Plan Synthesis for Probabilistic Activity Recognition

Frank Krüger, Kristina Yordanova, Albert Hein and Thomas Kirste
Mobile Multimedia Information Systems Group, University of Rostock, Rostock, Germany

Keywords: Plan Synthesis, Activity Recognition, Probabilistic Reasoning, Causal Models.

Abstract: We analyze the applicability of model-based approaches to the task of inferring activities in smart environments. We introduce a symbolic approach to representing human behavior that enables the use of prior knowledge on the causality of human action and outline its probabilistic semantics. Based on an experimental analysis of a real world scenario from the smart meeting room domain, we show that such a symbolic approach allows to build reusable behavior models that compete with data-driven models at the performance level and that are able to track human behavior across a wide range of scenarios.

1 INTRODUCTION

Smart environments aim at assisting everyday activities by providing automatized responses to specific situations. Everyday activities can often be regarded as goal driven: they are performed in order to achieve a specific objective (e.g., prepare a meal, conduct a meeting). If a smart environment is able to estimate the objective – the goal state – that is the underlying cause for the activity, it can assist in achieving the desired goal state. We use the term *activity recognition* (AR) for the process of estimating a person's action sequence. One option for AR is to use plan recognition which analyzes whether observed activities can be interpreted as prefix of a plan that is known to lead to the goal in question.

Sensor observations are noisy and ambiguous, requiring probabilistic plan recognition that uses inference techniques such as Bayesian filtering. Works such as (Sadilek and Kautz, 2012) and (Hiatt et al., 2011) show this approach to be viable. The parameters of probabilistic models are in general inferred from training data. In smart environment settings, the acquisition of training data is expensive: it requires real-time observation of human behavior in natural environments. In addition it is difficult to reuse training data across different instances of the same domain: location sensor data from a meeting room at company *A* will not fit the spatial structure of a meeting room at university *B*.

Plan recognition is based on explicit symbolic representations of action sequences. These "plans" allow to capture existing prior knowledge on the structure of human behavior and its environment dependencies.

Plan recognition requires a library of plans to be available. Setting up such a plan library can be a tedious task (Roy et al., 2011), as there may be numerous action sequences achieving the same goal. However, works such as (Ramirez and Geffner, 2011) and (Hein et al., 2009) show that plan libraries can be automatically synthesized by employing planning technology. Here, just the set of available actions has to be provided. These actions are represented by pairs of first order formulae, *preconditions* and *effects*, describing the causal semantics of action execution.

From the viewpoint of AR for smart environments, a plan synthesis approach is interesting for three reasons: (1) it allows to use prior knowledge on the causal structure of human actionsto reduce the need for training data; (2) it provides a convenient means for determining the goal state that the acting person tries to achieve, enabling further state-based deliberation of assistive actions; (3) it allows for reusable models. A model definition is applicable across scenarios as long as their initial state and their goal state are representable in the model's state space.

However, for the plan synthesis approach to become a viable alternative to established data driven approaches, it (i) needs to show that it is able to compete with respect to recognition performance, and it (ii) needs to show that the effort of translating prior knowledge on human behavior into symbolic causal models is worthwhile by proving that such models indeed can be reused across different scenarios. In this paper we present an experimental study on whether a plan synthesis approach to AR for smart environments applications is able to fulfill this requirement.

The further structure of this paper is as follows:

Krüger F., Yordanova K., Hein A. and Kirste T..
Plan Synthesis for Probabilistic Activity Recognition.
DOI: 10.5220/0004256002830288
In *Proceedings of the 5th International Conference on Agents and Artificial Intelligence* (ICAART-2013), pages 283-288
ISBN: 978-989-8565-39-6

Sec. 2 discusses the current modeling concepts of interest for synthesis-based AR. In Sec. 3, we outline an enhanced modeling approach.The experimental setting for evaluating the viability of this approach is described in Sec. 4. Quantitative results are presented in Sec. 5 and in Sec. 6 we present our final conclusions based on the experiment.

2 PLAN SYNTHESIS

Activity recognition based on plan synthesis uses a symbolic representation of available actions from which possible plans (action sequences) are generated mechanically. This concept complements proposals for AR, where plans and libraries of plans are explicitly provided by human experts. An example for the latter approach is (Roy et al., 2011), relying on manually created ontology-based plan libraries. These plans represent partially ordered sequences of actions that must be carried out in order to achieve a goal. Another such work is the Asbru plan representation language that can be used to write time-oriented, intention-based, skeletal plan-specifications for clinical guidelines (Kaiser and Miksch, 2004); Asbru is able to represent the different subgoals of a plan as a hierarchy of plans with different temporal relationships for their execution. Library-based models are inherently unable to solve the problem of library completeness stated in the introduction – however, they may be important additions to a synthesis based approach, being able to convey knowledge on typical, preferred, or normative courses of activities.

A second option for arriving at a suitable set of plans is to mine action sequences from observations of human behavior. Such an approach is proposed by Storf et al. (Storf et al., 2009), using a rule-based plan representation. Starting from an initial rule set defined by a specialist, the system adds new rules to its library during a training phase by observing the user. Similarly, Okeyo et al. (Okeyo et al., 2011) use an ontology based approach to manually define an initial library of behaviors. Afterwards, observations of user activities are used to add behavior variations or remove obsolete behaviors. While providing interesting solutions to the problem of keeping plan libraries up-to-date, both concepts rely on initial manual definitions of behavior. So here too we think that a training based approach complements a synthesis based model rather than competing with it.

Regarding current approaches to plan synthesis, there are two routes being investigated: the use of cognitive models providing mechanisms for estimating human problem solving strategies based on a cognitive theory, and the use of decision theoretic approaches based on a situation calculus model.

As an example for employing cognitive models, (Hiatt et al., 2011) Hiatt et al. present an approach for plan recognition based on the cognitive architecture ACT-R, a sub-symbolic production system. It allows the description of the possible actions in terms of preconditions and effects, while the state of the world is modeled as information chunks that can be retrieved from the memory of the system. Every chunk has an activation level, and when a retrieval request is received matching more than one chunks, the one with the highest activation level is selected. Note that this action selection strategy is a heuristic model of human action selection. Additionally, by introducing different goals and initial world states, different behaviors can be followed.

A decision theoretic approach is investigated by Ramirez et al. (Ramirez and Geffner, 2011). The objective here is to identify the goal of an agent whose action library is defined in a PDDL-like notation. The idea is to consider the agent as solving a partially observable Markov decision problem (POMDP) for action selection: the policy that solves the POMDP is used to define the probability by which the agent selects a specific action; the higher an actions reward in a given state, the higher the probability that the agent will choose this action. This approach is quite similar to the model of (Hiatt et al., 2011): the main difference being the heuristics used for approximating human action selection policies.

We see that these approaches to plan synthesis basically use the same paradigm – precondition-effect rules– relying just on different heuristics with respect to modeling human action selection policies. Thus, the viability of a plan-synthesis approach to AR can be considered as proven by these investigations.

However, currently no investigations exist, that show whether the plan-synthesis approach can be used to define *reusable* models of activities. The case studies in the above works have concentrated on showing that symbolic models for *specific* scenarios can be established; it has not been shown that such models can be successfully *reused* across different scenarios belonging to the same application domain.

In addition, the focus has been on single user scenarios (no interleaved actions by multiple independently acting entities) that did not need to take action durations into account. Furthermore, problems of only limited state space size had been considered. So there is the question if the plan synthesis approach is able to measure up to the state space complexity of realistic scenarios, where multiple persons interact in time and space. In the next section, we outline an extended precondition-effect model for plan-synthesis based AR that aims at integrating these requirements.

3 EXTENDED PLAN SYNTHESIS

This section introduces the core aspects of the experimental system we have targeted in our experimental evaluations. We do not aim at a mathematical rigorous introduction of the modeling features of this system, but focus on providing an understanding of the feature set that has been available for evaluation.

Consider a state space X and an observation space Y. Let $y_{1:t}$ be a sequence of observations where $y_i \in Y$ for $i \in 1..t$. The objective of Bayesian filtering is to compute the posterior distribution $p(x_t \,|\, y_{1:t})$ that gives the probability of being in a state $x_t \in X$ after having seen the observations $y_{1:t}$. This computation is based on three distributions defining the filtering task: the observation model $p(y\,|\,x)$ that describes the probability of observing an event given a certain state; the transition model $p(x_{k+1}\,|\,x_k)$ that describes the probability of ending up in state x_{k+1} when the current state is x_k; and the prior distribution $p(x)$, which describes how the states probability is distributed. If a synthesis-based activity recognition system is able to provide these three distributions, it can readily be used for probabilistic activity recognition using the Bayesian filtering framework.

Such plan synthesis system is based on a set of actions A and a state space X, where actions $a \in A$ are defined by *preconditions* π_a and *effects* ε_a. An action a is said to be *applicable* in a state x if π_a is valid in x. If an action is applied to a state x, the action's effects ε_a are then valid in the new state $x' = a(x)$.

As we have a probabilistic system, we also want to model the probability that a person will select action a in state x_k, which is annotated as $p(a\,|\,x_k)$. There are different ways to define $p(a\,|\,x_k)$. One option is to use the goal distance $\delta(a(x_k))$. The cost of achieving the goal from the state reached by applying a to x_k is defined by

$$p(a\,|\,x_k) \propto \exp\left(-\lambda\,\delta(a(x_k))\right)$$

a similar strategy has been used by (Ramirez and Geffner, 2011). The λ-parameter models the impact of the goal distance on action selection. In contrast to this goal-driven heuristic, the ACT-R based approach by (Hiatt et al., 2011) focuses on situation-driven, opportunistic heuristics that consider aspects such as specificity and recency . As different heuristics seem reasonable, we believe that a model of action selection should provide for a *combination* of heuristics.In our experiments we have used a combination of three features: goal distance, saliency, and revisiting penalty (see below).

Based on these underlying considerations, we have set up a tool for synthesis-based activity recognition that uses a PDDL-like language for defining the set of actions for an inference domain and its state space. The tool compiles scenario definitions consisting of a domain file, a problem file, and an observation definition into executables that compute the posterior $p(x_t\,|\,y_{1:t})$ from observation data. Actions can be attributed with :agent, :duration, and :saliency declarations; domain definitions can be extended by :observation clauses.

An action's **:saliency** value $s(a)$ defines its saliency feature. It is used in the action selection model and allows to manually assign simple "priorities" to actions. An action's **:agent** value defines the execution thread for this action, which may depend on the action's parameters. If more than one agent is defined in a domain, the domain's effective action alphabet is given by the cartesian product of the actions of all agents. The **:duration** clause allows to declare the distribution of the action duration $p(\Delta_t\,|\,a)$. With respect to defining the observation model $p(y\,|\,x)$, domain definition and observation densities are linked via the **:observation** clause.

Translation of a complete model is rather straightforward: after parsing domain and problem file, the object sets for all defined types are computed, the actions are grounded relative to these object sets, preconditions and effects of actions are put into disjunctive normal form, and finally some simplifications are applied (a similar approach is used in (Bonet and Geffner, 2005)). More detailed information about the model semantics and the plan synthesis as well as the inference mechanism could be found in (Kirste and Krüger, 2012)

In order to evaluate our synthesis-based activity recognition system, in the next section we discuss the experimental scenario we used in this work.

4 EXPERIMENTAL SETUP

Objective of the experimental setup is to investigate whether reusable plan synthesis models can be created for AR in "real world" domains. We have used a "meeting" domain for this experiment: several persons meet in a room to hold presentations and discussions. Objective of the system is to recognize the sequence of activities that have taken place.

This setting involves several persons acting in parallel, requires the cooperation of persons, and contains temporal aspects. A reusable meeting model should be able to reconstruct meetings with different spatial and temporal topologies (different locations of seats, whiteboards; different sequence and durations of agenda items). Thus "meetings" pose basic challenges to the ability to handle interleaved durative activities by multiple agents. At the same time, meetings can be structured simple enough to allow for the construction of a "classical" training-based recognition system, using simple timed Markov models.

Based on this general setting, the following increasingly challenging hypotheses were to be tested:

• H_1 – The use of observation models based on the geometric room layout is viable for the meeting domain. The performance of those is competitive to models learned from training data.

• H_2 – Using prior knowledge in place of training data is valid. The accuracy of synthesis-based AR for a specific scenario is not significantly different from HMMs built from training data.

• H_3 – Using prior knowledge allows to build reusable models. It is possible to create a general model for detecting activities that is usable for *different* scenarios of an application domain, while being able to compete with training-based models.

• H_4 – Multi-agent modeling is viable. The modeling of independent agents does not decrease the performance of AR while at the same time provides additional information on the state of activities.

The hypotheses H_1 and H_2 are tested on a data set (data set D_1) containing sensor data from twenty sample meetings. These sample meetings have a simple structure: three persons A, B, and C meet in a room, there are three presentations scheduled (one for each person), and a final discussion. The presentation durations for A, B, and C are 70, 100, and 50 sec., respectively, the final discussion is scheduled for 30 sec. (4 min total duration.). The seats and presentation stages for the three persons have known locations. The twenty sample meetings contain different sequences through the agenda; the recorded data are the locations of A, B, and C tracked by a Ubisense UWB location tracking system.

For hypothesis H_1 we created four different observation models, two of which are created from the room layout (*circle* and *rcircle*) the other learned from training data (*gauss* and *mixture*).

To prove hypothesis H_2 we have trained a timed HMM on this meeting data corpus, and we have built a minimal plan-synthesis model (4 actions, 6 ground actions, 31 states). Both models were then applied to the meeting data set, using the different duration and location models.

For testing hypothesis H_3 an additional meeting with a significantly changed temporal and spatial structure has been recorded (data set D_2). Here, the presentation durations have been 16, 17, and 18 min. (52 min. total duration). Obviously, the HMM built for hypothesis (A) was not able to recognize the action sequence of this meeting. For the plan synthesis approach, we decided to build a multi-agent model that should be able to recognize the action sequences of both data sets.

For testing hypothesis H_4 the performance of the minimal plan synthesis model was compared to that of the multi-agent model using the dataset D_1. The reason for that is that the simple model describes the single agents behaviour as emerging from the team behaviour, while the multi-agent model describes the team behaviour as emerging from the interactions of the single agents.

In all, the following models have been build:

• HMM (Hidden Markov Model) – an HMM (with 17 states) with transition probabilities and observations estimated from the training data in D_1. This model provides a basic "sanity check" for the other models and serves as "baseline" for assessing the accuracy achieved by the symbolic models.

• SCM (Single-threaded Causal Model) – a single-threaded precondition-effect model of meeting activities, where each model action represents a compound action of all team members, as for the HMM. (This model provides a basic "sanity check" for the applicability of causal modeling to the domain)

• MCM (Multi-threaded Causal Model) – a multi-threaded precondition-effect model, employing the full descriptive and inferential capabilities of the system outlined in Sec. 3

Note that all models are applied on each observation model and each timing mode.

5 RESULTS

The evaluation of the filtering performance is based on the D_1 and D_2 meeting data sets as described in Section 4. We compare the four observation models *circle*, *rcircle*, *gauss*, and *mixture* and the two filtering methods exact, where applicable, and approximate, with respect to the three models (HMM, SCM, and MCM). In the results below a label like "$\mathrm{MCM}^{ap}(c)$" stands for the Multi-threaded Causal Model with circular observation model, gaussian duration model and approximate filtering. Similarly *rc* stands for rcircle observation model, *g* for gaussian observation model, and *mix* for mixture. The abbreviation *ex* denotes that exact filtering was applied. For all configurations the forward filtering performance was determined.

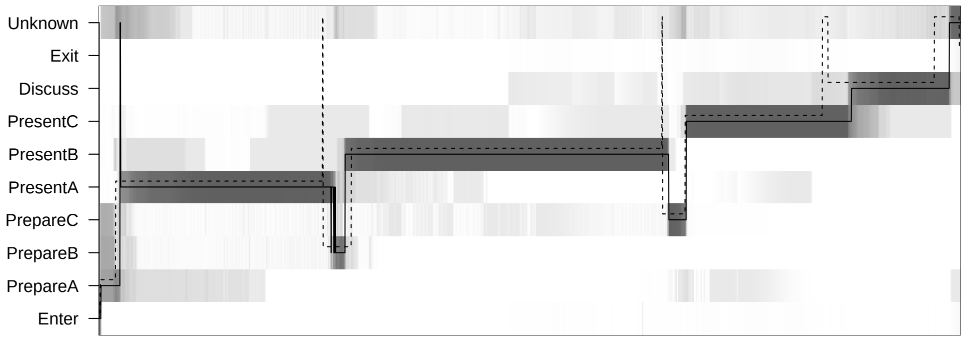

Figure 1: Monte-Carlo approximation of the posterior distribution ($p(x_t|y_{1:t})$) using 10000 particles of 50 runs. The dotted line illustrates the ground truth whereas the solid line shows the estimated meeting action sequence.

Table 1: Performance for data sets D_1 and D_2.

	Model	Team Level				Agent Level			
		$\widetilde{acc}$	$\overline{acc}$	$\widetilde{prec}$	$\overline{prec}$	$\widetilde{acc}$	$\overline{acc}$	$\widetilde{prec}$	$\overline{prec}$
Dataset D_1									
1	$HMM^{ex}(c)$	**.8964**	.8943	.8892	.8814	-	-	-	-
2	$HMM^{ex}(rc)$	**.8887**	.8852	.8612	.8640	-	-	-	-
3	$HMM^{ex}(g)$	**.8994**	.8981	.8893	.8858	-	-	-	-
4	$HMM^{ex}(mix)$	**.9225**	.9053	.9236	.9206	-	-	-	-
5	$SCM^{ap}(c)$	**.8969**	.8939	.9067	.9039	-	-	-	-
6	$SCM^{ap}(rc)$	**.8871**	.8855	.9014	.8976	-	-	-	-
7	$SCM^{ap}(g)$	**.9093**	.8987	.9188	.9085	-	-	-	-
8	$MCM^{ap}(c)$	**.9115**	.9020	.9019	.8940	**.9435**	.9400	.9505	.9501
9	$MCM^{ap}(rc)$	**.8993**	.8972	.8992	.8838	**.9443**	.9453	.9477	.9459
10	$MCM^{ap}(g)$	**.9091**	.9039	.9076	.9016	**.9385**	.9376	.9498	.9473
Dataset D_2									
11	$MCM^{ap}(c)$	**.9893**		.9874		**.9867**	.9888	.9883	.9904
12	$MCM^{ap}(rc)$	**.9861**		.9884		**.9853**	.9881	.9890	.9905
13	$MCM^{ap}(g)$	**.9767**		.9725		**.9841**	.9840	.9822	.9819

Table 2: Performance comparison of different models.

	Model 1	Model 2	$\widetilde{acc1}$	$\widetilde{acc2}$	$\Delta\widetilde{acc}$	W	p	$\widetilde{\Delta acc}$
1	$HMM^{ex}(c)$	$SCM^{ap}(c)$	.8964	.8969	**-.0005**	107	**.6435**	.0015
2	$HMM^{ex}(g)$	$SCM^{ap}(g)$	.8994	.9093	**-.0099**	74	**.2549**	-.0034
3	$HMM^{ex}(rc)$	$SCM^{ap}(rc)$	.8887	.8871	**.0015**	106	**.9851**	-.0000
4	$HMM^{ex}(c)$	$MCM^{ap}(c)$	.8964	.9115	**-.0151**	58	**.0826**	-.0139
5	$HMM^{ex}(r)$	$MCM^{ap}(rc)$	.8887	.8993	**-.0107**	13	**.0010**	-.0125
6	$HMM^{ex}(g)$	$MCM^{ap}(g)$	.8994	.9091	**-.0097**	54	**.0594**	-.0075
7	$SCM^{ap}(c)$	$MCM^{ap}(c)$	.8969	.9115	**-.0146**	66	**.1506**	-.0141
8	$SCM^{ap}(rc)$	$MCM^{ap}(rc)$	.8871	.8993	**-.0122**	30	**.0054**	-.0135
9	$SCM^{ap}(g)$	$MCM^{ap}(g)$	.9093	.9091	**.0002**	61	**.1044**	-.0062
10	$HMM^{ex}(mix)$	$MCM^{ap}(c)$	.9225	.9115	**.0110**	124	**.4898**	.0112
11	$HMM^{ex}(mix)$	$MCM^{ap}(rc)$	.9225	.8993	**.0231**	132	**.3225**	.0196
12	$HMM^{ex}(mix)$	$MCM^{ap}(g)$	.9225	.9091	**.0134**	131	**.3411**	.0102

For each of the 20 sample meetings D_1 and meeting D_2, respectively exact and approximate inference runs were performed. In order to increase robustness approximate results were computed by majority vote from 50 independent particle filter runs using 10,000 particles. Figure 1 illustrates the posterior distribution of all 50 runs and the estimated action sequence.

Table 1 summarizes the results of the different experiments with respect to accuracy and precision. For the runs on D_1 (Table 1), $\tilde{x}$ gives the median value of the 20 results for the files in D_1 and $\bar{x}$ the mean. For the D_2 run the single value is reported. The columns labelled "Agent Level" give the performance results for individual agents using the multi-threaded model.

Pairwise comparisons of the accuracy of different model configurations over 20 meetings together with the Wilcoxon Matched-Pairs Signed-Ranks Test statistics (W) and p-values are given in Table 2. The p-value indicates whether the accuracies from both models arise from the same distribution or differ significantly (p-value < 0.05). The median of the differences as well as the difference of the median accuracies illustrate performance differences.

In order to prove hypothesis H_2 we compare the performance of the different runs of the HMM and the SCM. The comparison (row 1–3 in Table 2) show that the recognition accuracy of both models does not differ significantly and is about 89% in all cases. The difference of the accuracies is about 0.1%. Modeling multiple agents can increase the accuracy furthermore (rows 4–6 of Table 2).

Rows 7–9 show that even the use of multiple agents, which increases the state complexity at runtime, does not decrease the recognition rate (Hypothesis H_4). Rows 11–13 and 8–10 in Table 1 prove that a reusable model for the meeting domain can be created, without decreasing the performance of the recognition (Hypothesis H_3).

Hypothesis H_1, which allows the usage of observation models based on the room layout instead of models learned from training data, can be shown to be true. Rows 10–12 of Table 2 show a decrease of about 1% for the use of causal models together with observation models based on the room layout instead of handcrafted HMM's with observation models learned from data.

The evaluation of the results show that the hypotheses (H_1–H_4) can be accepted, which means that the approach plan synthesis from causal models by means of precondition and effect rules is competitive with state of the art methods for AR.

6 DISCUSSION AND OUTLOOK

To summarize the experiments: the results showed that using geometric observation models based on the room topology instead of trained observation models, does not hinder the AR and is competitive to the trained models. Even more, in the case of the generated models SCM and MCM combined with the geometric models, they outperformed the same causal models combined with trained observation models. This was observed in the experiments performed on both datasets, thus supporting hypothesis H_1.

Additionally, both scenario-specific and generic synthesis-based models are able to achieve a performance at a similar level as the trained model, provided that suitable observation and duration models are used. Thus we consider hypothesis H_2 as supported. Furthermore, the generic multi-agent model is able to correctly recognize not only dataset D_1 but also dataset D_2 and at a high performance level (98.9% accuracy on the team level and 98.6% accuracy on the agent level), providing evidence for hypotheses H_3. Finally, the data show that the use of multi-agent modeling does not decrease performance, but indeed is able to increase it, supporting hypotheses H_4. The multi-agent model allows to temporally decouple state changes of agents, thus providing a fine grained state estimation at the agent level. The data show a solid performance of 94% (D_1) resp. 98% (D_2) for this agent-level state estimation.

We thus conclude with the statement that plan-synthesis approaches indeed allow the construction of reusable models. Furthermore, our experiences show approximate inference is feasible, enabling realistic problem sizes and multi-agent interactions – however, a successful application of these techniques beyond short term activities requires the ability to support appropriate duration models. Finally, the usage of training free observation models allows competitive to the trained models inference performance.

As next step, we intend to show that the modeling approach proposed here is also applicable to other domains, such as recognizing activities of daily living. While we believe this to cause no fundamental problem, we expect to gather new experiences on modeling methodology. As the approach uses the Bayesian inference paradigm, adding additional inference tasks from this paradigm, such as prediction and parameter estimation, are, at least in principle, tasks for which solutions exist. (We look forward to results of learning the weights of action selection features from data, as this would provide empirical data on the weight of heuristics in human action selection strategies.)

REFERENCES

Bonet, B. and Geffner, H. (2005). mGPT: A Probabilistic Planner Based on Heuristic Search. *J. Artif. Intell. Res. (JAIR)*, 24:933–944.

Hein, A., Burghardt, C., Giersich, M., and Kirste, T. (2009). Model-based Inference Techniques for detecting High-Level Team Intentions. In Gottfried, B. and Aghajan, H., editors, *Behaviour Monitoring and Interpretation*, volume 3. IOS Press.

Hiatt, L. M., Harrison, A. M., and Trafton, J. G. (2011). Accommodating Human Variability in Human-Robot Teams through Theory of Mind. In *Proc 22nd Intl J Conf on Artif. Intell. (IJCAI)*.

Kaiser, K. and Miksch, S. (2004). Treating Temporal Information in Plan and Process Modeling. Technical Report Asgaard-TR-2004-1, Institute of Software Technology and Interactive Systems, Vienna University of Technology, Vienna.

Kirste, T. and Krüger, F. (2012). CCBM-A tool for activity recognition using Computational Causal Behavior Models. Technical Report CS-01-12, Institut für Informatik, Universität Rostock, Rostock, Germany. ISSN 0944-5900.

Okeyo, G., Chen, L., Wang, H., and Sterritt, R. (2011). Ontology-Based Learning Framework for Activity Assistance in an Adaptive Smart Home. In Chen, L. et al., editors, *Activity Recognition in Pervasive Intelligent Environments*, volume 4. Atlantis Press.

Ramirez, M. and Geffner, H. (2011). Goal Recognition over POMDPs: Inferring the Intention of a POMDP Agent. In *Proc 22nd Intl J Conf on Artif. Intell. (IJCAI)*.

Roy, P. C. et al. (2011). A Possibilistic Approach for Activity Recognition in Smart Homes for Cognitive Assistance to Alzheimer's Patients. In Chen, L. et al., editors, *Activity Recognition in Pervasive Intelligent Environments*, volume 4. Atlantis Press.

Sadilek, A. and Kautz, H. (2012). Location-Based Reasoning about Complex Multi-Agent Behavior. *J. Artif. Intell. Res. (JAIR)*, 43:87–133.

Storf, H., Becker, M., and Riedl, M. (2009). Rule-based activity recognition framework: Challenges, technique and learning. In *Proc. 3rd Intl. Conf. on Pervasive Computing Technologies for Healthcare*.