Intention Recognition for Partial-Order Plans Using Dynamic Bayesian Networks

Peter Krauthausen and **Uwe D. Hanebeck**
Intelligent Sensor-Actuator-Systems Laboratory (ISAS),
Institute for Anthropomatics,
Universität Karlsruhe (TH), Germany.
Krauthausen@ira.uka.de, Uwe.Hanebeck@ieee.org

Abstract – *In this paper, a novel probabilistic approach to intention recognition for partial-order plans is proposed. The key idea is to exploit independences between subplans to substantially reduce the state space sizes in the compiled Dynamic Bayesian Networks. This makes inference more efficient. The main contributions are the computationally exploitable definition of subplan structures, the introduction of a novel Layered Intention Model and a Dynamic Bayesian Network representation with an inference mechanism that exploits consecutive and concurrent subplans' independences. The presented approach reduces the state space to the order of the most complex subplan and requires only minor changes in the standard inference mechanism. The practicability of this approach is demonstrated by recognizing the process of shelf-assembly.*

Keywords: *Probabilistic Plan Recognition, Intention Recognition, Dynamic Bayesian Networks, Human-Robot Cooperation.*

1 Introduction

Recognizing the intention driving an agent's actions based on noisy and partial observations is of interest to many fields of research. Possible applications include traffic monitoring [1], visual surveillance [2], user assistance in office software [3], mailing software [4], computer games [5] or human-robot-cooperation [6]. A lot more applications can be found in [7].

If the intentions span several levels of detail and several consecutive actions, the notion of intention is used synonymously to a plan. For this reason, the intention recognition problem is most often termed as plan recognition. Initially defined in [8], the plan recognition problem consists of the recognition of an agent's goal as well as the plan by which it attempts to achieve this goal based on observing the agent's actions. When only the final goal of the user is of interest, the plan recognition problem reduces to goal recognition [9]. The plan is the sequence of actions chosen and performed by the agent. Thus, inferring an agent's plan can be understood as the problem of matching the observed actions to a plan contained in a plan library. The hardness of this problem stems from the fact that agent behavior may be governed by multiple interleaved plans at a particular time, the large number of possible plans and the ambiguity of the *action-to-plan* mapping. In addition, the recognition is based on noisy and partial observations. Due to all these reasons, a plan recognition algorithm must be robust to sensor noise and failure as well as capable of maintaining multiple plan hypotheses simultaneously. The recognition should allow for plan revisions due to the limitations of the measurement system or reconsiderations by the agent. Because of these requirements, probabilistic approaches capable of exploiting negative evidence and therefore limiting the hypothesis space are especially applicable to this problem.

A considerable amount of research on plan recognition has been performed since [8]. The first distinction to make is keyhole vs. intended plan recognition. In the field of natural language processing, one assumes that the speaker's utterances are *intended* at conveying information to the listener and the listener may communicate with the speaker for clarification. In contrast, plan recognition performed by office tools [3] or in computer games [5] is based on the non-interactive observation of the agent's actions. The recognition works as if seeing the user through a *keyhole*: observing the mouse and keyboard actions only. Furthermore, symbolic and probabilistic plan recognition may be distinguished. The first plan recognition approaches used first-order predicate calculus predominantly [10]. Since [11], plan recognizers were developed that associate probabilities with plan hypotheses. Since then plans are inferred based on their possibility and likelihood.

For the rest of this paper only keyhole recognition of the entire plan will be considered. Furthermore, partial-order plans (POPs) are considered, which are plans specified by a set of subtasks and precendence relations on them. POPs only define some temporal relations, allowing all other tasks to be executed in an arbitrary order. The paper is structured as follows. First, related work will be discussed and an exact definition of POPs will be given. Then the notion of *subplans* and a novel *Layered Intention Model* are introduced. On this basis, the decomposition of the state space according to concurrent plans and the precedence structure between subplans will be shown and exploited to make inference with this model more efficient. Finally, the model and the decompositions will be demonstrated with the recognition of a shelf-assembly.

2 Related Work

The symbolic-probabilistic approaches that most resemble our approach are [12] and [13]. The central idea of [12] is the maintenance of a *pending set* of actions during the recognition process whose preceding actions completed and which are possible successive actions. The set is updated based on observed actions and their compliance with the plan library and the preceding pending set. Poole's *Probabilistic Horn Abduction* is used to maintain this set allowing for an explicit modeling of uncertainties. The pending set enables efficient handling of POPs as it keeps record of finished actions disregarding their order. A drawback of this approach is the lack of an explicit account for the world state and the effects of the observed agent's actions. [12].

In [13] a symbolic plan recognizer that is robust to missing observations and can handle interleaved plan executions based on a plan library was proposed. Robustness is achieved by applying a sophisticated *Feature Decision Tree* for mapping observations to plan steps, which, e.g., is augmented with missing value branches in order to compensate for possibly missing observations. Time consistency of the mapped actions is supported by *marking* already executed plans. Only if all preceding plan steps have completed, a consecutive plan step is assumed to be observable. In addition, the plan graph is extended with *duration models* and *memory flags* storing the last observation time up to a *maximum interrupt time* and thus allows resuming interrupted plans. Even though this approach is capable of the recognition of POPs, it does not support a consistent treatment of the uncertainties inherent in the problem.

Regarding probabilistic approaches to plan recognition, [1] and [2] are most similar to our approach. In [1, 14], *Probabilistic State Dependant Grammars* are used to generate *parse trees* offline that correspond to a plan library. The parse trees including the production probabilities are compiled into a Dynamic Bayesian Network (DBN) [15, 16] and standard inference algorithms are applied to infer the pursued plan. Even though the use of a *compact belief state*, which neglects intermediary production probabilities, was proposed [1], the PSDG approach does not scale well for POPs. In this approach, POPs need to be represented by a full enumeration of all derivable totally ordered plans[1], making inference intractable.

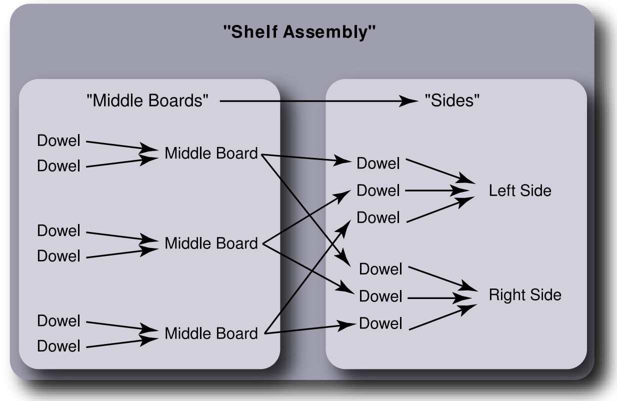


Figure 1: TPG for the shelf assembly.

In [2], plan recognition is conceived as inference based on an *Abstract Hidden Markov Memory Model* (AHMEM). This model consists of layered Markov policies encoding a hierarchy of plans as in the models above. In the lowest hierarchy level, an abstract policy is defined in terms of transition probabilities mapping the current world state - as estimated from the observations - to the preceding world state for *all* primitive actions. In higher levels, policies are based on lower-level policies instead of these primitive actions. The resulting AHMEM is compiled into a DBN. Like [13, 1], the AHMEM stores the execution status of the current level or action. In addition, the use of approximate inference is proposed and a formulation is given for the use with a Rao-Blackwellized Particle Filter (RBPF) based on contextual independence within the plan hierarchy given the execution status. Yet, in [2], no potential structure of the plans at hand is exploited.

From the approaches above, only [13, 12, 2] can handle POPs efficiently. A consistent processing of the uncertainty inherent in the problem is performed in [1, 2] only. Thus, only the AHMEM approach can recognize POPs efficiently and process uncertain information consistently. In the following section POPs will be described, subplans as useful decompositions of POPs will be introduced, and a novel intention model that allows for the recognition of multiple interleaving plans with a consistent treatment of the uncertainty entailed in the problem will be presented. Additionally, the problem of an exponentially growing state space is addressed by exploiting subplan independencies and concurrencies.

[1] Cf. [14], Section 7.1.1.

3 Partial-Order Plans

It is assumed that a library of POPs is given in the form of *Task Precedence Graphs* (TPG). Plans that do not specify the exact execution order are useful in applications, where the number of totally ordered plans is substantial. Generalizing the totally ordered plans into partial-order plans reduces the computational cost drastically. A prominent example as to where complex plans of this kind arise are *Programming by Demonstration* procedures, e.g., for humanoid robots, where they allow for motion synthesis as well as action and plan recognition [17]. In general, the partial-order plans arise in a lot of problems from as simple as cake making to complicated tasks as route planning [18].

Definition 1 (Task Precedence Graph [17]) *A TPG is a graph $P = (T, R)$ and T a set of subtasks and $R \subset T \times T$ is a set of precedence relations. The relation $(\mathbf{a}, \mathbf{b}) \in R$, $\mathbf{a}, \mathbf{b} \in T$ implies that subtask $\mathbf{a}$ has to be completed before $\mathbf{b}$ can start.*

A TPG defines the precedence order in which subtasks need to be performed. The term subtask refers to a primitive or composite actions. Fig. 1 shows a TPG for a shelf-assembly. The nodes correspond to subtasks and directed edges correspond to precedences, e.g., in Fig. 1 dowels need to be inserted prior to the boards. Due to the partial order, parts of the plan may interleave or alternate. For small domains, a conversion of a POP into a set of total-ordered plans might be feasible and allows for the application of standard probabilistic graphical model inference. Nevertheless, for non-trivial tasks, this approach becomes intractable due to the explosion of possible subtask combinations. Besides this problem, enumerating total-order plans for evolving domains *a priori* might be impossible. A more realistic approach includes storing the facts necessary for a future evaluation of the preconditions of subtasks within the graphical model. Aside from allowing for the recognition of interleaved plans, this approach allows taking up interrupted plans. Plan continuation is possible because the relevant facts at the interrupt time are stored. Yet, several new challenges arise. First, the type of preconditions needs to be specified. Second, storing a possibly large number of facts will lead to an exponential increase in the state space and thus leads to intractable inference, too.

The first purely probabilistic approach capable of POP recognition is [2]. As mentioned earlier the recognition of policies is investigated in [2], thus the world and internal state are included in the model. The scalability issue is addressed by the application of an RBPF. Yet, the precedence structure of the conditional probabilities and the independence relations between concurrent plans is not exploited, even though it allows for substantial computation reductions.

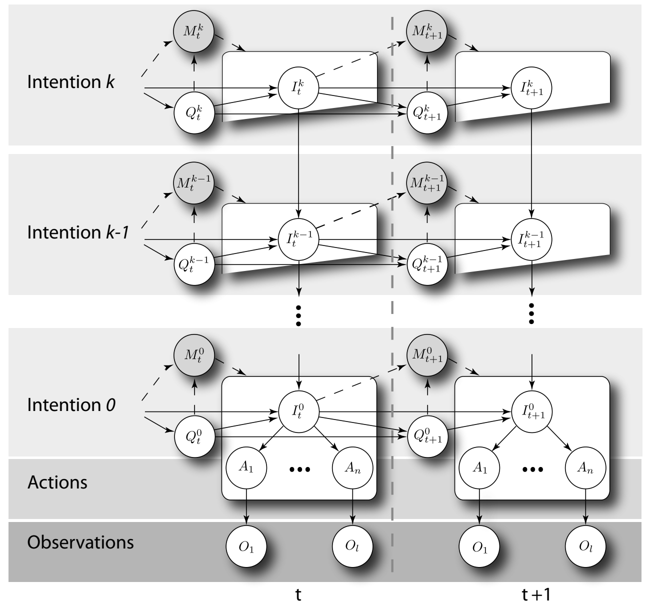


Figure 2: Layered Intention Model (LIM) for the consecutive time slices t and $t+1$.

Definition 2 (TPG Subplan) *A graph $P' = (T', R')$ is a subgraph/subplan of $P = (T, R)$ iff all of the following conditions hold*

1. $T' \subset T$, $R' \subset R$,
2. $\forall\, a \in T \,, b \in \overline{T} \,:\, \nexists\, (a, b) \in R'$,
3. $\forall\, a \in T \,, b \in T' \,:\, \nexists\, (a, b) \in \overline{R}$,
4. $\forall\, (a, b) \in \overline{R}$, $a \in T'$, $b \in \overline{T} \,:\, \forall\, c \in \overline{T}$, $\nexists\, (c, b) \in R$,
5. $\forall\, (a, b) \in R'$, $a \in \overline{T}$, $b \in T' : \forall\, c \in \overline{T}$, $\nexists\, (b, c) \in R$.

Precedence relations $R_i \subset T_i \times T_i$ are given for sets of subtasks T_i and $T \setminus T' = \overline{T}$ as well as $R \setminus R' = \overline{R}$.

The first three conditions state that the subplan is part of the plan, no relations in the subgraph depending on successive non-subgraph subtasks as preconditions exist, and no non-subplan ordering of the subtasks exists. The other two conditions guarantee that the subplan can be understood as an atomic subtask, which can not be reordered as all preceding/succeeding subtasks are related to subtasks entailed in the subplan or subtasks that precede/succeed. For the rest of this paper, subplans identical to single subtasks will be ignored. By using this assumption, the TPG in Fig. 1 is composed of the subplans *Middle Boards* and *Sides*.

4 Layered Intention Model

Similar to related approaches, a layered model of the agent behavior is proposed. The Layered Intention Model (LIM) is motivated by the idea that an agent's higher-level intentions drive lower-level intentions down to the heuristic action-level. In our shelf-assembly example these layers would be: *assemble main board*, *insert dowel* with the actions *grasp a dowel*, and *approach the board*. These actions in turn would produce observations like *approaching board* or *grasping dowel*.

The proposed layer structure is very generic as it might be extended by an increasing number of layers. In the example, the above layers may be part of layers *refurbish room* or *moving from city A to city B*. In comparison to [2], the policy nodes correspond to intention nodes $\underline{\mathbf{i}}_t^k$, which represent intention-state combinations. Yet, the introduced causal forward model differs in that the world and internal states are aggregated in one variable $\underline{\mathbf{q}}_t^k$. This distinction reflects that most higher-level states are abstract states not directly accessible, in contrast to [2]. The purpose of the mode variables $\underline{\mathbf{m}}_t^k$ will be explained later on, but note that a box represents a model selection function. For all random variables in Fig. 2, the lower index denotes the time index t and the upper index k gives the intention layer membership. Observation nodes $\underline{\mathbf{o}}_i$ and actions nodes $\underline{\mathbf{a}}_i$ are not connected between time slices.

For the sake of brevity, the number of possible observations per action node was set to one. In real-world applications, multiple observations need to be mapped to one action. Furthermore, it is assumed that actions are independent of each other. In the rest of this section, an analytic model description will be given and exposed how the precedence relations between and within subplans can be exploited for efficient inference.

4.1 Generative Model

The model given in Fig. 2 is a causal forward model of an intention hierarchy. The generative description is given in the form of the following equations.

$$\underline{\mathbf{i}}_t^l = \underline{f}^{(l,t)}(\underline{\mathbf{i}}_{t-1}^l, \underline{\mathbf{i}}_t^{l+1}, \underline{\mathbf{q}}_t^l, \underline{\mathbf{m}}_t^l) \tag{1}$$

$$\underline{\mathbf{q}}_t^l = \underline{f}_q^{(l,t)}(\underline{\mathbf{i}}_{t-1}^l, \underline{\mathbf{q}}_{t-1}^l) \tag{2}$$

$$\underline{\mathbf{a}}_j = \underline{g}^{(j)}(\underline{\mathbf{i}}_t^0) \tag{3}$$

$$\underline{\mathbf{o}}_i = \underline{h}^{(i,j)}(\underline{\mathbf{a}}_j) \tag{4}$$

The generative description of the dependence of intention $\underline{\mathbf{i}}_t^l$ on the higher-level intention $\underline{\mathbf{i}}_t^{l+1}$, the prior intention at level l as well as the state variable $\underline{\mathbf{q}}_t^l$ for this level is given in (1)[2]. The relations between the lowest intention and the action-level as well as between observations and action-level are given by (3) and (4), respectively. The first has to be understood as an abstract layer that consists of more sophisticated action *subtrees* that are omitted for simplification. The sensor models and the description of noise and accuracy are subsumed by (4). The effects of the actions on the (abstract) world state are described by (2). The missing relation for $\underline{\mathbf{m}}_t^l$ is a model switching function and will be introduced in Section 4.3. In Section 5.1, it will be shown how to instantiate such a generative model.

[2]Underlines denote vector-valued variables. Bold faced variables are random variables.

4.2 Exploiting Parallel Plans

Important substructures of any LIM (cf. Fig. 2) are the multiple inter-temporal connections between $\underline{\mathbf{i}}_{t-1}^l$, $\underline{\mathbf{q}}_t^l$, $\underline{\mathbf{m}}_t^l$ and $\underline{\mathbf{i}}_t^l$. These loops exist throughout all layers. As noted in [2], an exact calculation of the belief state is intractable for large k. In [2], this problem is addressed by using an RBPF based on hierarchical independence of layers given the respective policy execution status.

Additionally, exploiting the independence of subplans of concurrent plans in the conditional probabilities in order to foster an efficient inference is proposed. For this purpose, the joint state description of each layer $f(\underline{\mathbf{i}}_t^l, \underline{\mathbf{q}}_t^l, \underline{\mathbf{m}}_t^l) = P(\underline{\mathbf{i}}_t^l, \underline{\mathbf{q}}_t^l, \underline{\mathbf{m}}_t^l)$ is decomposed. For this subsection, $\underline{\mathbf{m}}_t^l$ is subsumed by $\underline{\mathbf{q}}_t^l$. The resulting state $P(\underline{\mathbf{i}}_t^l, \underline{\mathbf{q}}_t^l)$ describes the intention probability by combinations of all attainable states and intentions in this layer. The decomposition is motivated by the fact that different plans or (concurrent) subtasks rely on subsets of $\underline{\mathbf{q}}_t^l$ only. Subtasks $\mathbf{a}$ and $\mathbf{b}$ are assumed to be given, thus the state decomposes to

$$\underline{\mathbf{i}}_t^l = \left[\underline{\mathbf{i}}_t^{(l,a)} \underline{\mathbf{i}}_t^{(l,b)}\right]^T , \qquad \underline{\mathbf{q}}_t^l = \left[\underline{\mathbf{q}}_t^{(l,a)} \underline{\mathbf{q}}_t^{(l,b)}\right]^T$$

and

$$\bar{\underline{\mathbf{i}}}_t^l = \left[\underline{\mathbf{i}}_t^{(l,a)} \times \underline{\mathbf{q}}_t^{(l,a)} \quad \underline{\mathbf{i}}_t^{(l,b)} \times \underline{\mathbf{q}}_t^{(l,b)}\right]^{\mathrm{T}} .$$

The resulting state vector is denoted by $\bar{\underline{\mathbf{i}}}_t^l$ and replaces $\underline{\mathbf{i}}_t^l$. This decomposition limits the exponential growth of the state vector to the cross product of the all actions and all states to the subplan actions and subplan states. Thus, this approach helps to let this layered intention model scale. Note that this decomposition is not limited to LIM and may, e.g., be applied to [2] too.

4.3 Exploiting Subplan Precedence

In the last section, independence relations between concurrent plans regarding their respective state spaces were exploited. In this section, exploiting the independence of successive subplans $\tilde{\mathbf{a}}$ and $\tilde{\mathbf{b}}$ will be considered. Thus, introducing a precendence relation $\tilde{R}$ on subplans $(\tilde{\mathbf{a}}, \tilde{\mathbf{b}}) \in \tilde{R}$ implies that subplan $\tilde{\mathbf{a}}$ has to be completed prior to the execution of subplan $\tilde{\mathbf{b}}$. Roughly speaking, the estimated completion of $\tilde{\mathbf{a}}$ fixes the state at the beginning of subplan $\mathbf{b}$. For large world states $Q_{\tilde{a}}$ and $Q_{\tilde{b}}$ of the respective plans, the following holds

$$|Q_{\tilde{a}}| + |Q_{\tilde{b}}| \ll |Q_{\tilde{a}}| \cdot |Q_{\tilde{b}}| . \tag{5}$$

Furthermore, new intentions might become possible and existing intentions might be obsolete. This relation may be exploited on the basis of the estimated subplan completion. In order to exploit this precedence, a variable has to store the completion status. The neglected variable $\underline{\mathbf{m}}_t^l$ in the LIM is meant to store the execution status of the ordered subplans within the l-th layer. From a signal processing standpoint, the $\underline{\mathbf{m}}_t^l$ may be interpreted as the system's mode. This model with a changing dependence structure may be interpreted as a *switching* model. The variable $\underline{\mathbf{m}}_t^l$ may be subsumed by augmenting the variable $\underline{\mathbf{i}}_t^l$ at the cost of a larger CPT (5). In order to avoid this additional computational cost, the use of a *multinet* model [19] is proposed. The attained values allow for the following definition

$$\underline{\mathbf{m}}_t^l = z_t^l(\underline{\mathbf{i}}_{t-1}^l, \underline{\mathbf{q}}_t^l)\,. \tag{6}$$

The random variable m_t^l is discrete-valued. By using the multinet approach, (1) can be reformulated as

$$\begin{aligned}\underline{\mathbf{i}}_t^l &= f^{(l,t)}(\underline{\mathbf{i}}_{t-1}^l, \underline{\mathbf{i}}_t^{l+1}, \underline{\mathbf{q}}_t^l, \underline{\mathbf{m}}_t^l)\\ &= f_m^{(l,t)}(\underline{\mathbf{i}}_{t-1}^l, \underline{\mathbf{i}}_t^{l+1}, \underline{\mathbf{q}}_t^l)\,. \end{aligned} \tag{7}$$

This means that the transition model and $\underline{\mathbf{i}}_t^l$ as well as $\underline{\mathbf{q}}_t^l$ are chosen depending on the distribution of $\underline{\mathbf{m}}_t^l$. For the sake of clarity, indexing the random variables will be omitted. Thus, the cardinality of $\underline{\mathbf{i}}_t^l$ and $\underline{\mathbf{q}}_t^l$ may vary largely. Yet, this model implies that a linear combination of models gives rise to the overall estimates for this layer. In a realistic scenario, negative evidence will help reduce the amount of models that need to be calculated in parallel. The practicability of this approach will be demonstrated in the next section.

4.4 DBN Representation and Inference

As implied in the generative model proposed in Sec. 4.1 actions, intentions, states, and modes may be continuous- or discrete-valued. The rest of this paper will be limited to purely discrete DBN. Yet, note that for a set of continous-valued variables, e.g., time, position, temperature, savings can be obtained similarly. As before, a library of TPGs is assumed to be given. Considering only one TPG for the moment, the first step is to determine the subplans contained in the TPG and their relations. In order to instantiate the LIM, one needs to obtain the actual intentions and states per layer from the subplans τ. The elementary subtasks are the actions τ_{a_j}. The level-0 intentions comprise of combinations of actions. Higher-level intentions $\tau_{\underline{\mathbf{i}}^l}$ correspond to subplans and subplans of subplans in the TPG. After determining the state space for $\tau_{\underline{\mathbf{i}}^l}$ the relevant (abstract) world states for this subplan $\tau_{\tilde{a}}$ need to be determined, e.g., the number of repetitions or the sequence of lower-level subtasks. Given the above defined variables for all TPGs, the joint variables comprise of a concatenation of state spaces. The joint variables will be the variables in our DBN. The multinet variables, which comprise of entries for the active and succeeding subplans per TPG will be concatenated too, but will not be appended to the DBNs as variables. So far, the variables of the DBN were determined. Thus, the state spaces have already been reduced as the *concatenation* of the states per subplan replaced a combination of all subplans' states with all other subplans' states. In order to complete the DBN, the missing conditional probability tables (CPT) need to be derived from the generic model in Sec. 4.1. This is a typcial problem in Bayesian inference [20, 16] and may vary largely with the problem at hand. Given the variables and the conditional probability tables, a DBN for each concurrent subplan combination was derived. The switch between the networks needs to be determined. The corresponding multinet variables are given for all combinations. The distribution of the multinet variables will be determined merely by the application of the multinet function $z(.)$ to the posterior estimates of the variables in (6). This function incorporates the domain knowledge, e.g., in the form of confidences and thresholds. Alg. 1 summarizes the compilation process.

Algorithm 1 DBN Compilation

Input: set of TPGs

for all TPG **do**
  determine subplans
  determine $\tau_{a_j}, \tau_{\underline{\mathbf{i}}^l}$ and $\tau_{\underline{\mathbf{q}}^l}$ for all layers
  determine $\tau_{\underline{\mathbf{m}}^l}$
end for
// concatenate TPGs' variables
DBN $\leftarrow a_j$, $\underline{\mathbf{i}}^l$, $\underline{\mathbf{q}}^l$, $\underline{\mathbf{m}}^l$ and CPTs for all layers

During run-time, inference in the active subplans' DBNs is performed. To this end, the inference algorithm of choice can be used. On the basis of the derived posterior probabilities, the posterior state of the multinet variable is determined, which then in turn reweights the posterior results of the subplans' DBNs. Depending on the value of the switching function, a DBN switch causes the abandonment of active and the introduction of consecutive subplan DBNs. The inference routine in the compiled DBN is described in Alg. 2.

Algorithm 2 DBN Inference

Input: set of active DBN

repeat
  observations $\rightarrow$ active DBN
  // inference mechanism of choice
  calculate posterior probabilities of DBN variables
  calculate posterior value $\underline{\mathbf{m}}^l$
  active DBN $\leftarrow \underline{\mathbf{m}}^l$
until end

5 Experiments

In this section, the practicability of the model and the decomposition for the recognition of a shelf-assembly is shown. This example is motivated by research in human-robot-interaction for humanoid robots in household settings. In this realistic scenario complicated tasks like a shelf-assembly arise. Recognizing the human's assembly plan and the state of the shelf-assembly will allow for robot assistance to the human. A virtual

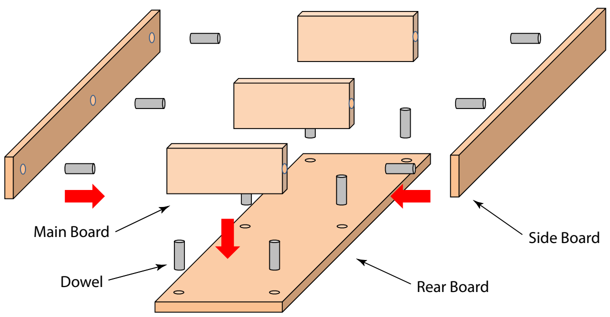


Figure 3: Unassembled shelf parts with graphically annotated assembly instructions.

household with objects and furniture of identical size to a real-world household was used as a testbed. In this virtual environment a user simulated the assembly by manually setting the trajectories of the head and hand as well as setting the hand status (open, grasping). Zero-mean Gaussian noise ($\sigma^2 = 20\,\mathrm{cm}$) was added to these measurements. This noise level is comparable to the noise of positions extracted from the real robot's vision system. In the following the details of the assembly and the recognition results will be presented.

5.1 Shelf Assembly

Fig. 1 depicts the TPG for the shelf-assembly shown in Fig. 3. Roughly speaking, one starts with unconnected boards and a set of dowels and assembles the shelf by completing the two consecutive subplans. The first subplan consists of inserting dowels into the rear board and affixing boards until all slots are filled. Each of these ($k_{\max} = 3$) middle boards is attached to the rear board using ($l = 2$) dowels. It is assumed that during the assembly every two consecutive dowels are inserted in one row. Leaving the order unspecified as to which dowels and boards are inserted, the number of permissible plans amounts to $\prod_{i=1}^{k-1} l^i$. It is exactly this combinatorial explosion that makes this problem challenging and will show the practicability of the proposed approach. After insertion of $l_{max} = 2k_{max}$ dowels and $k = k_{max}$ boards, the consecutive subplan starts. The plot of the second subplan resembles the first subplan as again ($\bar{l} = 3\bar{k}$) dowels need to be inserted prior to attaching the $\bar{k} = 2$ sides of the shelf. The initial state is given by $\bar{l} = l - 2k$ and $\bar{k} = k - 3$. Thus, the following independence relations hold if $\underline{\mathbf{m}}_t^l$ is given

$$\bar{l} \perp l, k\,, \qquad \bar{k} \perp l, k\,. \tag{8}$$

In Sec. 5.2 it will be shown how to exploit subplan precedence according to (8). Independence of concurrent subplans can be exploited more easily as all other intentions are independent of the state of the shelf assembly reducing the world state to the state of the shelf-assembly.

5.2 Assembly Rule Base

In the $\underline{\mathbf{i}}_t^0$-layer, the following primitive actions are considered: $A = \{\underline{d}\text{owel}, \underline{j}\text{oin}, \underline{r}\text{ead}, \underline{c}\text{lean}\}$, $a \in A$. The relevant set of states is given by $Q = \{\#dowels\} \times \{\#boards\}$, $q_{l,k} \in Q$, i.e. $q_{l,k} := (\#d = l, \#j = k)$. If the human performs a manipulation with an action a, there is a change in the state $q_{l,k}$ or more formally $\left(q_{t+1}^0\right) = f^q(q_t^0, a_t^0)$. Regarding the shelf-assembly, the state becomes the state of the shelf-assembly and the actions are the actions used in the assembly. The POP in Fig. 1 conveys that $f^q(.)$ has a sparse structure. Converting the TPG into sets of rules per subplan gives Tab. 1. Two disjunct sets of rules and states separable

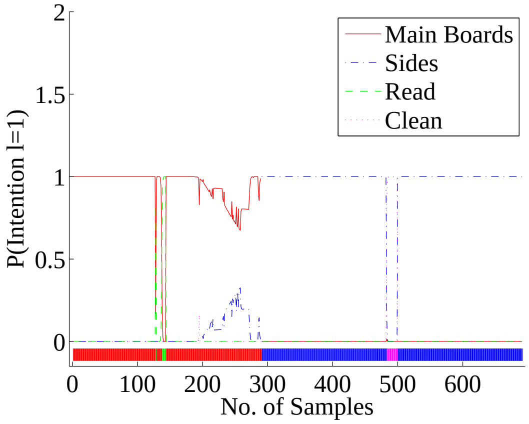
$$\begin{array}{r|lcl|l}
m = 1 & (\mathbf{d} \wedge q_{l,_}) & \rightarrow & (q_{l+1,_}) & l \leq 6 \\
 & (\mathbf{j} \wedge q_{l,k}) & \rightarrow & (q_{l,k+1}) & l \geq 2k \\
 & (\mathbf{r} \wedge q_{l,k}) & \rightarrow & (q_{l,k}) & \\
 & (\mathbf{c} \wedge q_{l,k}) & \rightarrow & (q_{l,k}) & \\
 & & & & \\
m = 2 & (\mathbf{d} \wedge \bar{q}_{l,_}) & \rightarrow & (\bar{q}_{l+1,_}) & \bar{l} \leq 6 \\
 & (\mathbf{j} \wedge \bar{q}_{l,k}) & \rightarrow & (\bar{q}_{l,k+1}) & \bar{l} \geq 3(\bar{k}) \\
 & (\mathbf{r} \wedge \bar{q}_{l,k}) & \rightarrow & (\bar{q}_{l,k}) & \\
 & (\mathbf{c} \wedge \bar{q}_{l,k}) & \rightarrow & (\bar{q}_{l,k}) &
\end{array}$$

Table 1: Rules for the shelf-assembly.

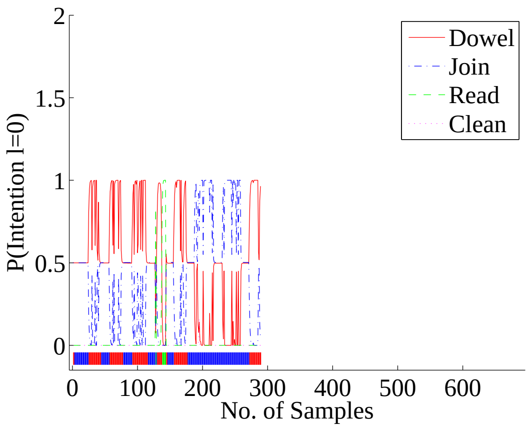
by $m \in \{1, 2\}$ may be distinguished that reflect the two subplans entailed in the TPG. The plot is roughly the same for both subplans: inserting dowels and board until completion. Yet, for the two subplans different constraints regarding the necessary number of dowels and boards apply. These constraints are given in the rightmost column. For each subplan, the first constraint is the maximum number of dowels insertable for this subplan. Second, there is a minimum number of dowels that need to be inserted before a further board may be inserted. This condition ensures that these dowels are not used for holding a board already. The subplans are complete as soon as $q_{\max,\max}$ respectively $\bar{q}_{\max,\max}$ are attained with a probability larger than a specified threshold. Thus, all middle boards were attached to the rear and the sides were affixed to the middle boards.

5.3 Results

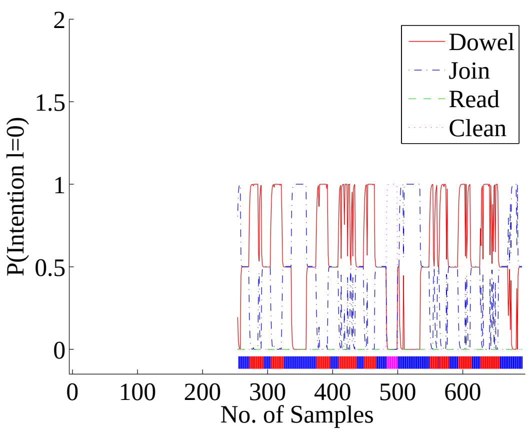
Using the generative model of a shelf-assembly described above, a LIM was generated, compiled into a



(a) Estimates for $l = 1$ intentions.



(b) Estimates for subplan *Assemble Main Board*.



(c) Estimates for subplan *Sides*.

Figure 4: Posterior intention estimates ($l = 1$) for the respective subplans are given in (b) and (c). Note that both estimates span the identical observation period and overlap. The averaged posterior estimates for level 1 intentions are depicted in (c). In each figure the lowest bar gives the maximum likelihood estimates.

DBN, and used with simulated data. Fig. 4 (a)-(c) give the recognition results. In these figures, the posterior probabilities for the respective intentions over a time span of $\sim$ 640 samples are given. This corresponds to recorded data of about $3\frac{1}{2}$ minutes. During this time, the human assembled the shelf in an arbitrary manner but in adherence with the plan given in Fig. 3. As interleaving plans, *Read Manual* ($\sim$ sample 140) and *Clean* ($\sim$ sample 490) occurred. Thus, during the shelf-assembly, the human started to read the manual to assure himself of the assembly plan and he cleaned one board prior to assembling the sides. Fig. 4 (a) gives the posterior probabilities of the highest-level intentions (*Assemble Main Board*, *Sides*, *Read Manual*, *Clean*) for the entire shelf-assembly. Below, Fig. 4 (b) and (c) show the posterior probabilities of the lowest-level intentions (*Insert Dowel*, *Attach Board*, *Read Manual*, *Clean*). Fig. 4 (b) shows the estimates for the first subplan *Assemble Main Board* and Fig. 4 (c) gives the results for the *Sides* subplan. The effectiveness of the subplan precedence can be seen in the stability of the overall results in Fig. 4 (a). Fig. 4 (b) and (c) give the results for the competing subplans. After initializing the second subplan model, both subplan models coexist until the likelihood for the *Sides* model allows for neglecting the initial model. The coexistence can be seen in Fig. 4 (a) around sample 250. The impact of negative evidence is noticeable after the first *Join* action: the first subplan is no longer possible and therefore only the last subplan remains. The results in Fig. 4 (a) are robust during the switching period even though strong noise acts on the measurements (samples 200-300).

5.4 Conclusion and Future Work

In this paper, a probabilistic approach to plan recognition based on DBNs was presented. The proposed approach is especially applicable for the keyhole recognition of multiple interleaving partially ordered plans. On the basis of the introduced definition of a subplan, the plan structure can be exploited to address the scalability issues plaguing all layered or hierarchical approaches. In detail, it is demonstrated how the state size can be reduced by exploiting the subplan precedences and plan concurrency. Using the derived subplans, a Layered Intention Model can be compiled into DBNs with smaller state spaces. Inference in these DBNs with multinet variables amounts to standard inference in each DBN, calculation of the posterior probability of the multinet variables, weighted averaging of the calculated posterior probabilities and a possible DBN switch. Thus, only minor changes to existing inference mechanisms are necessary. Even though the presented decompositions allow for significant reductions of the state spaces, the exponential growth in complexity as the state space increases remains. We believe this problem needs to be addressed by an exploitation of the plan structure in conjunction with approximate inference, e.g., an RBPF. Another main challenge is an increasing depth of the layer structure. This problem will become eminent as domain sizes grow and the level of detail increases. A more practical issue is the robustness of the proposed switching approach. At the moment, the practitioner will have to custom-tailor his switching function. It remains future work to investigate how the plan structure may be exploited (semi-)automatically.

6 Acknowledgments

This work was supported in part by the German Research Foundation (DFG) within the Collaborative Research Center SFB 588 on "Humanoid robots – Learning and Cooperating Multimodal Robots".

References

[1] D. V. Pynadath and M. P. Wellman. Probabilistic State-Dependent Grammars for Plan Recognition. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI-00)*, pages 507–514. Morgan Kaufmann Publishers, INC., San Francisco, California, 2000.

[2] H. H. Bui. A General Model for Online Probabilistic Plan Recognition. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1309–1315, 2003.

[3] E. Horvitz, J. Breese, D. Heckerman, D. Hovel, and Koos Rommelse. The Lumière Project: Bayesian User Modeling for Inferring the Goals and Needs of Software Users. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, pages 256–265. Morgan Kaufmann, San Mateo, California, July 1998.

[4] N. Lesh, C. Rich, and C. Sidner. Using Plan Recognition in Human-Computer Collaboration. In *Proceedings of the Seventh International Conference on User Modeling, pages 23–32*, 1999.

[5] D. Albrecht, I. Zukerman, and A. Nicholson. Bayesian Models for Keyhole Plan Recognition in an Adventure Game. *User Modeling and User-Adapted Interaction*, 8(1-2):5–47, 1998.

[6] O. C. Schrempf and U. D. Hanebeck. A Generic Model for Estimating User Intentions in Human-Robot Cooperation. In *Proceedings of the 2nd International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, pages 251–256. Barcelona, Spain, September 2005.

[7] S. Carberry. Techniques for Plan Recognition. *User Modeling and User-Adapted Interaction*, 11 (1-2):31–48, 2001.

[8] C. F. Schmidt, N. S. Sridharan, and J. L. Goodson. The Plan Recognition Problem: An Intersection of Psychology and Artificial Intelligence. *Artificial Intelligence*, 11(1–2):45–83, 1978.

[9] N. Blaylock and J. F. Allen. Fast Hierarchical Goal Schema Recognition. In *Proceedings of AAAI 06*. AAAI Press, Boston, Massachusetts, 2006.

[10] H. A. Kautz. A Formal Theory of Plan Recognition and its Implementation. In *Reasoning About Plans*, pages 69–125. Morgan Kaufmann Publishers, San Mateo, California, 1991.

[11] E. Charniak and R. P. Goldman. A Bayesian Model of Plan Recognition. *Artificial Intelligence*, 64:53–79, 1993.

[12] R. P. Goldman, C. W. Geib, and C. A. Miller. A New Model of Plan Recognition. In *Proceedings of the 1999 Conference on Uncertainty Artificial Intelligence*, pages 245–254. Morgan Kaufmann, INC., 1999.

[13] D. Avrahami-Zilberbrand, G. A. Kaminka, and H. Zarosim. Fast and Complete Plan Recognition: Allowing for Duration, Interleaved Execution, and Lossy Observations. In *IJCAI Workshop on Modeling Others from Observations*, 2005.

[14] D. V. Pynadath. *Probabilistic Grammars for Plan Recognition*. PhD thesis, University of Michigan, 1999.

[15] T. Dean and K. Kanazawa. A Model for Reasoning about Persistence and Causation. *Computational Intelligence Journal*, 5(3):142–150, 1989.

[16] K. Murphy. *Dynamic Bayesian Network : Representation, Inference and Learning*. PhD thesis, University of California, Berkeley, 2002.

[17] M. Pardowitz, R. Zöllner, and R. Dillmann. Learning Sequential Constraints of Tasks from User Demonstrations. In *Proceedings IEEE-RAS Intl. Conf. on Humanoid Robots (HUMANOIDS)*. Tsukuba, Japan, 2005.

[18] S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach (International Edition)*. Prentice Hall, 2003.

[19] J. Bilmes. Dynamic Bayesian Multinets. In *Proceedings of the 16th Conf. on Uncertainty in Artificial Intelligence*, pages 38–45. Morgan Kaufmann Publishers, INC., San Francisco, California, 2000.

[20] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan-Kaufmann Publishers, INC., 1988.