Capabilities-based plan recognition

Robert Suzić
Dept of Systems Modelling
Swedish Defence Research Agency
SE 164 90 Stockholm, Sweden
robert.suzic@foi.se

Pontus Svenson
Dept of Data and Information Fusion
Swedish Defence Research Agency
SE 164 90 Stockholm, Sweden
ponsve@foi.se

Abstract -

The new types of opponents and new kinds of situations that the Swedish defence forces are facing today calls for new information fusion methods. In order to provide commanders with the ability to predict the enemy's future actions, tools for automatic plan recognition are needed. In this paper, we take the first step towards constructing such a method based on recognizing plans using information about the capabilities of the enemy. The method combines our previous work on plan recognition using bayesian networks based on comparing enemy movements to their doctrines and methodology for force aggregation using capabilities. We describe how the plans of the enemy are built up so that their intended effects are achieved. The relations between these, their resources and the context in which they are acting are used to construct the plan recognition network. We discuss the need for including termination states in the plan recognition method and describe ontologies that are aimed to support the construction of the bayesian networks needed for capability-based plan recognition. We conclude with a discussion of possible extensions of the method.

Keywords: plan recognition, operations other than war, predictive situation awareness, bayesian networks, ontologies

1 Introduction

The vastly increased amount of sensors that are used in today's battlefield make it necessary for commanders to use computer tools in order to get sufficient situational awareness and increase the quality of their decisions.

An important part of decision-making for commanders in the field is to have some sort of predictive situation awareness. This is achieved when, in addition to knowing where own and opposition forces are located, the commanders are also able to pose and rank hypotheses regarding the enemy's future behaviour. Such hypotheses, of course, are always uncertain, since we can never be sure what the enemy's goals are.

One aim of threat assessment, or level 3 information fusion [1], is to produce computer tools that help commanders reason about possible future plans of actions for the enemy. In this paper, we present an extension of a previously introduced [2] method for plan recognition. Recognizing the enemies plans as soon as possibly after they start acting on them enables commanders to act pro-actively instead of passively reacting to enemy actions.

Previous work on plan recognition has been based on using doctrinal knowledge to determine plans based on the movement patterns of enemy units. In the military operations other than war (MOOTW) that the Swedish defence forces are performing now and in the future, we will face new kinds of opponents. Instead of a technologically advanced adversary that follows rigid doctrines for their behaviour, we meet gangs, clans, militias and other loosely organized groups whose behaviour is more governed by religion, culture, and feelings than by military doctrine. Often, we will be in situations where there are several opposing sides, none of which follows a clear doctrine, and where we need to predict the behaviour of all the different parties, taking into account the reactions of the other sides to the actions of one. Such complicated interaction between several antagonistic and/or neutral sides might possibly be modeled using game theory.

In this paper, we concentrate on the problem of determining possible plans of actions for the enemy based on what capabilities that they are seen to have. Capabilities, which will be further discussed in section 7, can be seen as as resource of some kind that is at the right place and time (i.e., in the right context) to be used. The simple idea underlying the work presented here is that by looking at what equipment the enemy is bringing, we can infer what their plans are. For example, if we in a riot situation see a group of people with a grenade-launcher, it might be reasonable to conclude that their aim is the presidential palace, since the grenade-launcher can be used to attack it even though it is surrounded by walls and a garden. This simple example illustrates the importance of the context in which we see a particular resource. If there is no appropriate target for a grenade-launcher in the area, then the presence or non-presence of it tells us nothing about the enemy's intentions.

Capabilities have previously been used to classify clustered groups of objects in MOOTW situations [3], providing one component of a force aggregation method which is applicable when the enemy's unit structure does not follow a known doctrine. In this paper, we define the concept of capability more precisely. By combining the plan recognition with the aggregation presented in that paper, the first version of an information fusion system for operations other than war scenarios could be built.

This paper is outlined as follows. Section 2 gives the idea behind the plan recognition presented here and discusses how it can be combined with previous work to get a more robust estimation of the enemy's plans. Sections 3 and 4 discusses the relationship between plans, intentions and effects. The following sections discuss the need for termination states in the bayesian networks used for plan recognition and describe the ontology that is needed to map capabilities via plans to effects. How to get capabilities from observations is discussed in section 7, which is followed by a discussion and suggestions for future work.

2 Plan recognition in several ways

By observing a group of agents, we can infer their future actions (their plans) in three different ways.

1. By looking at the way that the agents move around in the environment, we can determine if they are aggressive and also try to predict where they are headed. For example, a crowd that moves randomly will pose less of a threat than a crowd which purposefully moves towards their goal. In addition, if the crowd tries to sneak around police patrols, we get additional information regarding their behaviour.

2. A more difficult way of inferring goals is to look at what kind of equipment and resources that the agents are carrying. This will impose limits on what they are capable of doing. For instance, if we observe a person in the crowd carrying dynamite, we know that they are capable of blowing things up. This kind of plan recognition is the focus of this paper. Instead of just looking at the equipment and resources that an enemy possesses, we will use information about the capabilities that they have. The method that infers plans based on the observed capability of the enemy will be called CbPlR (for capabilities-based plan recognition) in the following.

3. The most robust plan recognition is obtained by combining the two methods. This will be done in future work.

If we are given several different estimates of the enemy's future actions, it is useful to compare them and see in what way the outputs of the different methods differ [4, 5]. Doing this gives us more confidence in the results of the plan recognition method and also allows us to concentrate on those plan alternatives for which the methods produce significantly different probabilities. The comparison could also be used to guide information gathering resources, so that the dissimilarity between the outputs is minimized.

3 Plan recognition for Effect Based Assessment

The new concept of effect based operations (EBO) is a way of thinking where desired effects (higher order goals at strategic level) are put into focus when planning, executing, and assessing military operations. EBO forces decision makers to look at outcomes and their explanations more so then actions taken [6], [7]. The focus of EBO is on causal explanations (models of mechanisms) that represent relations between action and cause (effect). Plan recognition is methodology that is aimed to facilitate an EBO process by modeling adversary (agents) activities and thereby provide predictive situation awareness for the user (own force tactical commander). Plan recognition is one of the methodologies that that is aimed to transform information about adversary into usable knowledge for EBO. Plan recognition gives focus of attention (alert) to the user, identifies threatening behaviors of an agent or group of agents and gives a clue about the (most probable) effects that agents may cause.

In this paper, we assume that the behaviour of the enemy can be modelled using an EBO approach. Thus, we assume that there are relations between the plans that the enemy follows and their effects, even if the enemy does not plan according to the EBO process.

By agent we mean everything that can act using its actors and perceive using its perceptors. An agent or group of agents acts in a certain environment. Plan recognition is the process of inferring (reasoning about) agents plans given a priori knowledge about agent's behavior, our current observations and knowledge about the environment.

By W we denote all possible states of interest of the agents. $w \in W$ will denote a current (actual) states.

We divide state w into five components. The x component contains states that are influencing the agent's decisions and e are the states that are assumed to be controlled and/or achieved by the agent. This division of actual world state w seems to be a natural choice considering the definition of agents where both perceiving part (in analogy to x) and acting/effecting part (in analogy to e) are present. E.g. x can be weather conditions, what other agents do and e are the effects to be achieved (e.g., making opponent agents to believe in certain hypothesis or staying at same place and protecting an important road).

More explicitly, we will write

$$w = [x, e, ps, rs, ts] \qquad (1)$$

Local effects e_i caused by different agents in a certain situation produce eventually aggregated (global) effects. When an agent is executing a certain plan it changes environment or reveals itselfs. Those states, influenced by plans, we call reveling states (rs). A way of executing plans in military terms is called doctrines. Knowledge about agent doctrines can provide hints about what plans are most probable given certain behavior patterns. Precondition states (ps) are states that are plan material and what the agent is capable of doing. Those states are closely related to what agents, their physical resources, positions and eventually capabilities are. In an expanded way the factors as the motivation are taken into concern. Termination states (*ts*)are conditions that prevent certain plan alternatives from being possible. In this paper we focus on precondition states and the termination states that are related to capabilities. Those state descriptions are of use in plan recognition in a MOOTW setting. We will construct a conceptual framework in the form of ontologies that integrate various state descriptions.

However, each plan does not automatically lead to the desired effect due to frictions. The mission (war) frictions can be divided into environmental and agent based. Change in environment such as weather change or lacking knowledge about the environment may prevent achieving desired states. By agent based frictions we mean that other agents influence or prevent desired outcomes. Therefore even effects are a probability distribution where each effect is combination of all agents actions (both hostile and enemy) and the environment where agents are acting.

4 Plan recognition and intentions

The implementation of a plan can be seen as the changing function of current state w_t to a future state w_{t+1}. A sequence of states $(w_1, \dots, w_i, \dots, w_T)$ eventually leading to the end, desired, state $w_T = end\_goal$ that reflects agent's *intention*. To avoid to be too predictable in military applications it is undesired in some situations to execute the same plan more than once. The main difference between intention and plan is that intention is some desired (end) state while a plan is a way for transforming state w_t into new state w_{t+1}. Executing the plan eventually leads to the desired (intended, goal) state w_{end}. Most often case is that no plan $\pi_{t=end}^{t=start}[1]$ is exactly alike as other plan $\pi_{t=end}^{t=start}[2]$. Therefore it is important to describe a space of possible plans in a *generic* manner that captures recognition of plans in a robust manner but also makes it possible to distinguish between plan alternatives. In [2] it was shown that, by using the soft computing methodology, it is possible to find the best matching candidates to plan alternatives $\pi = (alt_1, \dots, alt_i, \dots, alt_n)$ given relevant states w. What is output from the method if a distribution

$$P(\pi_t|w) \tag{2}$$

if possible plans given the observed actions.

The method uses observations, a model of the environment, strength balance and other agents perception properties to derive these distributions. The plan recognition model was designed to use both dynamic on-line observations and static knowledge in order to find out about the agents plans.

Instead of modeling plans as all possible sequences of actions and sub plans we assume that a markov property holds for the plans: only the plans and actions at the immediately preceeding step influence the current values. The method uses dynamic bayesian networks to estimate plans at current time step, π_t, by using information about plans on plans at previous time step, π_{t-1}. Thanks to the markov property, the modeling process and on-line inference becomes tractable.

The plan recognition method introduced in [2] is a soft computing method that used both fuzzy sets and Dynamic Bayesian networks (DBN) [8]. A DBN with a proper description offers flexibility beyond hierarchical modeling in a consistent manner.

In Figure 1 the multi-agent plan hierarchy consists of the plan of the company at the top level, the platoon plans at the next level and the tank plans. Our DBN modelling approach is that the company plan causes change in platoon plans and platoon plans cause change in group (tank) plans. One of the key variables that reveal agent plans is their formation, the spatial pattern they form. It is represented as a Bayesian node in the network. According to the model in Figure 1, the variable "Observed Formation", grey node, depends on

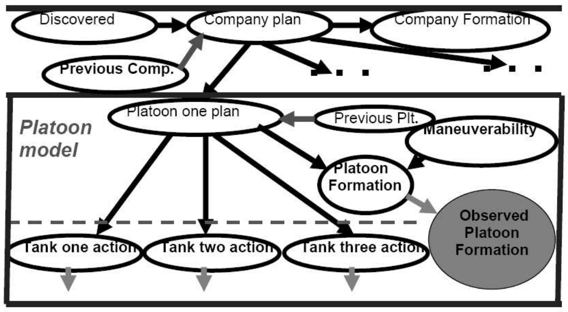

Figure 1: A schematic illustration of the DBN used for plan recognition based on enemy movements.

the actual formation. Due to environment, uncertain observations and possible agent's coordination problems we are not always able to observe its formation pattern. A fuzzy set function takes the estimates of the tank positions as inputs and as output returns the distribution of the observed formation's values. The result is entered as soft evidence in the variable "Observed Formation". By Bayes rule it has influence, backwards, on the value "Formation". As we see In Figure 1, platoon plans and company plan are connected over time because of higher inertia than inertia of tank plans (actions). If the whole company is attacking at one time step there is a significant probability that the company will continue to execute its plan alternative attack in the next time step.

5 Termination state that depend on capability

Plan recognition in previous approaches has been exemplified for non-urban military applications. Agents were acting in an open-terrain environment. Here, we shift focus from classical warfare in open terrain environments, to military operations other than war in urban environments. Urban environments offer many opportunities for agents to act. However, the urban environment has more crisp/hard restrictions than open field terrain. For example, the passages between buildings allows only a *limited* number of rioters to pass at once; in crowd situations this restriction becomes more obvious. In open terrain, hard restrictions are rare and they were modeled in soft manner in [2] (e.g. carrying of the terrain in plan recognition is a soft variable with the probability that tanks can pass).

In [9] , a plan recognition model has been introduced for recognition of agent's plans in an office environment; one of the assumptions in that an agent can not have plan alternative (activity) of copying papers if there is no copying machines available in the room where the agent is. This part of their plan recognition model is called termination variable (node in an *abstract hidden markov model* (AHMM)) that represent a stopping condition (state) for certain plans (also called policies in the paper [9]). The (crisp) stopping, termination, condition was not modeled in [2]. Instead, soft precondition states in open terrain environment were used. Hence, we propose to use such termination states in CBPlR for MOOTW in urban areas. The benefit of introducing termination states in plan recognition is twofold.

The first benefit is that by using termination state and excluding certain plan alternatives, we *reduce computational complexity*; the complexity is lower if we remove certain plan alternatives from conditional probability tables rather than put zero values depending on termination states. The second benefit is that the user (tactical commander in our case) gets a *smaller number of hypotheses* (guesses about agents plans) to consider. (An alternative way of achieving this is to group alternatives into classes: most probable, less probable and least probable, or by using *equivalence classes* of plans[10, 11].) In CBPlR, if the agents do not have a capability that is crucial for executing a certain plan, then that plan alternative should be not be considered. An example of this could be that if an agent moves to a place where its capability cannot be used, some plan alternatives can no longer be carried out, (see `Context for Deployment` class in Figure 2). Another example is that an agent during an ongoing operation uses up its resources and thereby losses a capability or several capabilities that are necessary for execution of certain plan alternative. The risk and disadvantage with termination states is that it reduced the robustness of the method. Hence, termination states should be used only in situations where we are certain that certain precondition states do or do not exist.

6 Ontology for Capability Based Plan Recognition

The concept of Multi-Entity Bayesian networks (MEBN), described in [12], is a first step in direction of building Bayesian networks (BN) [13] in a *flexible* manner. MEBN are based on separate graphical models. Those models are reusable pieces that during the process of situation-specific BN construction produce a sequence of BNs contextual variables. MEBN has the ability to absorb new facts about the world, incorporate them into existing theories, and/or modify theories in light of evidence. MEBN fragments, network entities, specify local dependencies among a collection of related hypotheses. Consequently, they specify joint probability distributions over unbounded and possibly infinite number of hypotheses. These properties lead us to propose MEBN as the key methodology when designing knowledge fragments for CBPlR.

In particular, here we focus on giving an ontology for MEBN that support CBPlR at different abstraction and software development levels. In [14] a generic ontology for MEBN has been proposed. Ontology stands for a specific perspective, or an assumption, about the target application area to be represented. The reason ontologies are becoming so popular has to do in large part with what they promise: a shared and common understanding of some domain that can be communicated among people and application systems [15].

Here we present the following generic ontologies for capability based plan recognition:

1. Upper ontology that can be used both inside but also outside the scope of CBPlR. (represented in UML)

2. Mid-level ontology whose purpose is to be a conceptual framework for dynamic integration of MEBN (represented in BN)

3. MEBN specific description on how fragments are put together (an example of connected MEBN).

6.1 Upper-level ontology for Capability Based Plan Recognition

In [16] an upper (generic) ontology for C2 was introduced in Unified Modeling Language (UML) [17]. It was later modified to suit development and interoperability between plan recognition systems in [18].

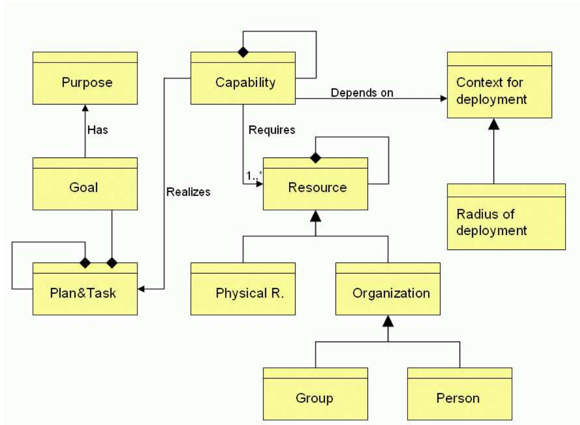

Figure 2: Upper-level ontoloogy in UML

In Figure 2 we introduce higher-level ontology, represented in UML, that can be used beyond capability based plan recognition domain. It is meant to enable a common understanding of terms and most representative relations. The use is twofold. An upper ontology is a knowledge representation model in a structured manner; this implies that level of common understating, model reuse and interoperability become much higher by using a common ontology. It is also a guideline for how to implement middle ware and low-level ontologies that are intended to be machine interpretable to a much higher degree.

In [18] a capability class was not represented. To support CBPlR by ontologies, here we introduce capability as a class. We see capability as a class that can be aggregated, i.e. a capability can consist of other capabilities. However, for a capability to exist requires both possession of resources and the right context to use them. Depending on what capabilities an agent has, different plan alternatives can be realized. A plan can be subordinated to a task and can be superior to other tasks, see [19].

As exemplified in [20] a resource can be physical and organizational. A resource or resources in ”right” place and time constructs a capability. Generally, a capability is more dynamic (situation-dependent) than a resource.

6.2 Middleware ontology for MEBN

A plan structure often follows an organizational structure. Such structure can in MOOTW case be obtained from capability based aggregation [3]. The plan spaces (hypothesis spaces) in capability based plan recognition is based on what capabilities agent or agents has in the given context. Next step is composing MEBN structure based on force aggregation that is aimed to recognize plans (activities) of agents in a dynamic environment. Composing such structure needs a human or some rule based program that uses this ontology, a set of predefined rules and finally compose a context-specific MEBN. As a last step, such MEBN is fed input and produces qualified guesses of agents activities (plans).

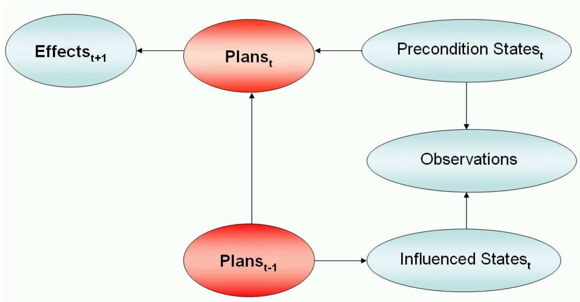

Figure 3: Aggregated DBN

To enable this process of matching MEBN fragments we introduce a middleware ontology that is expressed as an *aggregated* dynamic Bayesian network, see Figure 3. Its purpose is to classify MEBN fragments for insertion into the plan recognition DBN. It consists of plan networks that describe organizational structure. The capability networks are combinations of network fragments describing capability resource and context dependencies. Influenced (revealing) states describe dependencies between evidence we observe and relations to hidden states, that are in next turn related to certain plan alternatives.

6.3 Lower level ontology for MEBN

Lower level ontology describes properties of fragments, which type of nodes a certain fragment consists of and information about their validity in different domains. This was described in [14]. In contrast to upper-level ontology that is meant to be tool for common understanding between software developers and domain experts the focus of lower level ontology is to support machines to interpret this ontology. The lower-level ontology is a border case between a ontology and rule based model.

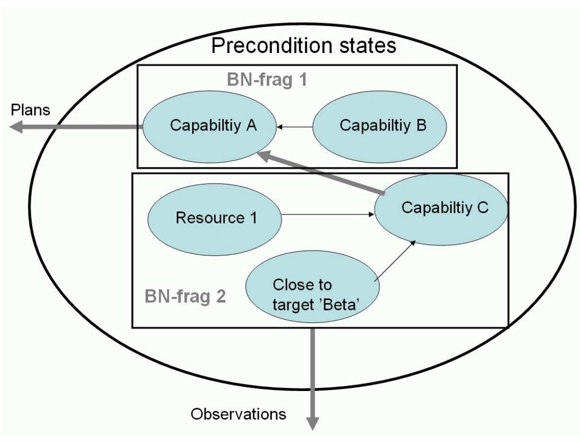

Figure 4: Bayesian network fragment belonging to precondition states.

The aggregated node of precondition states described in middle-ware ontology consists of different BN-fragments. In Figure 4 we show two of the BN-fragments. BN-fragment one (BN-frag 1) is aimed to describe causal relation between how capability *B* influence capability *A*. As explained in upper-level ontology section a capability is not only dependent on resources deployed. Capability is also dependent on (influenced by) the context in which the resource is used. A basic example, depicted in Figure 4, is how close a resource is to the *presumed* target beta.

The middle-level ontology classify and model *allowed* connections from precondition state fragments to the aggregated nodes *plans* and *observations*.

7 Capabilities and observations

Resources can be observed and identified by human or some automatic classification process. Eventually, capability could be deduced by human or some data base where capabilities are matched to resources. However, a resource does not entail the same capabilities in all situations (contexts). In other words, capability is not only tactical strength of the resource or joint strength of resources. It is also dependent on the context in which agents are operating. A resource can be very useful in some situations and useless in others. BN-fragments can encapsulate knowledge about local relations between capabilities resources and environment where agents are acting. However, evidence that enter states in MEBN need to be context dependent. I.e. a process of contextualizing data into relevant evidence has to be performed. For example, data available about position of the agent and its presumptive target beta couldd be contextualized into classes "Far" and "Near". This contextualization can be achieved by using fuzzy functions [21]. In similar manner, the capabilities can be derived from data about agents resources and type of the environment where agent is acting.

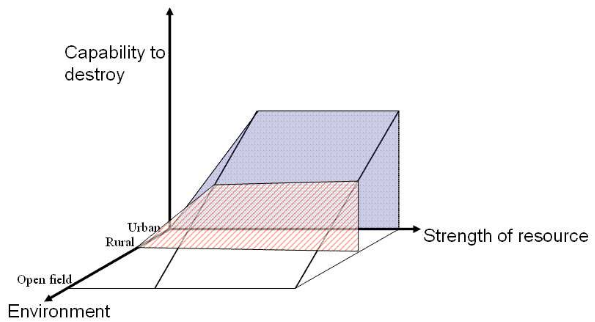

Figure 5: Capabilities in relation to resources and type of environment.

In Figure 5, we show an example where the capability of destroying depends on strength of resources and environment. For example, some anti-tank weapons can be easier deployed in urban environment and achieve greater impact than in open field. Rural environment is something between urban and open field. The fuzzy functions can be discretisized regarding type of environment (qualitative situation parameters such as different types of environment). In other words, in a certain type of environment a certain fuzzy function is valid. Capabilities achieved by fuzzy functions can be used in next turn in the BN-fragments that aggregate capabilities regarding observations and put them into plan recognition context.

8 Discussion and future work

In this paper, we discussed a method to help commanders achieve predictive situation awareness by recognizing opponents plans. The plan recognition was based on what capabilities the enemy units and groups are seen to have instead of on their movements. Early warning of what the opponents are planning gives our commanders the opportunity to act pro-actively instead of just reacting to events as they occur. It also gives them the possibility of prioritizing among the hostile units that are present, in order to be certain that the most harmful opponent does not get a chance to achieve their goals.

A system based on the ideas presented here could be used not only in MOOTW situations, but also in battle against a technologically advanced opponent. It could also be useful for police in riot situations, which are often very confusing and difficult to determine which parts of the area of responsibility that should receive the most attention. By defining appropriate capabilities and goals, the concept presented here could also be extended to be used for plan recognition on a strategic level. This would enable intelligence analysts to more quickly reach conclusions regarding, for example, the intents of a country that suddenly starts producing a resource that can be used for producing weapons of mass destruction.

There is ample opportunity for future work in this area. It would be very interesting to integrate the idea of mixed initiative reasoning [22, 23] with the current method. The human operator could for instance influence in real-time the ontologies that describe how capabilities relate to plans, or the bayesian networks and fragments that are used in the actual plan recognition.

Future challenges are modeling purpose of action, i.e. why the agent wants to achieve certain goal by using a certain plan. As described in [18], a plan stands in relation to goals in following manner. A task/goal is superior or a plan a task can involve several actions, other plans and tasks. In an AHMM, abstraction levels represent abstraction of plans. In abstraction of plans is closely connected to level in command and control, i.e. organizational structure. In [3] the plan structure is depending on the capability force aggregates (more loose connections than classical warfare agents C^2 structures). Goal lattices [24] describe prioritization structure between sensor actions and the user goals; the edges in the goal lattice are weighted and priorities from higher order goals are propagated down to sensor actions. The issue for future research of plan recognition would be to connect plan recognition with goal lattices representing agents' doctrinal (behavior) knowledge. In MOOTW case such doctrines are very hard to describe and differ. Machine learning of different (behavior) models where both agent's goals and priories are represented by goal lattices could greatly utilize efficient plan recognition.

It would also be interesting to extend the framework presented here so that it could utilize also negative information, i.e., if the enemy does not have a certain capability, we can limit the set of hypotheses regarding their future actions. Since we can never be sure that a non-observation is due to the fact that the enemy does not have the capability or if it's caused by our lack of sensor observations, care must of course be taken so that the set of hypotheses is not limited too much.

Another possibility for future work is to combine the new plan recognition method with resource allocation. This could be done in a similar way as in [25, 26], where the objective is to direct our sensor platforms so that we get as good a plan recognition as possible. Another alternative would be to use the present method as a component of a planning system, which helps our commander to plan the deployment of soldiers and other resources. Such a system could show how the enemy's plans might change given that our movements provide them with new opportunities to use their capabilities against us.

Acknowledgements

We would like to thank the following persons for supporting this work, Per Svensson (FOI), Farzad Kamrani (KTH), Martin Eklöf (FOI), Ronnie Johansson (KTH), Stefan Arnborg (KTH) and Joel Brynielson (KTH).

References

[1] D. L. Hall and J. Llinas, editors. *Handbook of Multisensor Data Fusion*. CRC Press, Boca Raton, FL, USA, 2001.

[2] R. Suzić. Representation and recognition of uncertain enemy policies using statistical models. In *Proceedings of the NATO/RTO Conference on Military Data and Information Fusion, Prague, Czech Republic*, in press, 2003.

[3] Pontus Svenson. Capabilities-based force aggregation using random sets. In *Proc 8th International Conference on Information Fusion*, 2005.

[4] H. Sidenbladh, P. Svenson, and J. Schubert. Comparing multi-target trackers on different force unit levels. In Proc. SPIE Vol. 5429 *Signal Processing, Sensor Fusion, and Target Recognition XIII*, pages 306–314, 2004.

[5] H. Sidenbladh, P. Svenson, and J. Schubert. Comparing future situation pictures. In *Proc 8th Int Conf Information Fusion*, 2005.

[6] Maris McCrabb. Effects-based coalition operations: Belief, framing and mechanism. In *Second International Conference on Knowledge Systems for Coalition Operations KSCO-2002*, 2002.

[7] Joshua Ho. The dimensions of effects-based operations. *Australian Army Journal*, II(1), 2004.

[8] Kevin Patrick Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, 2002.

[9] H. H. Bui. A general model for online probabilistic plan recognition. In *Proc 18th International Joint Conference on Artificial Intelligence (IJCAI-2003)*, Acapulco, Mexico, 2003.

[10] Christian Mårtenson and Pontus Svenson. Evaluating sensor allocations using equivalence classes of multi-target paths. In *Proc 8th International Conference on Information Fusion*, 2005.

[11] Pontus Svenson. Equivalence classes of future paths for sensor allocation and threat analysis. In *Signal and Data Processing of Small Targets*, volume 5913 of *Proceedings of SPIE*, pages 363–9, 2005.

[12] K.B. Laskey. *MEBN: First-Order Bayesian Logic for Open-World Probabilistic Reasoning*. George Mason University Department of Systems Engineering and Operations Research, US, 2004.

[13] F.V. Jensen. *An introduction to Bayesian Networks*. UCL press, London, UK, 1996.

[14] Robert Suzić. A generic model of tactical plan recognition for threat assessment. In *Multisensor, Multisource Information Fusion: Architectures, Algorithms, and Applications 2005*, volume 5813, pages 105–116. SPIE, 2005.

[15] M. Eklöf, R. Suzić, and C.H. Yi. Network based intelligence (nbi) — knowledge base development and use. Technical report, Swedish Defence Research Agency, 2005. report no FOI-R-1757-SE.

[16] Klas Wallenius. A generic model of management and command and control. In *7th International Command and Control Research and Technology Symposium*, Quebec City, QC, Canada, September 16-20 2002.

[17] H.E. Eriksson and M. Penker. *UML Toolkit*. Wiley, 1998.

[18] Robert Suzić. Knowledge representation, modelling of doctrines and information fusion. In *CIMI conference*, Enkping, Sweden, 2003.

[19] Robert Suzić. *Knowledge Representation and Stochastic Multi-Agent Plan Recognition*. NADA, Stockholm, Sweden, 2004. Licentiate Thesis, TRITA-NA-05124.

[20] Robert Suzić. Generic representation of military organisation and military behaviour: UML and Bayesian networks. In *Proceedings of the NATO RTO Symposium on C3I and M&S Interoperability*, October 2003.

[21] L. Zadeh. Fuzzy sets. *Information and Control*, (8:338-353), 1965.

[22] J. Allen. Mixed initiative interaction. *Proc. IEEE Intelligent Systems*, 14:14, 1999.

[23] Erik P. Blasch and Susan Plano. JDL level 5 fusion model ”user refinement” issues and applications in group tracking. In *Signal Processing, Sensor Fusion, and Target Recognition X!*, volume 4729 of *SPIE*, page 270, 2002.

[24] Kenneth J. Hintz and G.A. McIntyre. Goal lattices for sensor management. In *Proceedings of the SPIE International Symposium on Aerospace/Defence Sensing & Control*, volume 3720, pages 249–255. SPIE, 1999.

[25] Ronnie Johansson and Robert Suzić. Bridging the gap between information need and information acquisition. In *Proceedings of the 7th International Conference on Information Fusion*, volume 2, pages 1202–1209. International Society of Information Fusion, 2004.

[26] Robert Suzić and Ronnie Johansson. Realization of a bridge between high-level information need and sensor management using a common dbn. In *The 2004 IEEE International Conference on Information Reuse and Integration (IEEE IRI-2004)*. IEEE, November 2004.