planning	Yes
Jarvis et al.	Extended HTN plan libraries	Abductive	Terrorist intention recognition	No, allows varying degrees of accuracy in observations
Sindlar et al.	Transformed BDI-type plan libraries	Abductive	Generic	No, allows partial observability
Demolombe& Fernandez	Situation calculus and GOLOG	Probabilistic	Generic, but with focus on recognizing intentions of airplane pilots	Yes
Pereira & Anh	Causal Bayes nets	Baysian	Generic, but with focus on elder care	No, allows partial observability
Dragoni et al.	Multi-context logical theories with bridge rules	Abductive	Recognising intentions behind speech act utterances	Yes
Quaresma & Lopes	Event calculus	Abductive	Recognising intentions behind speech act utterances	Yes

Table 1. A summary of some of the features of the reviewed works

Intention recognition has been a long-standing research area. Not surprisingly, its application areas have changed during the intervening years, moving from Unix help facilities and language understanding to broader areas of ambient intelligence and intrusion and terrorism detection. Advances in activity recognition, sensor technology and RFID tags and readers allow further developments towards realistic and difficult applications.

Many challenges remain. Much of the current work assumes the existence of plan libraries. This requires much human effort in predicting and formalizing plans, and may be unrealistic in many cases. It may also be unrealistic to assume that the observer agent has knowledge of the plan libraries of the observed agent. Furthermore, the intention recognition system is hampered in cases where an agent attempts novel ways of achieving a goal.

Much work on intention and plan recognition assumes one single observed agent, with one single intention and plan. Consequently it attempts to find hypotheses consisting of one intention or one plan that explains all the observed actions of that agent, with the possible exception of what it considers as “insignificant” actions. However, terrorism detection and other applications increase the demand for multi-agent intention recognition, where several agents co-operate on the same or on several related intentions. This brings with it many challenges, for example to identify related clusters of agents, to identify which agents are contributing to which intentions and to identify intentions and plans from interleaved actions. The challenges become harder if agents deliberately act to mislead the observers.

Dealing with partial observability of actions, whether in a single agent case or a multi-agent one remains an issue. Current solutions include attempting to detect that an action has been executed from changes in the environment (Geib and Goldman 2001), even though the action itself has not been observed, and attaching probabilities to non-observation of actions (Geib and Goldman 2005). However, a general, efficient and extendible solution remains a challenge. This may be an area where abduction can make a further contribution in intention recognition. Abductive reasoning can be used to generate hypotheses about what actions have been executed, unobserved, compatibly with actions that have been observed, so that together they can account for the changes in the environment.

Most systems in **keyhole** and intended recognition cases assume that the actor is rational and is following a correct plan. This may not be the case in some applications. For example in tutoring systems the student may make mistakes, and in ambient intelligence systems the actor, maybe an Alzheimer patient, may execute actions in confusion.

Acknowledgement

I am grateful to Bob Kowalski for discussions about the topic of intention recognition, and to him and Luis Pereira for reading and commenting on earlier drafts of this paper. I am also grateful to the anonymous reviewers.

References

Appelt, D.E., Pollack, M.E. (1992). Weighted Abduction for Plan Ascription. *Journal of User Modeling and User-Adapted Interaction*, Vol. 2, Nos 1-2, March 1992, 1-25.

Avrahami-Zilberbrand, D., Kaminka, G.A.(2005). Fast and complete symbolic plan recognition. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI) 2005, 653-658.

Avrahami-Zilberbrand, D., Kaminka, G.A. (2007). Incorporating observer biases in keyhole plan recognition (efficiently!). In Proceedings of the Twenty-Second National Conference on Artificial Intelligence (AAAI-07), 2007.

Barger, T., Alwan, M., Kell, S., Turner, B., Wood, S., Naidu, A. (2002) Objective remote assessment of activities of daily living: analysis of meal preparation patterns. Poster presentation, http://marc.med.virginia.edu/pdfs/library/ADL.pdf, 2002.

Bui, H. H. (2003). A general model for online probabilistic plan recognition. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI) 2003.

Carberry, S., Elzer, S. (2007). Exploiting evidence analysis in plan recognition. UM 2007, LNAI 4511 C. Conati, K. McCoy, G. Paliouras (Eds.), Springer-Verlag Berlin Heidelberg 2007, 7-16.

Charniak, E., McDermott, D. (1985). *Introduction to artificial intelligence*. Addison Wesley, Reading, MA, 1985.

Cheng, D. C., Thawonmas, R. (2004). Case-based plan recognition for real-time strategy games. In Proceedings of the 5th Game-On International Conference (CGAIDE) 2004, Reading, UK, Nov. 2004, 36-40.

Cohen, P.R., Perrault, C.R., Allen, J.F. (1981). Beyond question answering. In Strategies for Natural Language Processing, W. Lehnert and M. Ringle (Eds.), Lawrence Erlbaum Associates, Hillsdale, NJ, 1981, 245-274.

Cox, M.T., Kerkez, B. (2006).Case-Based Plan Recognition with Novel Input. *International Journal of Control and Intelligent Systems* 34(2), 2006, 96-104.

Demolombe, R., Fernandez, A.M.O. (2006). Intention recognition in the situation calculus and probability theory frameworks. In Proceedings of Computational Logic in Multi-agent Systems (CLIMA) 2006.

Dragoni, A.F., Giorgini, P., Serafini, L. (2002) Mental States Recognition from Communication. *Journal of Logic Computation*, Vol 12, No. 1, 2002, 119-136.

Erol, K., Hendler, J., Nau, D. (1994). Semantics for hierarchical task-network planning. CS-TR-3239, University of Maryland, 1994.

Fung, T.H., Kowalski, R. (1997). The IFF Proof Procedure for Abductive Logic Programming. *Journal of Logic Programming*, 1997.

Geib, C. W., Goldman, R. P. (2001). Plan recognition in intrusion detection systems. In the Proceedings of the DARPA Information Survivability Conference and Exposition (DISCEX), June, 2001.

Geib, C. W., Steedman, M. (2003). On natural language processing and plan recognition. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI) 2003, 1612- 1617.

Geib, C. W., Goldman, R. P. (2005). Partial observability and probabilistic plan/goal recognition. In Proceedings of the 2005 International Workshop on Modeling Others from Observations (MOO-2005), July 2005.

Gelfond, M, Lifschitz, V. (1992). Representing actions in extended logic programs. In Proceedings of the International Symposium on Logic Programming, 1992.

Giroux S., Bauchet J., Pigot, H., Lusser-Desrochers, D., Lachappelle, Y. (2008). Pervasive behavior tracking for cognitive assistance. The 3rd International Conference on Pervasive Technologies Related to Assistive Environments, Petra'08, Greece, July 15-19, 2008.

Goodman, B. A., Litman, D. J. (1992). On the interaction between plan recognition and intelligent interfaces. User Modeling and User-Adapted Interactions 2(1-2), 1992, 83-115.

Hu, P.H., Son, T.C., Baral, C. (2007). Reasoning and planning with sensing actions, incomplete information, and static laws using Answer Set Programming. *Theory and Practice of Logic Programming* 7(4), July 2007, 377-450.

Jarvis, P., Lunt, T., Myers, K. (2004). Identifying terrorist activity with AI plan recognition technology. In the Sixteenth Innovative Applications of Artificial Intelligence Conference (IAAI 04), AAAI Press, 2004.

Jarvis, P.A., Lunt, T.F., Myers, K.L. (2005). Identifying terrorist activity with AI plan-recognition technology, *AI Magazine*, Vol 26, No. 3, 2005, 73-81.

Kakas, A., Kowalski, R., Toni, F. (1998). The role of logic programming in abduction. In: Gabbay, D., Hogger, C.J., Robinson, J.A. (Eds.): Handbook of Logic in Artificial Intelligence and Programming 5, Oxford University Press, 1998, 235-324.

Karlsson, B., Ciarlini A.E.M., Feijo B., Furtado A.L. (2006). Applying a plan-recognition/plan-generation paradigm to interactive storytelling, The LOGTELL Case Study. Monografias em Ciência da Computação Series (MCC 24/07), ISSN 0103-9741, Informatics Department/PUC-Rio, Rio de Janeiro, Brazil, September 2007. Also in the Proceedings of the ICAPS06 Workshop on AI Planning for Computer Games and Synthetic Characters, Lake District, UK, June, 2006, 31-40.

Kautz, H., Allen, J.F. (1986). Generalized plan recognition. In Proceedings of the Fifth National Conference on Artificial Intelligence (AAAI-86), 1986, 32-38.

Konolige, K., Pollack, M.E. (1989). Ascribing plans to agents: preliminary report. In 11th International Joint Conference on Artificial Intelligence, Detroit, MI, 1989, 924-930.

Kowalski, R. A., Sadri, F. (2009). Integrating logic programming and production systems in abductive logic programming agents. Invited papers, the 3rd International Conference on Web Reasoning and Rule Systems, October 25-26, Chantilly, Virginia, USA, 2009.

Kowalski, R., Sergot, M. (1986). A logic-based calculus of events. In *New Generation Computing*, Vol. 4, No.1, February 1986, 67-95.

Lesh, N., Rich, C., Sidner, C. L. (1999). Using plan recognition in human-computer collaboration. In Proceedings of the Seventh International Conference on User Modelling, Canada, July 1999, 23-32.

Levesque, H., Reiter, R., Lesperance, Y., Lin, F., Scherl, R. (1997). GOLOG: A Logic Programming Language for Dynamic Domains. *Journal of Logic Programming* 31, 1997, 59-84.

Mancarella, P., Terreni, G., Sadri, F., Toni, F., Endriss, U. (2009). The CIFF proof procedure for abductive logic programming with constraints: theory, implementation and experiments. *Theory and Practice of Logic Programming* 9, 2009.

Mayfiled, J. (2000). Evaluating plan recognition systems: three properties of a good explanation. *Artificial Intelligence Review* 14, 2000, 351-376.

Mao, W., Gratch, J. (2004). A utility-based approach to intention recognition. AAMAS 2004 Workshop on Agent Tracking: Modelling Other Agents from Observations.

Mihailidis, A., Fernie, G.R., Barbenel, J. (2001). The use of artificial intelligence in the design of an intelligent cognitive orthosis for people with dementia. *Assistive Technology* 13:23-39, 2001.

Modayil, J., Bai, T., Kautz, H. (2008). Improving the recognition of interleaved activities. In the Proceedings of the 10th International Conference on Ubiquitous Computing (UbiComp), 2008, 40-43.

Mulder, F., Voorbraak, F. (2003). A formal description of tactical plan recognition, *Information Fusion* 4, 2003, 47-61.

Myers, K.L. (1997). Abductive completion of plan sketches. In Proceedings of the 14th National Conference on Artificial Intelligence, American Association of Artificial Intelligence (AAAI-97), Menlo Park, CA: AAAI Press, 1997.

Myers, K. L., Jarvis, P. A., Tyson, W. M., Wolverton, M. J. (2003). A mixed-initiative framework for robust plan sketching. In Proceedings of the 13thInternational Conferences on AI Planning and Scheduling, Trento, Italy, June, 2003.

Park, S., Kautz, H. (2008). Hierarchical recognition of activities of daily living using multi-scale, multi-perspective vision and RFID. The 4th IET International Conference on Intelligent Environments, Seattle, WA, July 2008a.

Park, S., Kautz, H. (2008). Privacy-preserving recognition of activities in daily living from multi-view silhouettes and RFID-based training, AAAI Fall 2008 Symposium on AI in Eldercare: New Solutions to Old Problems, Washington, DC, November 7 - 9, 2008b.

Pereira L.M., Alferes, J.J., Aparicio, J.N. (1992). Contradiction removal semantics with explicit negation. In Proceedings of the Applied Logic Conference, 1992.

Pereira, L.M., Anh, H.T. (2009a). Intention Recognition via Causal Bayes Networks plus Plan Generation, in: L. Rocha et al. (eds.), Progress in Artificial Intelligence, Procs. 14th Portuguese International Conference on Artificial Intelligence (EPIA'09), Springer LNAI, October 2009.

Pereira, L.M., Anh, H.T. (2009b). Elder care via intention recognition and evolution prospection, in: S. Abreu, D. Seipel (eds.), Procs. 18th International Conference on Applications of Declarative Programming and Knowledge Management (INAP'09), Évora, Portugal, November 2009.

Pereira, L.M., Saptawijaya, A. (2009). Modelling Morality with Prospective Logic, in International Journal of Reasoning-based Intelligent Systems (IJRIS), 1(3/4): 209-221.

Philipose, M., Fishkin, K.P., Perkowitz, M., Patterson, D.J., Hahnel, D., Fox, D., Kautz, H. (2005). Inferring ADLs from interactions with objects. *IEEE Pervasive Computing*, 2005.

Peirce, C. S. (1958). *The collected papers of Charles Sanders Peirce*. Vols. VII-VIII, Edited by AW Burks. Cambridge, MA: Harvard, 1958.

Quaresma, P., Lopes, J.G. (1995). Unified Logic Programming Approach to the Abduction of Plans and Intentions in Information-Seeking Dialogues. *Journal of Logic Programming*, Elsevier Science Inc., 1995, 103- 119.

Rao, M., Georgeff, P. (1995). BDI-agents: from theory to practice. In the Proceedings of the First International Conference on Multi-agent Systems (ICMAS'95), 1995.

Reiter, R. (2001). *Knowledge in action: logical foundations for specifying and implementing dynamical systems*. MIT Press, 2001.

Roy, P., Bouchard B., Bouzouane A, Giroux S. (2007). A hybrid plan recognition model for Alzheimer's patients: interleaved-erroneous dilemma. IEEE/WIC/ACM International Conference on Intelligent Agent Technology, 2007, 131- 137.

Sanchez, D., Tentori, M., Favela, J. (2008). Activity recognition for the smart hospital. *IEEE Intelligent Systems*, March/April 2008, 50-57.

Scmidt, C., Sridharan, N., Goodson, J. (1978). The plan recognition problem: an intersection of psychology and artificial intelligence. *Artificial Intelligence*, Vol. 11, 1978, 45-83.

Sindlar, M.P., Dastani, M.M., Dignum, F., Meyer, J.-J.Ch. (2008). Mental state abduction of BDI-based agents. Declarative Agent Languages and Technologies (DALT), Estoril, Portugal 2008.

Sukthankar, G., Sycara, K. (2008). Robust and efficient plan recognition for dynamic multi-agent teams (Short Paper). In Proceedings of 7th International Conference on Autonomous Agents and Multi-agent Systems (AAMAS 2008), International Foundation for Autonomous Agents and Multi-agent Systems, May 2008.

Sukthankar, G., Sycara, K. (2008). Hypothesis Pruning and Ranking for Large Plan Recognition Problems. In Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence (AAAI-08), June 2008.

Suzić, R., Svenson, P. (2006). Capabilities-based plan recognition. In Proceedings of the 9th International Conference on Information Fusion, Italy, July 2006.

Wilensky, R. (1983). *Planning and understanding*. Addison Wesley, Reading, MA, 1983.

Some definitions:

1. Intention recognition: the task of recognizing the intentions of an agent by analyzing some or all of their actions and/or analyzing the changes in the state (environment) resulting from their actions.
2. Keyhole intention recognition: Intention recognition in cases where the agent which is being observed either does not intend for his intentions to be identified, or does not care; he is focused on his own activities, which may provide only partial observability to the observing agent.
3. Intended intention recognition: Intention recognition in cases where the agent which is being observed wants his intentions to be identified and intentionally gives signals to be sensed by the observing agent.
4. Plan recognition: the task of recognizing not just the intention but also the plan (i.e. the sequence of actions, including future actions) the observed agent is following in order to achieve his intention.
5. Abductive reasoning: a form of defeasible reasoning allowing to draw hypothesis to explain some evidence or observations.
6. The event calculus: a causal theory of events, times and time-dependent properties formalized in Horn clause logic augmented with some form of negation.
7. The situation calculus: a causal theory formalized in classical logic specifying how actions change situations.