Abstract

To incorporate new technical advances into military domain and make those processes more *efficient* in accuracy, time and cost, a new concept of Network Centric Warfare has been introduced in the US military forces. In Sweden a similar concept has been studied under the name Network Based Defence (NBD). Here we present one of the methodologies, called tactical plan recognition that is aimed to support NBD in future.

Advances in sensor technology and modelling produce large sets of data for decision makers. To achieve *decision superiority*, decision makers have to act agile with proper, adequate and relevant information (data aggregates) available. Information fusion is a process aimed to support decision makers' situation awareness. This involves a process of combining data and information from disparate sources with *prior* information or knowledge to obtain an improved state estimate about an agent or phenomena. *Plan recognition* is the term given to the process of inferring an agent's intentions from a set of actions and is intended to support decision making.

The aim of this work has been to introduce a methodology where prior (empirical) knowledge (e.g. behaviour, environment and organization) is represented and combined with sensor data to recognize plans/behaviours of an agent or group of agents. We call this methodology *multi-agent plan recognition*. It includes knowledge representation as well as imprecise and statistical inference issues.

Successful plan recognition in large scale systems is heavily dependent on the data that is supplied. Therefore we introduce a *bridge* between the plan recognition and sensor management where results of our plan recognition are reused to the control of, give *focus of attention* to, the sensors that are supposed to acquire most important/*relevant* information.

Here we combine different theoretical methods (Bayesian Networks, Unified Modeling Language and Plan Recognition) and apply them for tactical military situations for ground forces. The results achieved from several proof-of-concept models show that it is possible to model and recognize behaviour of tank units.

Keywords: Plan Recognition, Decision Making, Knowledge Representation, Information Fusion, Predictive Situation Awareness, Data Fusion.

Sammanfattning

Ett nytt koncept som utnyttjar de nya tekniska möjligheterna har under namnet Nätverkscentrerad krigföring utvecklats i USA. I Sverige utvecklas detta koncept under namnet Nätverksbaserad försvar (NBF). Syftet med NBF är att *effektivisera* processer i den militära domänen bland annat med avseende på önskade effekter, kostnader och tid.

Nya tekniska möjligheter av sensorteknologi producerar stora datamängder för beslutsfattare. För att beslutsfattare skall kunna handla tillräckligt snabbt och åstadkomma *beslutsöverläge* krävs det adekvat och relevant (automatiskt) förbehandlad information.

Processen vars syfte är att stödja beslutsfattarens situationsmedvetenhet kallas informationsfusion. Den kombinerar sensordata och a priori information från olika källor för att ge den bästa tillståndsuppskattningen. En del av informationsfusionen är planigenkänning. Den ger den bästa uppskattningen av en agents (opponentens) eller en grupp av agenters avsikter från en mängd observerade handlingsalternativ och är till för att stödja beslutsfattarens prediktiva situationsmedvetenhet. Planigenkänningsmetodiken som presenteras i denna avhandling inbegriper såväl kunskapsrepresentation av beteenden som oprecist och statistiskt resonerande. Denna metodik introducerar vi som *multiagentplanigenkänning* för beslutsfattande.

För en effektiv planigenkänning som sker i storskaliga system krävs det att de viktigaste sensordata skall vara tillgängliga. Av denna anledning introducerar vi en *brygga* mellan planigenkänning och sensorstyrning där resultat från vår planigenkänning återanvänds för att ge *fokus av uppmärksamhet* för sensorresurser.

Tekniker utvecklas och anpassas för att beskriva och förutsäga komplexa händelseförlopp, med exempel valda i militära insatsoperationer. Ett antal teoretiska metoder (bayesianska nät, Unified Modeling Language, och planigenkänning) anpassas och realiseras i avgränsade militära situationer.

Nyckelord: Planigenkänning, beslutsfattande, kunskapsrepresentation, informationsfusion, prediktiv situationsmedvetenhet, datafusion.

Acknowledgements

This research would not have started at all without the great support of the information fusion research director at Swedish Defence Research Agency (FOI) Per Svensson, Head of Systems modelling Department Farshad Moradi at FOI and Head of System Technology division Monica Dahlén at FOI.

I could not have imagined having a better advisors and mentors for my licentiate research. First of all my most profound thankfulness goes to my supervisor at Royal Institute of Technology (KTH) Stefan Arnborg for his scientifically proven and creativity encouraging guidance. Moreover, I would like to thank deeply Per Svensson for his enthusiasm waking guidance, scientific discussions and great support in writing scientific publications. Finally, my first steps in research community have been supported by Choong-ho Yi in a well structured manner that I am eternally thankful for.

I wish to thank licentiate Ronnie Johansson, member of the Decision Support Group[1] (DSG) at KTH, for enjoyable research cooperation that resulted in several shared publications. Moreover, I wish to express deepest thankfulness to DSG members Joel Brynielsson and Klas Wallenius for instructive and challenging conversations and pleasant cooperation. Additionally, I wish to express my gratitude for challenging discussions to Henric Christensen (KTH), Pontus Svensson (FOI), Hedvig Sidenbladh (FOI), Linus Eklund (KTH), Karim Oukbir (KTH), Jesper Fredriksson (KTH), Dirk Brade (FOI), Gunnar Holm (FOI), Farshad Moradi (FOI), Martin Eklöf (FOI), Magnus Sparf (FOI), Karsten Jöred (FOI), Kjell Ohlsson (FOI), Sten-Åke Nilsson (FOI), Marianella Garcia Lozano (FOI) and other members of the departments 62 and 77 at FOI and other members of KTH.

I wish specially to thank members of Swedish Armed Forces, Claes Henrikson and Jaak Kallak for their help in my doctrine knowledge survey.

Furthermore I wish to thank all my dearest family members and friends who have given me emotional support during the time I was writing this thesis.

[1] www.nada.kth.se/theory/dsg

Contents

Abstract i

Sammanfattning iii

Acknowledgements v

Contents vii

Chapter 1 Introduction 1

1.1 Background 1

1.2 Scientific Contribution 3

Chapter 2 Knowledge representation, modelling of doctrines and information fusion 5

2.1 Knowledge representation and doctrines 5

2.2 Doctrine 6

2.3 Doctrine as a priori source of knowledge for Information Fusion 7

2.4 Plan and Doctrine 8

2.5 Representation doctrines on combat tactical level 10

2.6 Conclusions 12

Chapter 3 Generic representation of Military Organisation and Military Behaviour 13

3.1 Knowledge representation of military units 13

3.2 UML doctrine models 14

3.3 Bayesian Networks (BN) 15

3.4 A Hostile Company Bayesian Network Model Example 16

3.5 Discussion 18

Chapter 4 Multi-Agent Plan Recognition 19

4.1 Introduction to Plan Recognition 19

4.2 Soft computing method for plan recognition: DBN and Fuzzy membership functions 21

4.3 Description of plan recognition model 22

4.4 Simplified representation of the implemented DBN for plan recognition 25

4.5 Scenarios, simulation and results 27

4.6 Contributions 29

Chapter 5 Plan Recognition and Sensor Management 31

5.1 Framework for bridging the gap between high level information need and information acquisition 32

5.2 Realisation of a Bridge between High-Level Information Need and Sensor Management 34

Chapter 6 Further Development 39

Bibliography 41

Chapter 1
Introduction

1.1 Background

To incorporate new technical advances into military domain and make those processes more *efficient* in accuracy, time and cost, a new concept of Network Centric Warfare has been introduced in the US military forces. In Sweden a similar concept has been studied under the name Network Based Defence (NBD). Here we present one of the methodologies, called tactical plan recognition that is aimed to support NBD in future.

Recently, the trend of military operations is moving to *effect based operations* (EBO). "Effects-based operations are coordinated sets of actions directed at shaping the behaviour of friends, foes, and neutrals in peace, crisis, and war.", [2]. One of the main features of EBO is flexibility and the interoperability that includes cooperation and coordination between different defence forces, different defence alliances and even interoperability between defence structures and civil authorities. Still, behind those flashy terms there are significant historical connections. The first example is from the WW2; the German forces had flexible units called "Kampfgruppen" assembled depending on type of tasks [3]. The second example is the coordination between German air and ground forces on the operative level. A final example of the historical connections to EBO and NCW is the example of the coordination of air and ground troops using radio by the Israeli military forces in the Six Days War [4] on the tactical level. The natural question to ask is: What does become conceptually new in NCW? What do we gain by that new concept? The difference between the concepts used several decades ago and those proposed today (NCW) is the use of new technologies and *new methodologies* that minimize costs and maximize utility (effect) of the operations. In NCW processes such as cooperation, information exchange and fusion (combining) of information ought to occur more rapidly than before; in some cases in a fraction of seconds. That would lead to *decision superiority*.

A key process is the *information fusion* process (IF). This is the process of combining data and information from disparate sources with *a priori* information or knowledge. Data and partial (imprecise) knowledge about different entities or phenomena are combined with previous information to gain new state estimates of interest (new information).

Current advances in sensor technology produce large sets of data and information. Particularly only in the period of 2002-2005 the amount of information has grown to levels never seen before in history [1]. Moreover, uncertain, contradictory and imprecise data presents a decision maker with high complexity and ambiguity. In worst case such data can cause confusion and overload. Decision makers, instead of making use of information, might be only concentrating on data interpretation and data processing. A plethora of hypothesis spaces that may depend on each other often leads to intractable problems. "At all levels, commanders are constantly forming decisions based on their current understanding of the world and their ability to forecast the outcome of actions being considered. This ability is forged through years of training, combat experience and a rigorous selection process. And yet, even experienced tacticians are only able to consider 2 or 3 possible courses of action for all but the simplest situations." [5]. Additionally, military commanders (decision makers) often don't have a good "feel" for the sensitivity of their own plans to variations or the unintended consequences associated with the expression of their decisions based on uncertain information. The process of recognizing opponent's plans and its intentions could be a much harder process then mission planning.

The aim of this work has been to introduce a *multi-agent plan recognition* methodology for future IF and NBD systems, where prior (empirical) knowledge of an agent [6], or a group of agents, is represented and combined with sensor data to recognize its plans/behaviours. The output of the plan recognition process is multi-hypothetical (statistical) qualified guesses of the agent's (hostile force) or agents' behaviours/plans. Those results should be a functional part of IFs *predictive situation awareness* (PSA) [5], [7]. Our methodology is supposed to aid/support decision makers, i.e., their PSA, by projection of current state, represented by the common operation picture, into a close future.

To be able to model and use the prior knowledge about agents for plan recognition we have to perform a structured knowledge representation. This part is discussed in Chapter 2 and Chapter 3. In chapter two the focus is on higher ontological level whereas in Chapter 3 we discuss the applicability of the two different methods to model knowledge at different ontological levels: *unified modelling language* and *(Dynamic) Bayesian Networks*. The kernel of the thesis is Chapter 4, where we introduce multi-agent plan recognition for tactical plan recognition. Finally, bearing in mind that plan recognition is heavily dependent on input (sensor) data, we introduce in, Chapter 5, a *bridge* between plan recognition and information acquisition (IA). As a result of the bridge methodology we achieve improved (pro-active) IA that is focused on the most *mission relevant* acquisition tasks.

1.2 Scientific Contribution

- Robert Suzić and Ronnie Johansson. **Realisation of a Bridge between High-Level Information Need and Sensor Management Using a Common DBN.** *In Proceedings of the 2004 IEEE International Conference on Information Reuse and Integration (IEEE IRI-2004). Summarised in Chapter* ***5***
- 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, June 28-July 1, 2004. *Summarised in Chapter* ***5***
- Robert Suzić. **Representation and Recognition of Uncertain Enemy Policies Using Statistical Models**. *In Proceedings of the NATO RTO Symposium on Military Data and Information Fusion,* Prague, Czech Republic, October 2003. *Summarised in Chapter* ***4***
- 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*, Antalya, Turkey, October 2003. *Summarised in Chapter* ***3***
- Robert Suzić. **Knowledge representation, modelling of doctrines and information fusion**. In Proceedings of the CIMI conference, Enköping, Sweden, May 20-22, 2003. *Summarised in Chapter* ***2***
- Robert Suzić. **Kunskapsrepresentation av doktriner och taktiskt uppträdande**. Technical Report FOI-R--0865-SE, Swedish Defence Research Agency, Stockholm, Sweden, June 2003. In Swedish. *Summarised in Chapter* ***2***

Chapter 2
Knowledge representation, modelling of doctrines and information fusion

In this chapter we focus on understanding doctrines and their representation in the object-oriented language Unified Modelling Language (UML). We present a conceptual model of one part of the information fusion process and describe the role doctrines should play in this process. The aim has been to supply information fusion processes with *a priori* knowledge, which in this case is knowledge about an agent or group of agents' tactical behaviour. Models in Figure 2 and 6 are partly reused and modified from [8].

2.1 Knowledge representation and doctrines

Within the project Information Fusion at the Swedish Defence Research Agency (FOI) we performed a study of knowledge representation of doctrines. One role of doctrines in the information fusion process is to prune the hypothesis space generated from sensor information. Another role is to recognise plausible types of behaviours, plans, and connect them to prediction of hostile force state.

The benefit of use of doctrine models is to supply information fusion processes with a priori knowledge, which in this case is knowledge about the agent's (enemy's) tactical behaviour and organisation. Doctrines are only one part of a larger a priori knowledge base which the information fusion process uses to combine with dynamic (sensor) information.

There are two kinds of doctrines. One is descriptive and the other is prescriptive. The descriptive doctrine is more theoretical than the prescriptive where methods and rules are specified. We see prescriptive doctrine as more convenient starting point when modelling doctrines. Doctrine is also divided into a hierarchy of levels from strategic to combat level. The higher in doctrinal levels we are the harder it is to present knowledge for computers. After a historical review considering the strategic and the operational level the focus of our study [9], [10] was the tactical and the combat technique level with a specialization on ground troops, armoured vehicles and to some extent infantry, on platoon and company level up to battalion level, see Figure 1. Additionally, bearing in mind that this is a methodology project and must be based on publicly available literature we used mainly Swedish doctrine publications, see [3], [11].

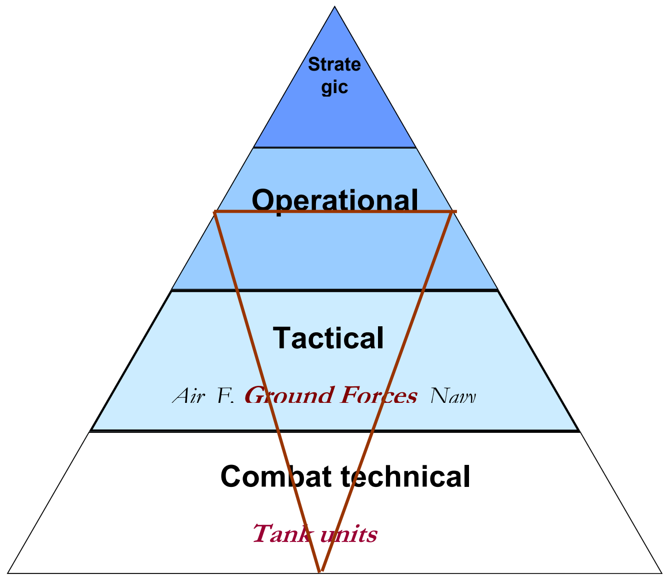


Figure 1. Focus in doctrine study

Our goal considering doctrines is to understand and represent doctrines in a structured manner with regard to uncertainty and information fusion process requirements.

2.2 Doctrine

Doctrine is a collection of knowledge about a military organisation. It gives advice on how military tasks should be solved; see [12].

Doctrine is divided into the following levels: strategic - , operational - , tactical - and combat technique. The fact and the trend is that doctrine levels overlap more and more with each other. Operational level is about coordination of operations between different types of units, e. g. combined action of navy and aircraft forces. On the tactical level one studies the single battle as well as coordination and effects of using different military units; see also [3]. On the combat technical level one studies how technology is used in battle and basic unit behaviours.

2.3 Doctrine as a priori source of knowledge for Information Fusion

Changing doctrines takes time and therefore doctrines are assumed to be"rigid" for longer time intervals, see [12]. Therefore we find the use of the doctrine as prior in IF motivated. However, due to linguistic vagueness it is hard to model doctrines. We find doctrines at (combat) tactical level as less linguistically vague than doctrines on operational level. Due to increased uncertainty, when modelling doctrines for information fusion processes, the first step should be to make models for tactical behaviour before and after the battle. Moreover, there are many unpredictable factors that can play a decisive role in warfare. An important observation made during this study (and confirmed by history) is that uncertainty increases dramatically as soon as the first shot is fired. Therefore it would be difficult to make battle models which give a precise prediction of the enemy's course of action, i.e. the plan that is most likely.

In general terms, we understand information fusion as the process of combining information and inference to obtain improved state representation, estimation and prediction, see also [13]. In military applications, the main challenge is to represent knowledge about the enemy in a dynamic and uncertain environment, i.e. in the fog of war. Doctrine knowledge should be a building block, part of a priori knowledge about the opponent, in a future information fusion system. An example how doctrine can be used is presented in by Matheus, Kokar and Baclawski in [14] where the doctrine description (knowledge) is conceptually seen as the specialisation of a situation object. The collected data/information can be anything from sensor information and terrain description to knowledge about tactical behaviour of the enemy given the knowledge about his doctrines and current observations see [15]. Some examples of what the world state can contain are: agent's position, type of agents, their plans and prediction of their state such as position.

The role of doctrines in the information fusion process is to:

1) Prune the hypothesis space generated from the sensor information and obtains more certain information

2) Recognise types of behaviours i.e., certain plan alternatives, and connect them to prediction of the state.

In order to achieve the goal of using doctrine knowledge in the information fusion process it is necessary to represent the knowledge in a structured and intelligible way, both for application engineers and computers. Keywords are knowledge representation and ontology; knowledge representation is "the science of designing computer systems to perform tasks that normally involves human intelligence" [16]; ontology stands for a specific perspective, or an assumption, about the target application area to be represented. The reason why ontologies are becoming so popular has to do with what they promise, "a shared and common understanding of some domain that can be communicated among people and application systems" [17].

There are different ways of expressing this knowledge. One way of representing knowledge is a textual description. Another way is a combination of textual representation and some formal computer language, e. g. UML, XML etc. In this thesis we use UML (Unified Modelling Language, see [18]) to describe different types of doctrines and their interrelationships. Software engineers use UML as standard when describing ontologies, see [19]. Moreover, UML provides easily understandable graphical representations of classes, sequences and use cases. Additionally, UML is more appropriate for software developers. The most known type of UML representation is the class-diagram representation.

2.4 Plan and Doctrine

To be able to automatically recognize behaviour, current plan or activity, we need models that explain a planning process including organization, rules of engagement and environment interaction, in a structured, generic and a flexible manner. In this chapter, we see the planning process from (hostile) agent's point of view, i.e. we consider this kind of problem in AI as the agent planning problem under uncertainty; see [6].

Agents plan and perform different tasks in an environment. There are also different hierarchies of agents in the information fusion process, like battalion, company, platoon etc. We used as our basis a generic model of command and control presented in [8]. We added some interesting aspects and subtracted other details which were of less interest for this kind of study, see Figure 2.

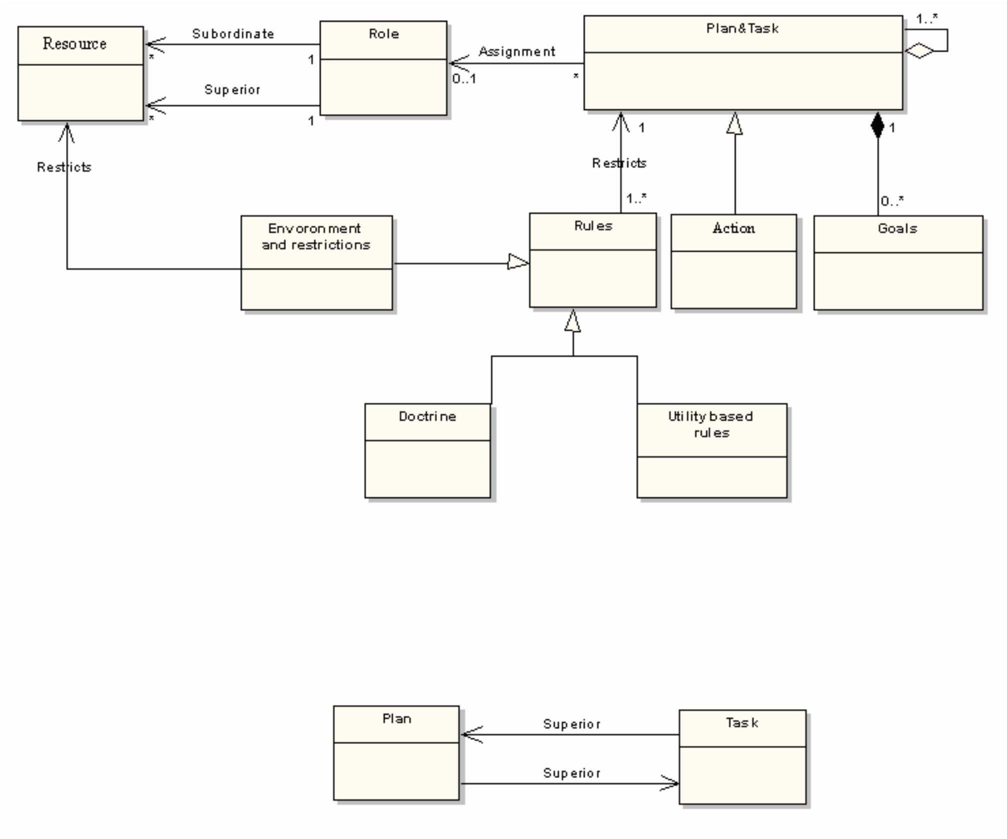


Figure 2. Planning, doctrine and environment

As we see in Figure 2, environment rules and doctrine rules are subsets of more general rules in an agent planning problem. Utility-based rules represent all rules that are not described in manuals but are (frequently) used. We say that utility-based rules are more specific for a particular situation, i.e. they are explicit and do not oblige fully to follow organizational (doctrine) restrictions. Additionally, some military or paramilitary organisations lack known doctrine rules. On the other hand doctrine rules are more general and a subset of doctrine rules is expected to be applied considering key factors such as environment, type of opponent, resources and superior goals. Plan and task are assigned to the role which can be for example a commander of a military unit. In order to solve the task and execute the plan a role has to use resources. The role can be part of a larger plan and be subordinated to a resource, e.g. platoon member is subordinate to platoon.

2.5 Representation doctrines on combat tactical level

Our approach of knowledge modelling is bottom up. Here we use tank units to exemplify some of the doctrine rules at the combat tactical level.

Each single agent is constrained by its doctrines and (current) surrounding environment. Depending on environment it is more likely that an agent will use certain doctrines. Physical conditions such as top speed of the agent, visibility, cover, weather, carrying and maneuverability in certain type of a terrain restricts the agent and thereby our hypothesis space on what the agent is doing is pruned. E.g., the hypothesis, "tank (agent) is on the hill" could not be so likely due to slopes around the hill that are higher than a certain number of degrees, see Figure 3. Here we show some physical restriction on a tank agent. The slope higher than 30^{o} is hard to overcome for a tank. From Figure 3 we can derive a number of physical rules when modelling a representation of a tank agent.

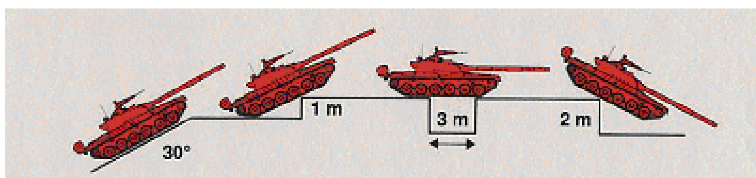


Figure 3. Passable terrain for tanks (source: [20])

Doctrines are used as the common understanding and working basis to coordinate actions within a group of agents. Such doctrines could be seen as guidelines or in some case even recipes on how to accomplish or deal with a certain type of task. Different kinds of ground troops have different mobility, different types of technical equipment and different formations (see Figure 4 and Figure 5). We see formation as a subset of behavioural (doctrinal) spatial patterns that can be modelled.

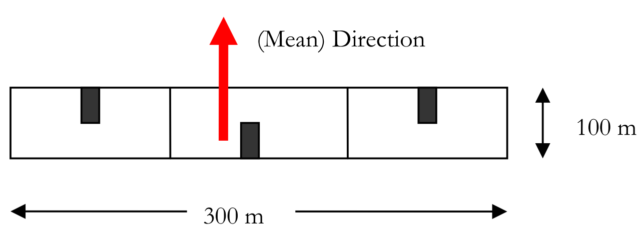


Figure 4. Battle Line

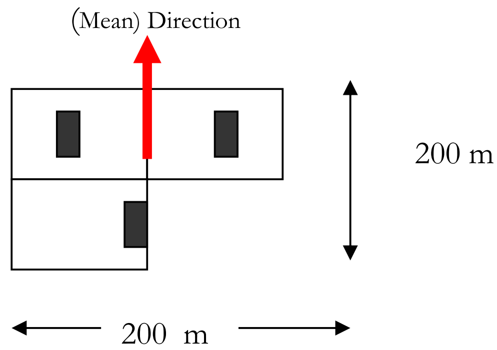


Figure 5. Battle Triangle

Both (spatial) patterns "Battle Line" and "Battle Triangle", differently constrained by the terrain, give a support for the hypothesis, plan alternative, attack. In Figure 6 we see a tank platoon model in UML where textual knowledge from [11] has been an important source. Some attributes of interest (aspects of interest) are presented.

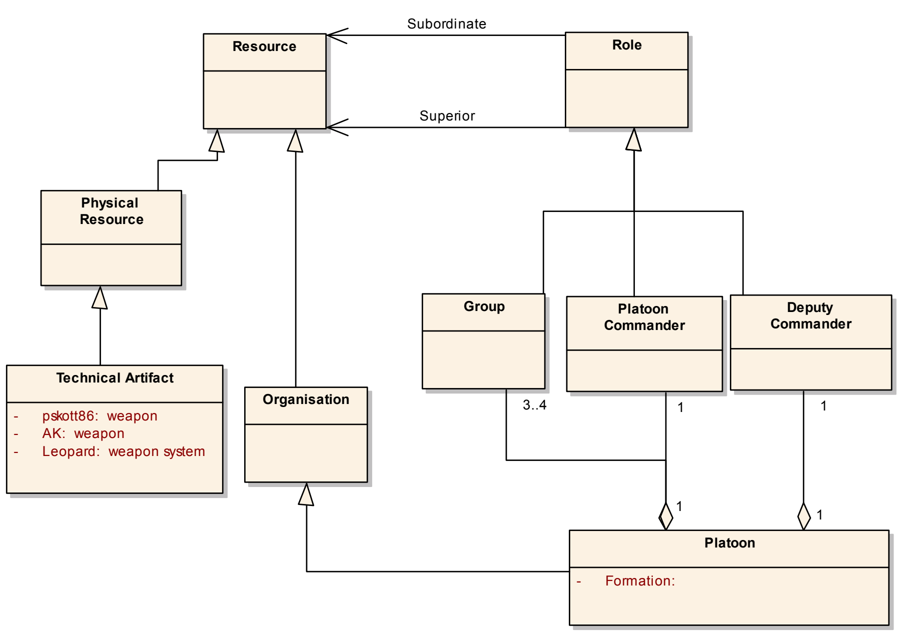


Figure 6. UML model of a platoon

As we see one platoon consists of tree or four groups, one platoon commander and one deputy commander. The platoon has an attribute formation with possible values column, battle column, battle triangle, battle line. Platoon is an organisation. The subset of an organisation is a class of technical artefact which contains attributes that correspond to the technical equipment of the platoon in this case.

2.6 Conclusions

The way of executing actions or plans is strongly related to the kind of environment where military forces, agents, are acting. Information on passable roads and terrain which gives better cover is important in the information fusion process in order to minimize the number of hypotheses generated by sensor data. Information about hostile formations, terrain and information from other sources gives us a clue to hostile force plans.

Chapter 3
Generic representation of Military Organisation and Military Behaviour

3.1 Knowledge representation of military units

The importance of developing generic models in command and control (C2) is increasing due to issues of co-ordination, co-operation, training, decision support etc. When modelling warfare, a plethora of factors has to be considered, [21]. In such complex problems the increasing need for classification of knowledge arises. We found it important to perform such a classification in a generic manner. The class models could then be reused with some modification and should be easy to update. Consequently, the modeling expert can concentrate on one part of the model at a time. In other words, one generic model of a military organisation and military behaviour can be reused for modelling different doctrines and for different purposes by using a well-known modelling technique.

In this Chapter we present a study of modelling military organisation and military behaviour in a generic manner, using two different knowledge representation techniques: the UML and Bayesian Networks, [22], (BN). The class diagram that is provided by UML is suited for representing military organisations whose structure is well-known, since military units and their interrelations can be represented as classes and interrelations between the classes. On the other hand, it is a much harder task to represent military organisations that are not well-known or military behaviour because of the uncertainty associated with them. Different behaviours are triggered in different environments using different doctrines, and the outcomes of the behaviours are uncertain. Due to complexity, time constraints and war friction, causal relations between different factors, which play an important role in warfare, may be uncertain. The purpose of this chapter is to highlight the need of interaction between UML and BN. Despite the fact that these techniques are very different and are used for different purposes, we propose an approach having generic UML modelling of military organisation and military behaviour as a first step towards modelling with BN.

3.2 UML doctrine models

Doctrines provide hints about how military tasks will be carried out. This means that some of the military behaviours can be classified. Given information about environment, force balance, opponent's position and other rules that have influence on military behaviour we can say that some behaviours are more probable to occur in some situations. UML has a very good expressive power for classification. Class diagrams in UML give very good overview but we cannot say anything about the probability that a given class, in this case a class describing a particular behaviour, will occur. E. g. we found it difficult to express how using UML a class representing frontal attack behaviour of some hostile military unit is likely to occur given the information that we are close to the enemy and the fact that visibility is good. In some cases certain classes are irrelevant and in other cases they are important.

To describe a class model in UML we first identify interesting classes. After performing this step we describe relations between them. Consequently, we make a generic structure that can be used for implementation for different purposes. Relations between attributes of different classes cannot be represented in UML class diagrams. Instead, in a UML class diagram we specify relations between different classes. On the other hand, the advantage is that the principle of encapsulation makes it possible to build implementations that have parts which are more autonomous, objects in UML. In BN, instead of attributes we have variables.

Figure 6 shows a UML model of a platoon. The *Platoon* has an attribute *Formation* with four possible values: "Lead", "Battle Line", "Stepped Formation" and "Battle Triangle". This variable, attribute in UML, will be represented in our BN model with corresponding values. In the similar manner as in Figure 6 we show in Figure 7 a company model. This model also represents the relation between company class and platoon class hence obtaining a hierarchical representation.

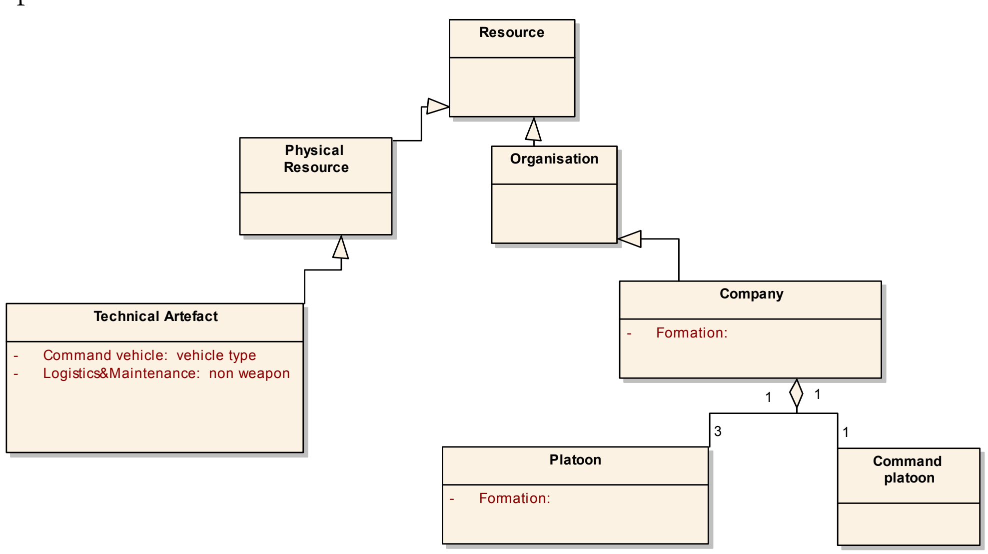


Figure 7: Company description with UML

It is not enough when modelling military doctrines to describe relations between different units, their roles, which resources they are part of, and which resources are put to their disposal. Military behaviour is however an important part of doctrines that is not part of the model. In concrete situations there is a list of the military behaviours/plans to be executed, see Figure 2.

Part of the model in Figure 2 is also the environment, which plays an important role when making plans. It is regarded by military commanders both as opportunity and as restriction to execution of their plans. Information about the opponent is also important when making own plans. However, representation of some "generic" opponent is not performed in our UML diagrams, although it was modelled with our BN model of a particular hostile tank company.

3.3 Bayesian Networks (BN)

In general, when modelling warfare, we have to deal with uncertainties. Prediction, fusion of the uncertain information, war friction, enemy courses of action etc., are examples of where a high degree of uncertainty is involved. Management of uncertainty is an issue related to uncertain, contradictory and incomplete information. Approaches to uncertainty management can be grouped, roughly, into three groups:

1. numerical (quantitative) approaches
2. non-numerical (qualitative) approaches
3. Hybrid approaches

BN are a hybrid approach that is both quantitative but has also qualitative meaning, see [23]. In our thesis we use both aspects of BN. However, in this section we focus primarily on BN expressiveness as qualitative method. BN is a statistical modeling method used to represent uncertain causal relations between different statistical variables. By using BN methodology it is possible to deal with uncertainty in a uniform and scientifically correct manner. The methodology has several potential areas of application within the IF and intelligence domain, for instance hypothesis management [24], [25], [26]; detecting threatening behaviours by insiders [27], antiterrorism risk management [28] or probabilistic assessment of homeland terrorist threats [29]. In Chapter 4, we use BN as key methodology for tactical plan recognition.

The graphical representation of BN is different from that of UML and uses nodes and arcs representation. The BN is thus a suitably labelled directed graph. Only one kind of relation between variables is described. This kind of relation is also called "influence relation" or "uncertain causal relation".

Each node represents a variable that can be either discrete or continuous. Variables and their states are represented by conditional probability distributions also called subjective probabilities. Bayesian Networks are also called *belief networks* since they describe our belief about the state of the variables. An advantage of the BN is that our knowledge is implemented in a piecewise manner. We only have to "explain" how a particular node depends on its parents.

According to [22] the formal definition of BN is:

- A set of variables and a set of directed edges connecting variables
- Each variable has a finite set of mutually exclusive states
- The variables together with the directed edges form directed acyclic graph (DAG) with variables as nodes
- To each variable A with parents B1 .. Bn there is attached a conditional probability table P(A| B1 .. Bn)

Mathematically expressed the BN defines a joint probability distribution over stochastic variables **X**:

$$P(X_1..X_n) = \prod_{i=1}^{n} P(X_i \mid par(X_i))$$

Where n is the number of the nodes in the network and X_i represents a stochastic variable no. i of the BN. The characteristic feature of a distribution defined by a BN is that the distribution of a variable X_i is conditioned only on its immediate (local) ancestors values, i. e. its parents par(X_i).

When we describe a time-dependent BN we speak about Dynamic Bayesian Networks (DBN). A DBN consists of several layers of BN with the same structure. The additional influences in DBN are the variables of the previous step(s) that make influence in variables for future step(s). Note that the term "dynamic" means that we are modelling a dynamic system, not that the network changes over time [30]. The variable values change over time but the network topology remains same.

3.4 A Hostile Company Bayesian Network Model Example

In military applications the issue is how to recognise certain military behaviours of the enemy.

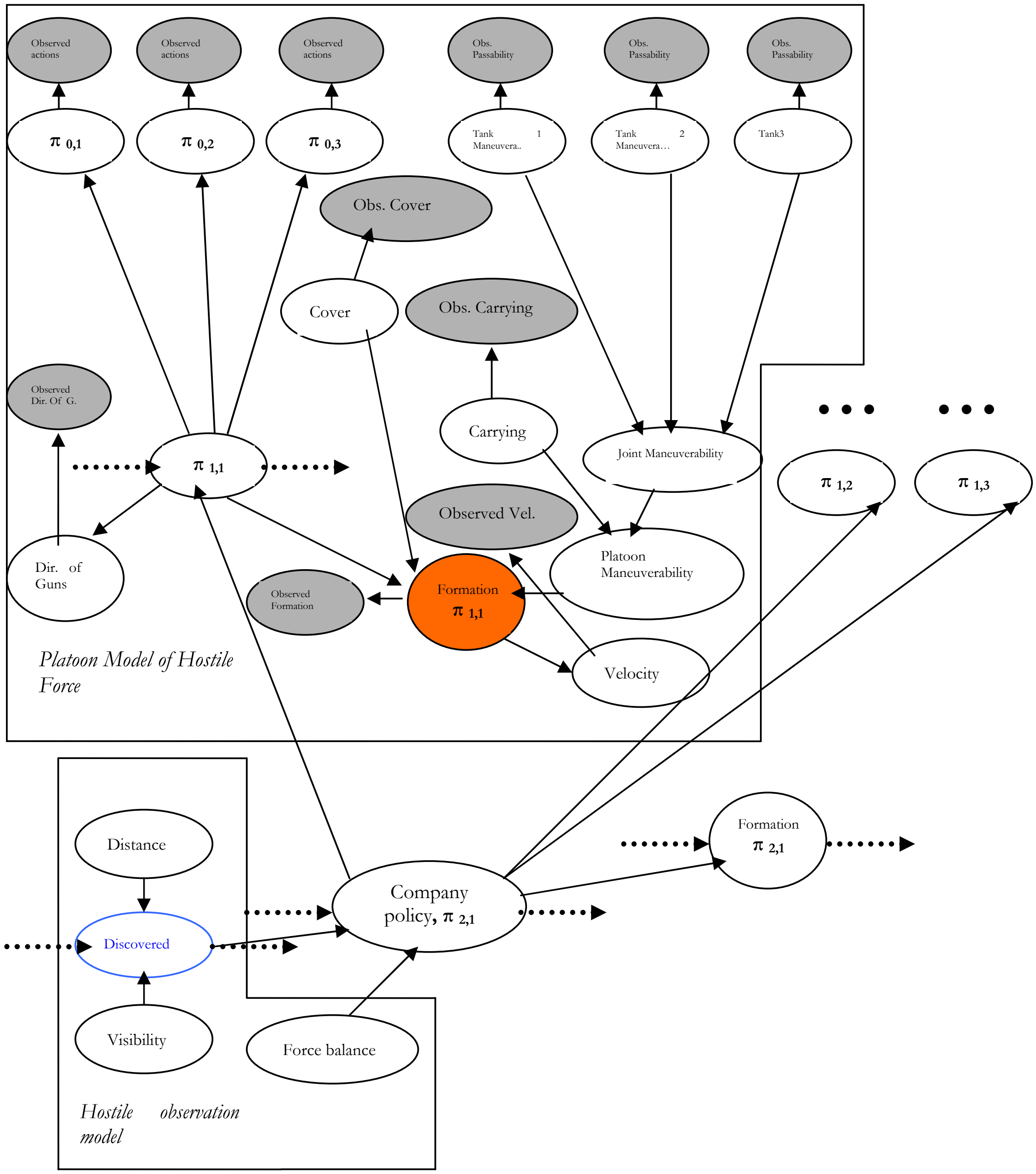


Figure 8. A BN Company Behaviour and Organization Model

Using the movement pattern, speed, distance, visibility, maneuverability distance to presumptive target etc., it might be possible to fuse the acquired knowledge about the enemy and use it in plan recognition. The advantage would be that military commanders, having better knowledge about the enemy's intentions, will be able to act earlier. The ability to act preventively increases as well.

As the first step in making a company BN, see Figure 8, we make a BN of a single hostile platoon. We specify variables in the graphical diagram, see Figure 8. After that, the causal relationships between variables are specified. Finally, we define conditional probabilities to "explain" how a certain variable's values depends on its parents values. E. g. we previously mentioned the variable Formation, see Section 3. 1, that may have the following values: Lead, Battle line, Stepped Formation and Battle Triangle. We define a conditional probability distribution over all possible combinations of these values and connect it to the platoon behaviour node.

After building a platoon model for plan recognition we define a company model that consists of three platoons. In this case we intended to define a platoon class with three instances. But modelling with classical BN does not support this kind of approach. Instead we had to perform a cut and paste process and when we wanted to change the model of a platoon we had to change it in all the three instances.

3.5 Discussion

The structure of BNs explains the model in a qualitative way, see [24]. Also, the results may be used for comparing hypotheses instead of expressing how probable they are. DBN are used in this implementation to represent our knowledge that is built-up in a piecewise manner. By using this kind of approach we obtain a better overview. The knowledge is transparent and the black box concept is avoided. Our model is still incomplete in the sense that we do not incorporate the association and identification problems.

However, BN seems to be a reasonable choice for representing uncertain military behaviour as well as uncertain military organizations, since this method combines uncertainty and a priori knowledge in a homogeneous way. We can compare those models and facilitate the verifying process. As result we get a more reliable BN and the modelling time decreases.

The important issue is how to build a BN from the UML class diagram. As a first step we create a BN representing a military unit, a company model in our case. The structure of the UML military unit and planning model facilitated the work of modelling (D)BN representing a hostile company but no formalism has yet been applied.

When we implemented the BN we realised that we cannot use the principle of reuse/generalisation more than copy and paste of the BN fragments. Therefore we need a structured framework/ontology of Bayesian network fragments for representing opponent behaviours. An interesting approach is presented in [31] and this is probable basis for further work on this problem.

Chapter 4
Multi-Agent Plan Recognition

4.1 Introduction to Plan Recognition

Plan recognition is the term generally given to the process of inferring an agent's intentions from its actions, see also [32, 33]. The representation that plan recognition offers is a rich and highly interrelated description that explains an aspect of agent's/agents' state and predicts goals and future actions of the agents.

Recognition of plans can be classified in different ways. One of these ways are *intended* and *keyhole* recognition, see [33]. *Intended* plan recognition is one of the classifications where an agent deliberately structures its actions to make his intentions clear to other, friendly agents; e.g. coordination of agents by plan recognition [34]. Here, we deal with hostile agents that try to hide their plans as long as possible (keyhole recognition).

We see plan recognition as the process of deriving hypotheses about agent's actions.

The 1990's advance in probabilistic methods in AI initiated research in stochastic plan recognition [39], [40], [41]. Previous approaches took only *one* hypothesis as the guess on agents plans [35], [36], [37], [38]. Such unilateral approaches may not produce reliable results in applications where several types of uncertainties exist. To achieve improved reliability a user should be offered more than one hypothesis by plan recognition. Employing probabilistic methods in AI has made it possible for plan recognition to support quantitative decision making and handle the problem of grounding belief in sensed experience, see [32].

Our application area is in the military domain for ground forces. "Battle space is an abstract notion that includes not only the physical geography of a conflict but also the plans, goals, resources and activities of all combatants prior to, and during, a battle and during the activities leading to the battle" [42]. Our plan recognition model deals with:

- Knowledge representation in statistical manner
- Uncertain observations
- Imprecise prior
- Terrain representation

Here we represent our prior knowledge in a statistical manner by using DBN [30]. Due to fog of the war and sensor capabilities, observations are assumed to be uncertain. This uncertainty is represented by our observation model. Moreover, there exists uncertainty if the outcomes of actions of the agent reflect its true intent (war frictions). Knowledge about agents is imprecise and suffers from linguistic vagueness. We found valuable the concept of using fuzzy set theory [43] where we model imprecise knowledge and combine it with sensor data making it situation (context) dependent. The result from fuzzy membership functions is entered into a DBN. Final problem that we deal with here is the terrain representation. It can be difficult to represent terrain for plan recognition in a tractable manner. Instead of representation of each terrain part with a Bayesian node (as in [39]) we use fuzzy membership functions to translate terrain data to contextual (relevant) information for plan recognition. This enables reusability and a generic approach for plan recognition.

Here, we show how our knowledge about an agent (opponent) or group of agents can be represented and their plans recognised by using combination of fuzzy membership functions and DBN. In order to focus on plan recognition, in this work we do not deal with the classical identification and association problems that are primary functions on level one of the JDL model. The stochastic nature of plans is derived from the fact that we do not have full knowledge about the enemy and his actions. This implies that military commanders should not only pay attention to the plan with the highest probability but to all plans that have a significant probability to be in use. The automatic process of plan recognition is performed for each new observation. The process is on-line and involves many agents (units) acting together. Thus we introduce *on-line multi-agent stochastic plan recognition.* However, in [44] we used the word “policy” instead of the word “plan” because we followed notation of the [39].

In military applications the issue is how to recognise certain military behaviours of the opponent. Using the movement pattern, speed, weather, terrain, distance to presumptive target (expected driving time), etc., it might be possible to fuse the acquired knowledge about the enemy and use it in plan recognition. The advantage would be that military commanders having better knowledge about the enemy’s intentions will be able to act earlier. The ability to act preventively increases as well. In this chapter we claim that by using our knowledge about hostile force doctrines and fusing this knowledge with sensor information we can recognise certain military behaviours of the hostile force. Our aim is to demonstrate the statement above in a *proof-of-concept model.* More concrete, we have implemented a hostile company multi-agent model, on three decision levels. This work uses a DBN model to represent our beliefs about hostile force units.

4.2 Soft computing method for plan recognition: DBN and Fuzzy membership functions

On-line multi-agent stochastic plan recognition, introduced in [44], aims to detect which plans an agent or group of agents are executing by observing the agents' actions and by using *a priori* knowledge about the agents in a noisy environment. Methodology for plan recognition belongs to weak/soft computing in computer science [45].

Soft computing is a type of calculations that is “tolerant” (unlike conventional computing) to incompleteness, uncertainty and approximation. We see our generic model for plan recognition as a soft computing method that combines Bayesian statistics and fuzzy membership functions. The latter has responsibility to connect observation data to DBN by entering context relevant evidence.

Plan recognition can be viewed as a classification problem: which plan is being executed?. Therefore techniques as neural networks could be applied. However, the use of symbolic methods is preferred where we can use abstract representation of intentions. Moreover, (fortunately) wars do not occur very often and the subjective knowledge becomes a more informative source than historical data, so symbolic representation is probably more suitable.

Here, we propose use of DBN as the base for plan recognition reasoning (inference). There are several reasons to use DBN. One of them is that it represents uncertainty in a structured and piecewise manner. By representing knowledge with a BN we can define large probability distributions. The second reason is that BN provides a transparent knowledge representation model. This implies that we are able to modify a part of the DBN, changing a direction or changing a local distribution, without being forced to change all nodes. This facilitates knowledge reuse. Finally, the DBN provides a qualitative knowledge representation [25].

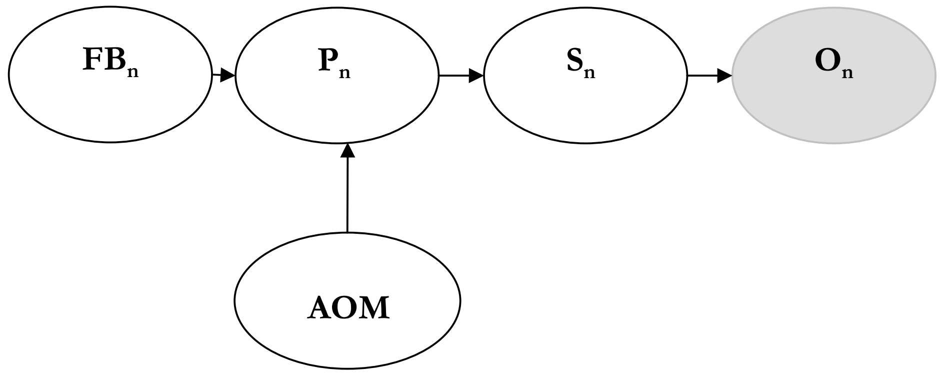
Doctrine (behaviour) knowledge is usually described in textual (free format) form. Linguistic impreciseness provides the elasticity to conserve a family of models capable of capturing the essence of the problem. However, representation of imprecise knowledge in a computer understandable manner may turn out to be a difficult process. Therefore when modeling a priori knowledge from textual documents it is hard to describe all aspects of knowledge properties.

Fuzzy membership functions have turned out to be a useful method to transform sensor data to classes based on incomplete, empirical knowledge even in other applications [46]. The parameter space in fuzzy set theory is vague which enables context based representation. To represent (empirical) behavioural a priori knowledge that is imprecise we use a family of fuzzy membership functions also in our plan recognition model. In fuzzy logic we are imprecise and can express even to which *degree* an attribute has its property, i.e. belongs to certain class. Those functions that express such degree are called fuzzy membership functions. E.g. a person either is long or not in "crisp" logic but in fuzzy set theory it can belong to the class long to certain degree. More advanced representations of fuzzy membership functions deal with problems where an attribute has more than one property. Such fuzzy functions are designated as family of fuzzy membership functions, i.e. an attribute property can belong at same time in various degrees to related classes. We see fuzzy membership functions as a tool to translate sensor data to evidence for DBN. This translation of context independent (sensor) data to context dependent data (information) with respect to incompleteness is the main advantage of fuzzy membership functions, for our purpose. Corresponding classes are represented as states in DBN's nodes.

4.3 Description of plan recognition model

We want to answer by plan recognition the question what an agent or group of agents is doing. In Figure 9, plan node (Pn) (hypothesis node) represents our belief about what a certain agent on decision level *n* is doing. The hypothesis space is spanned by a priori knowledge about the agent. This space contains discrete plan alternatives specific for decision level n. By abstraction level, n, we mean the level of discernment interesting for the current user (military commander) and/or the decision level in the opponent's structure.

Given a certain plan alternative the knowledge what we are expected to see, i.e. which states we are expected to observe is encoded. That is the reason why we say that plans or certain behaviours cause state change (S_n). E.g. it is possible to represent support, knowledge, to the hypothesis "agent is angry" if this agent is screaming and throwing things around, i.e. causes change of state. A tactical military example could be that alternative attack causes certain (generic) changes in (agent's) state such as regrouping into certain patterns.



Abbreviations list	
Abbreviation	**Explanation**
N	Abstraction/decision level
P_n	Plan node containing (discrete) alternatives (attack, march, …)
S_n	State node (terrain, visibility, doctrinal nodes such as formation)
O_n	Observation node contains observations of state nodes
FB_n	Force balance node with states "Stronger" or "Weaker"
AOM	Agent observation model that partly models of agent's observation's capability

Figure 9. Generic Plan Recognition Model

In DBN we use conditional probabilities. Our a priori knowledge is the knowledge what is the most probable state given plan alternatives P(State|Plans). For tactical plan recognition purpose we use knowledge about temporal, spatial and pattern properties. Doctrine manuals, terrain properties, weather conditions, formations, and expected time to impact are nodes supposed to be represented in DBN and are classed as State and State-Plan a priori knowledge.

Force Balance node (FB_n) takes capabilities between opponents into account. Its states are {Stronger, Weaker}; it is a greater probability that an agent will attack if it is stronger than if it is weaker. Expected losses and how much an agent would gain by executing certain plans is a trade-off that has been modelled.

An agent plans are also under strong influence of discovery. To trigger attack behaviour the agent (opponent) has first to discover our forces. Therefore in our implementation we also build a limited model that represents hostile force ability to observe us and we call it enemy (agent) observation model (AOM). In our implemented model we assume that the discovery probability depends only on the distance from the enemy to our forces and visibility.

Some states are easier to observe than some other states. With our state-observation relation we model the degree of trust in observations given states.

P(Observations|States). Observations can be positions but also aggregated containing patterns. As an example, we can state that one type of unit formation has lower probability to occur when maneuverability is bad, see also "The Movement-Analysis Challenge Problem" in [42]. Fuzzy functions take as input enemy's tank positions, position of our forces and terrain data. The outputs of those functions are the probability distributions over the variables *Cover, Distance, Tank Plan,* (Platoon) and *Tank Maneuverability*. This operation is performed for each tank. The uncertainty about the observations takes into account how the information of the pixels representing the terrain of interest should be weighted. In our example we used a simple approximation of the rectangular distribution over each position. More advanced representations of fuzzy membership functions deal with problems where an attribute has more than one related property that depends on each other. Such fuzzy functions are designated as family of fuzzy membership functions, i.e. an attribute property can belong at same time in various degrees to related classes. E.g. an attribute called "tank platoon's formation type" can belong to classes: "marching formation", "stepped formation" and "battle line formation" depending on formation angle.

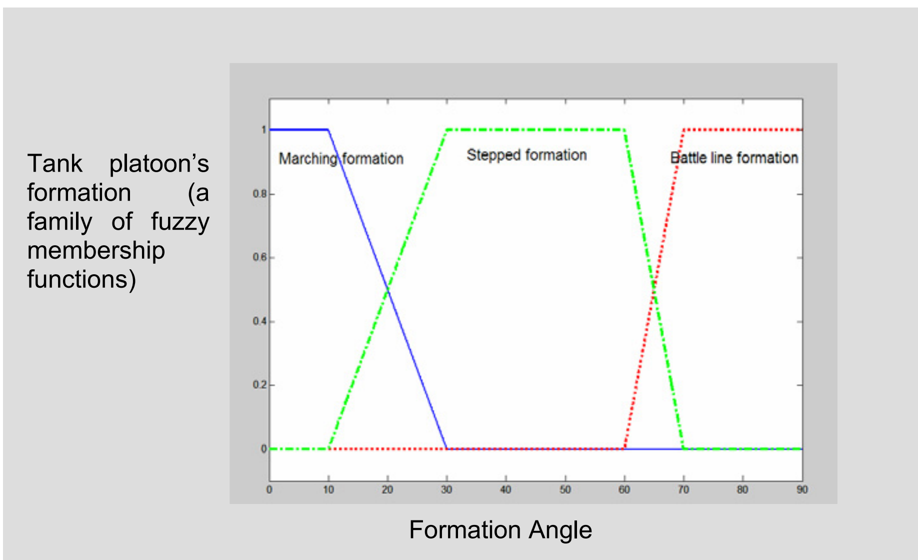


Figure 10. Formation fuzzy function

In Figure 10, we see a family of fuzzy membership functions of attribute **tank platoon formation**. To the left we see a fuzzy function representing formation type "marching formation". In the middle we see a fuzzy function of "stepped formation" and on the right we see a fuzzy function of "battle line formation". Formation angle relative to its moving direction is calculated by using observations of tank positions. For each angle we obtain a distribution of membership degrees of belonging to a certain formation. This distribution is entered to a Bayesian node *Formation* and it is interpreted as subjective probability distribution of states, i.e. each state corresponds to a fuzzy membership function.

4.4 Simplified representation of the implemented DBN for plan recognition

We have implemented a hostile company model, a multi-agent model, in MATLAB using K. Murphy's BN package [47]. The DBN representation of a hostile company's organization is visualised in Figure 11 and a simplified DBN model description is represented in Figure 12.

Our implemented model of a hostile tank company is hierarchical, corresponding to a hierarchical plan structure. The company consists of three tank platoons, each platoon containing three tanks. For each level there is a certain set of *plans* that are influenced by the higher level plans. The simplest plans, the *atoms*, consist only of a set of *actions*. In this example the simplest plan is the tank (group) plan. More complex plans consist of other plans, also referred to as *sub-plans*, or a mixture of sub-plans and actions. Higher level plans influence lower level plans down to their action atoms. The plan hierarchy with its decision levels, is represented by the DBN. Plan for each agent (hostile unit) is represented as a DBN node. The simplest plan is on tank (group) level, $n = 0$.

Tank *i*'s plan variable, $\pi_{n=0,i=1..3}$, has the following discrete states:

$\pi_{o,i}$ = <agent *i* is moving in the direction of our own force, agent *i* is moving in the direction opposite to our own force, agent *i* is moving in neutral direction>

On the next level we have the plan of the tank platoon *i* at level $n = 1$:

$\pi_{n=1,i}$:= <attack, defence, reconnaissance, march>

Finally on the top level we have the plan of the tank company at level $n = 2$:

$\pi_{n=2,i}$:= <frontal attack, frontal attack and flange attack, defence, delay battle, march>

In Figure 11 we show a Bayesian network representing the plan hierarchy model of a hostile company.

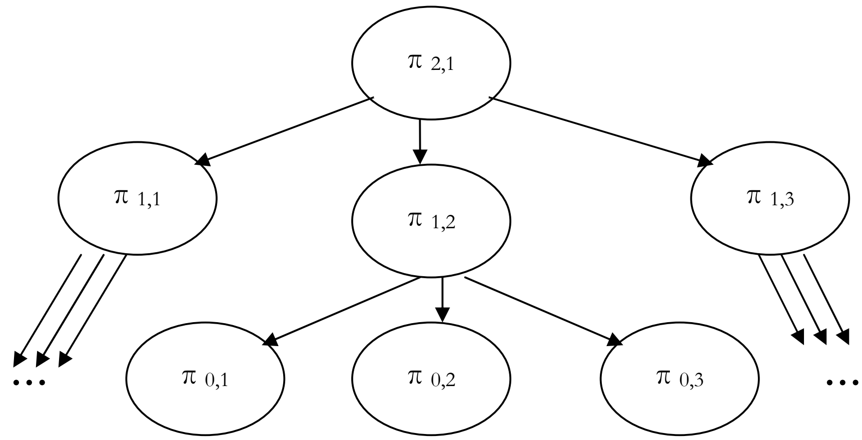


Figure 11. Plan hierarchy (Company organization model)

An additional reason for using this modelling approach is that this model follows military hierarchy; commanders give orders to their subordinates who are superior at the next level. Moreover, a DBN with a proper description offers flexibility beyond hierarchical modeling in a consistent manner.

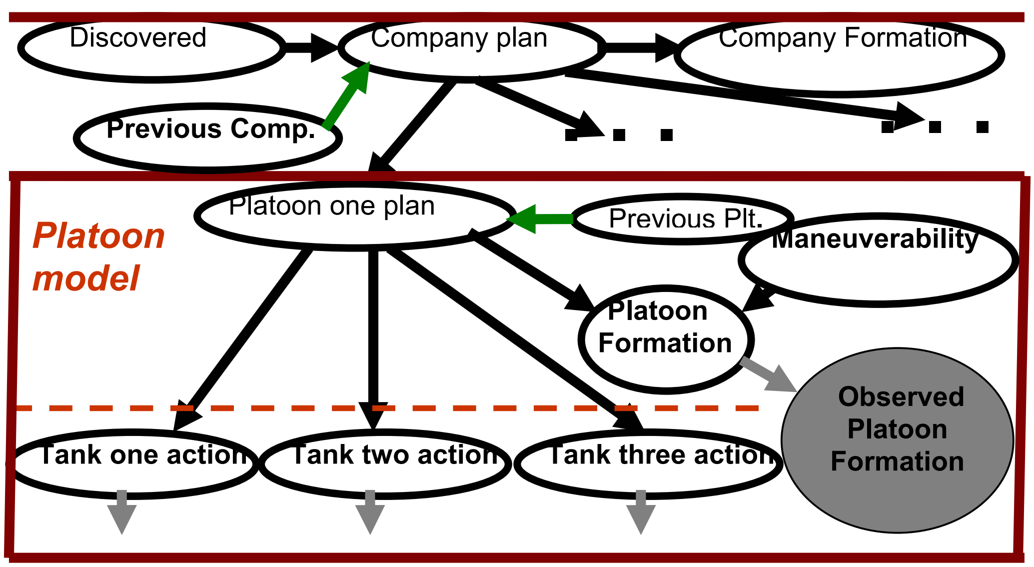


Figure 12. Company Model: simplified version

In Figure 12 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 doctrine manuals, when the enemy has the intention to attack it usually attacks in battle line formation. When transporting to a certain destination the enemy transports in "march formation". There are many reasons why the enemy may not use a certain type of formation. One of the factors that have influence on building a formation is the current terrain. When the platoon maneuverability is bad the enemy will not have the opportunity to attack in battle line and the probability of the formation type battle triangle increases. The probability that the enemy is performing reconnaissance increases when he is moving in a stepped formation.

According to our model, the variable *Observed Formation,* grey node in Figure 12*,* depends on the actual formation. Due to environment, uncertain observations and possible agent's coordination problems we are not always able to observe its formation pattern. It is the rule rather than the exception that the enemy's formations do not follow the same geometrical properties as described in the doctrine manuals. Therefore we implemented a fuzzy function in MATLAB that 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*.

To achieve tactical superiority on the battlefield, tanks maneuver very often. That implies for our modeling approach, that we do not connect nodes representing tank plans over time, see the nodes *Previous company* and *platoon plans*. We connect platoon plans and company plan 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.

4.5 Scenarios, simulation and results

We have visualized the movement of a simulated hostile company unit. The hostile company unit consists of three platoons. Each platoon consists of three tanks. We use a scenario for this simulation. In the beginning of the scenario, the tanks are marching in a neutral direction. The company is not spread over a vast area and the distance to our forces is initially large. Fuzzy membership functions take as input agent's (enemy's tank) positions, position of our forces and terrain data. The outputs of those functions are the probability distributions over the variables *Cover, Distance, Tank Policy,* (Platoon) *Formation, Company Formation* and *Tank Maneuverability*. This operation is performed for each agent (tank). The uncertainty about the observations takes into account how the information of the pixels of interest should be weighted. In our example we used a simple approximation of the rectangular distribution over each position.

In time step two observations about current positions and direction of the enemy arrive, see Figure 13. We performed the computation of plans for the company and the platoons on-line and the results for this step are documented in Table 1.

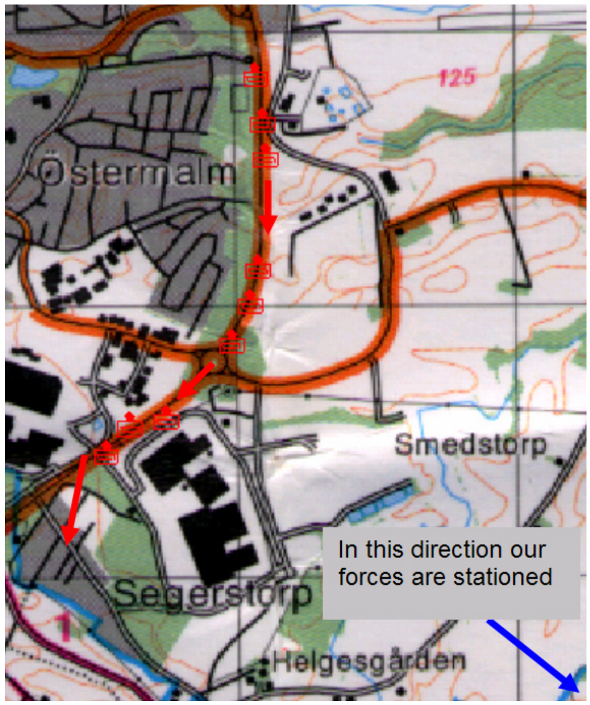


Figure 13. The Enemy Company is Approaching (Red boxes symbols)

Table 1. Company, platoon plans and values

Most probable and least probable plans (a snapshot of situation showed in Figure 13)		
	Most probable states *(Probability %)*	**Least probable *(Probability %)***
Company Plan	March *(47 %)* Flange attack *(21 %)*	Defence *(3 %)*
Platoon One Plan	March *(80 %)*	Defence *(5 %)*
Platoon Two Plan	March *(70 %)*	Defence *(7 %)* Attack *(8 %)*
Platoon Three Plan	Attack *(90 %)*	Defence *(0.5 %)* March *(1 %)*

The probability that the enemy company has discovered us is 33 %. The most probable state of platoon three, according to Table 1, is attack. The explanation is that this platoon is approaching in the direction towards us. However, the most probable state of the company plan is march. This is achieved by weighting with other nodes including the two other platoon plans. It is usually difficult to infer intentions of a single tank if this unit is not put in a greater context such as platoon or company.

After some time observations are received. The enemy begins to approach and then passes by. The movement and formation pattern indicates that enemy has not discovered us although the distance is short. Thus, the most probable hypothesis is that the hostile force is performing march (with probability 98 %). The most probable platoon two plan is march and is 97 % in this case. But for platoon three there is a probability of 44 % that the platoon will attack us and only 27 % probability that this platoon is marching. The probability that the enemy company has discovered us in this time step (time step = 6) is only 7 %.

4.6 Contributions

In this work we extended the case of the single agent [39] to the on-line multi-agent stochastic plan recognition problem using a network structure. By using knowledge of agents' interrelations we can create a plan structure that is compatible with that of a hostile military organisation. Using this approach we make use of existing knowledge about the military organisation and thereby strongly reduce the size of the hypothesis space. In this way we are able to bring down the problem complexity to a level that is tractable. Also, by using statistical models in plan recognition we are able to deal with uncertainty in a consistent way. For the information fusion purpose, we show with our plan recognition model that it is possible to integrate the pre-processed uncertain dynamical sensor data such as the enemy position and combine this knowledge with terrain data and uncertain a priori knowledge such as the doctrine knowledge to infer multi-agent plans in a robust and statistically sound manner.

Chapter 5
Plan Recognition and Sensor Management

In Chapter 4, we described plan recognition and here we present a *bridge* between plan recognition and the information acquisition process (IA) [48] that feed an IF process and thereby plan recognition with sensor data. Research work presented in this chapter is a result of cooperation with Ronnie Johansson whose main field of study is information acquisition for IF.

Plan recognition requires relevant and timely information to produce useful results. Sensor resources that are part of IA are typically limited. IA supplies IF with sensor data and cannot satisfy all information needs in all cases. To date (state) uncertainty in sensor observations has been a most important control parameter when controlling sensors and collecting data. Such, control does not take tactical importance, a kind of high-level information, into account. Here, we reuse high-level information obtained from plan recognition for focusing sensor resources on the most relevant tasks. By focusing sensor resources on the most relevant tasks we achieve improved predictive situation awareness.

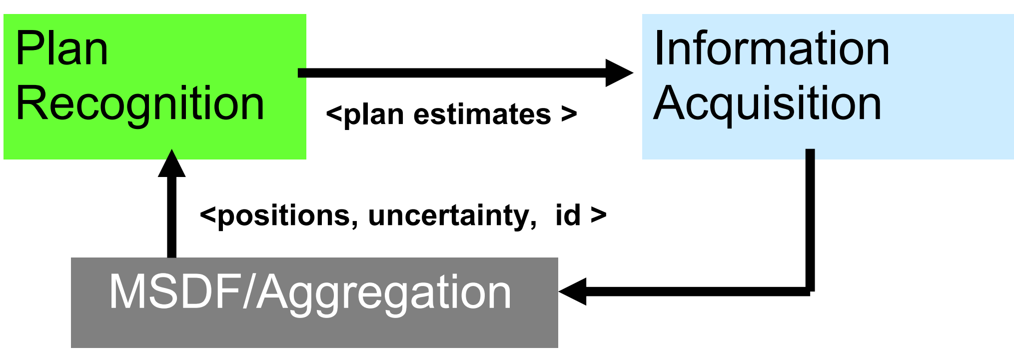


Figure 14. Sensor Management and Plan Recognition

Plan recognition produces plan estimates that are in IA used to prioritize sensor tasks, see Figure 14. Sensor management does its best given the guidelines it receives from plan recognition. New observations are made by the sensors which are fed back to the plan recognition process through multi-sensor data fusion and aggregation processes.

The main purpose of our cooperative research work, presented here, is to connect high-level information, in our case the outcome of plan recognition, to the control of the sensors that are intended to acquire most important information. Here, we introduce [49] and explore parts (aspects) of a framework, in an implementation [50], for expressing transition from information need to sensor control.

5.1 Framework for bridging the gap between high level information need and information acquisition

The general structure of the framework, depicted in Figure 15, involves two types of entities: *space* and *function*. The four space entities: *task origin*, *task*, *service* and *resource* are containers of structured information. The structure of information of each space entity should suit the intersecting function entities: *task creation and management*, *allocation scheme*, and *service management and resource allocation*.

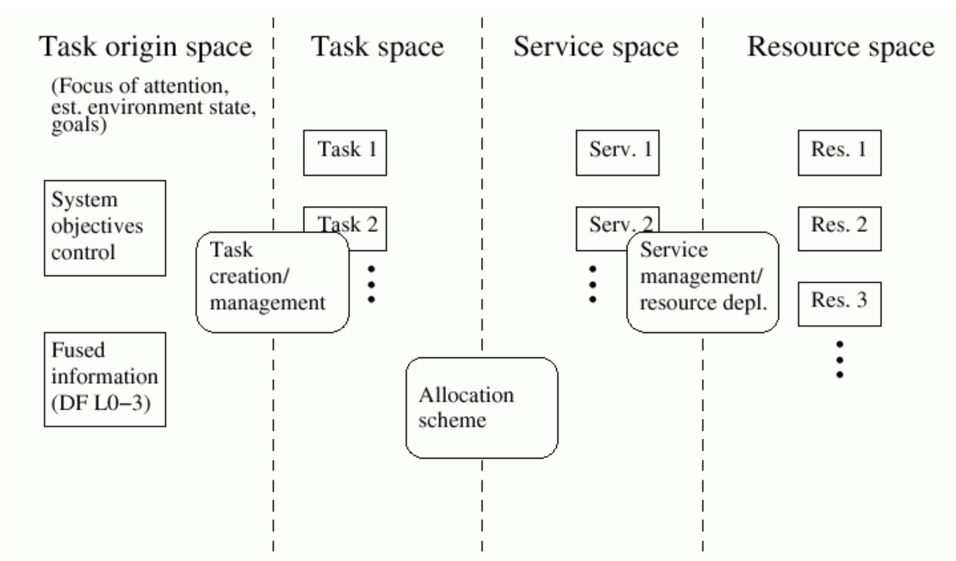


Figure 15. The Framework

The framework prescribes that information need (contained within the task origin space) is formulated as information tasks with assigned properties (e.g., priority or time horizon depending on what properties the system is designed to handle). Such tasks belong to the task space in our framework. The materialization of tasks to satisfy a certain information need could be the responsibility of the task creation and management function. The service space contains services that the sensors in the resource space (independently or jointly) can perform. The allocation scheme describes how tasks are connected to feasible services, i.e. services that are suitable for handling the tasks.

Plan recognition here belongs to "Fused information" box in the task origin space. Plan recognition produces estimates of agents' plans acting in the environment. We label this "high-level" information since it is interpretative and tries to provide an explanation. In contrast, "low-level" information typically originates directly from sensors and simply estimates observable properties of the environment (such as position, feature, etc.).

Plan recognition's results are in our framework used to prioritize sensor tasks. This is performed in the "Task creation/management" step of the framework. To automatically assign the priority of a task (e.g. monitoring/tracking an (hostile) agent) we use plan recognition results, estimates, with its variance/sensitivity. Additional advantage of our framework concept is that in such prioritization we do not need to take into account available resources at the "Task creation/management" step. Available resources are encapsulated by services that are an interface to sensor tasks. In the allocation scheme we take into account resources but via services. At this step we reprioritize sensor tasks concerning availability of services (joint or individual sensor resources capabilities). We propose bridging the gap between plan recognition, i.e. high-level information, and information acquisition as a stepwise process cued by our framework description.

However, the prioritization of tasks should not ultimately depend only on estimated plans. It should depend on information uncertainty as well. We propose an approach of measuring how sensitive inferred information is to changes of underlying information. In other words, we pose an issue: is the underlying information considering its uncertainty reliable and sufficient to infer robust a conclusion? If large changes in uncertainty have little effect on threat level then we can say that even large uncertainty is acceptable in this case.

On the other hand, there are cases where large dispersion in inferred results is obtained from information with low uncertainty. Consequently, we say that this kind of underlying information which for small changes causes significant changes in result is more sensitive. To measure sensitivity, we propose to use a method that samples and estimates the standard deviation of inferred plans. If it turns out that the estimations of plans for a particular unit is very sensitive to uncertainty of data for a unit in a particular situation then information about this unit should get higher priority.

Finally, we state that prioritization of tasks is conditioned on both activity (subsequently expected impact; further discussed in [51]) and information sensitivity. The solution to automatic task prioritization can be obtained by using some soft computing method, e.g., Fuzzy membership functions or BN, see [49].

5.2 Realisation of a Bridge between High-Level Information Need and Sensor Management

In this chapter we discuss an implementation of the framework. The bridge between sensor management and plan recognition presented here is that we reuse plan recognition estimates to prioritize sensor tasks.

Our aim was to demonstrate that our implementation gives reasonable results in a scenario, see Figure 16. Furthermore, interesting results are obtained in more complex scenarios. Here we consider an extensive geographic environment including two consumers (actors) located in the middle of the view (a1 and a2).

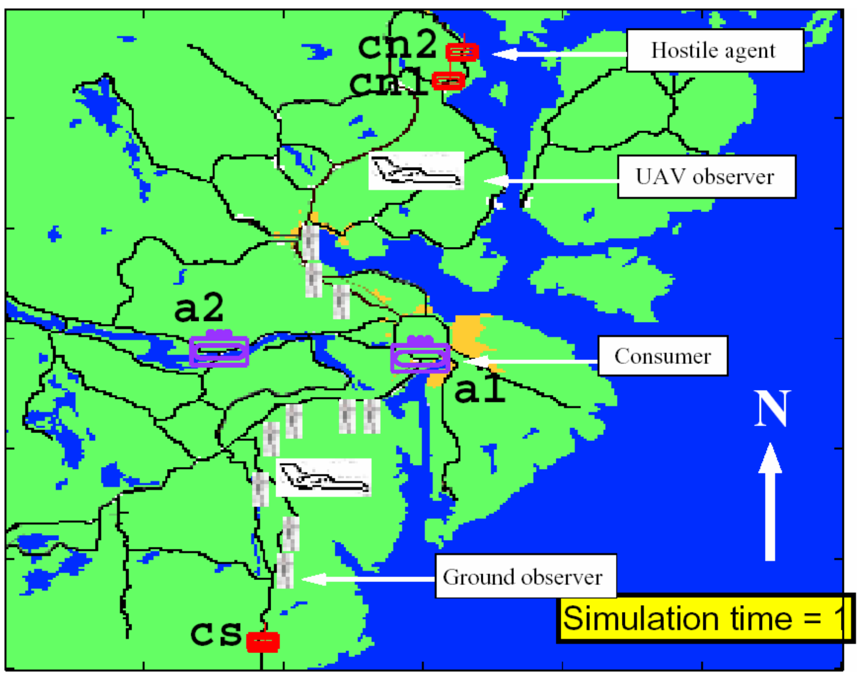


Figure 16. Decision Support Context

The purpose of plan recognition is here to support some *information consumers'* decision making. They perform plan recognition based on information about agent states. In the scenario there are nine (hostile) agents, i.e., *platoons*. Groups of three platoons belong to a *company*. There are two companies near the north perimeter (labelled CN1 and CN2 respectively) and one in the far south of the view (CS). There are two types of resources modeled. One type is the UAV observer which can travel quickly but can only give state estimates from a distance (to ensure its own security). The other one is the ground soldier who is limited in speed but who can hide himself close to the road and make comparatively precise state estimates of a passing vehicle. The IA has duties such as collecting measurements to track hostile agents, and to configure and engage sensors in tasks. The objectives of the consumers are to know as much as possible about plan recognition estimates of the hostile agents.

To achieve purposeful IA, task management prioritizes (orders) tasks. We introduce the notion of *threat* and the idea that the higher threat posed by an enemy unit the higher the priority of the corresponding task it should be. The threat calculation integrates estimated plans (*ep*). We introduce threat weights (w_j) whose magnitude is dependent on the danger (threat) corresponding to each plan alternative x_j. The probability for a plan alternative x_j is $p(x_j \mid obs)$ given observations. E.g. the weight corresponding to a plan alternative attack has greater magnitude than a weight for alternative march.

Hence,

$$ep = \sum_j w_j \cdot p(x_j \mid obs)$$

is a summarized threat value of the plan distribution. The threat estimate (T) becomes:

$$T = ep$$

We cannot be entirely satisfied with this calculation of threat estimation for task prioritization. It does not fully reflect the sensitivity caused by the uncertainty properties such as position of the enemy units. Ideally, we want to find the expected threat and threat variance of each enemy unit given estimated properties and uncertainties. In general, the expected threat and threat variance cannot be calculated analytically. We would therefore like to approximate these properties using Monte Carlo simulation. However, our current position uncertainty model is simply an uncertainty radius (*ur*) that grows with time when no observations are made. The model is unfortunately at present ignorant of terrain characteristics and sampling from it is computationally costly. A possibly feasible approach would be to let a terrain-aware particle filter [52] represent the position uncertainty. Sampling from the particles would yield more accurate estimates of expected threat and variance.

For prioritization of tasks, additional factors can be considered. The expected impact (*ei*) of the enemy unit attacking one of our units or essential resources could be explicitly represented. The priority could also depend on the time duration (*td*) before a particular enemy unit can engage in an operation against our resources; longer duration gives lower threat. We also motivate the use of *td* due to sensors limited velocity that may result in a considerable difference in time delay for different observations.

Hence in principle, the task priority becomes:

$$Priority = T + ur + ei + td$$

Finally, *focus of attention* for IA is maintained by the allocation scheme which has to consider both task priorities and the *availability of services*.

In Figure 17, we illustrate how our implementation changes its focus of attention to the three companies CS, CN1, and CN2, by changing tasks for sensors. The calculated task priority of the three companies varies during 35 simulation time steps. Initially, CS (the solid line in Fig. 17) is the greatest and it increases as long as the company has not been observed. The UAV sensor is accordingly allocated to CS. It is moving north along the road in the beginning of the scenario (while the companies in Rn remain stationary). After a short while (about time step 7), before the UAV has a chance to observe CS, CS is spotted by one of the ground observers in region. Since the uncertainty of the whereabouts of CS has been lowered, the priority decreases. At this point priorities of CN1 and CN2 (the dashed lines) exceed that of CS and the UAV starts to look for CN2 instead.

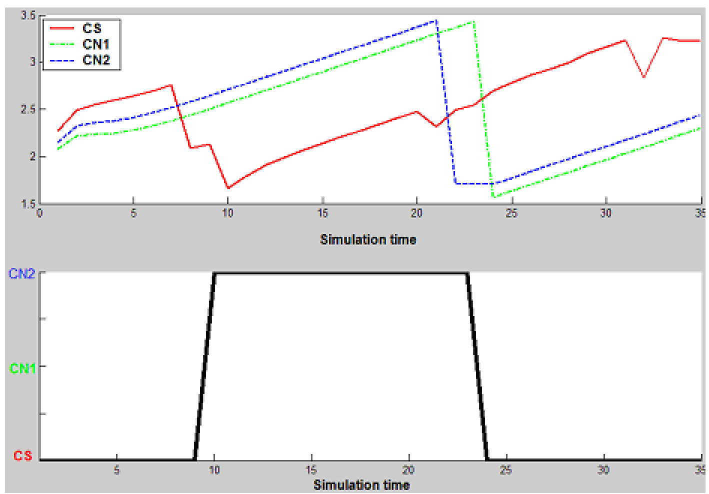


Figure 17. Task priorities and focus of attention

Around time step 20, the UAV observes CN2, but also CN1 which is in the vicinity. The threat levels of both CN1 and CN2 drop rapidly and CS has once again the highest threat level. The UAV changes its selected target back to CS as expected. This result suggests that the automatic management of sensors presented in this article agrees with an intuitive sensor control.

In this section, we introduce a definition of *plan estimate loss*. It is a quantitative measure of how well plan recognition estimates a particular, critical, plan alternative given a sensor configuration and a control method. In our realization we do not use different control methods we are able to vary the number of (start) sensor configurations. In our implementation the results of plan recognition are depend on a limited number of sensors(i) and tracking is performed by some sensor resource method M_r(i). In order to quantify how critical each plan alternative is we define the following penalty loss function. It assigns penalty measure $Pen(x_j)$ over plan space (χ) for each plan alternative (x_j).

Then we calculate $I(M_r(i), x_j)$, see Eq. 1, which is the area between plan estimate for the case of unlimited number of sensors $p_\infty(x_j)$ and for the case of limited number of sensors $p_i(x_j)$ in relevant surrounding. Finally, we define plan estimate loss during a *time period*: [t_start, t_end] as in Eq. 2:

$$I(M_r(i), x_j) = \int_{t\_start}^{t\_end} | p_\infty(x_j) - p_i(x_j) | \, dt \qquad \text{Eq. (1)}$$

$$Loss\ (i) = \sum_{xj \in \chi} Pen\ (x_j) \cdot I(M_r(i), x_j) \qquad \text{Eq.(2)}$$

In our experiment, we vary the number of sensors in region north. These sensors are of type "Markus" (ground troop soldier) and are assumed to observe objects (agents). We focus on one of the enemy companies in the north and estimate the probability that this agent (company) will attack the consumer (our force). We perform four simulations of 160 time steps. Each of them returns the probability for attack given a varying number of sensors. In the first simulation we assume that we are able to observe the enemy at all time steps. This is equivalent to using an infinite or sufficient amount number of sensors in the simulation. The attacking probability estimate for an infinite number of sensors is used as a reference when comparing to other attacking probability estimates with a limited number of sensors (observations).

In Figure 18, all attacking probabilities are equal while the enemy company is not moving. In other words, estimated position is the real position. The CN2 starts moving and we observe first divergence of the plan estimate for unlimited number of sensors (red line in Figure 18). In the case of three sensors we get a result that underestimates the attacking probability in a time interval (blue line), in other words the lack of (important) information in this case delays discovery of threat.

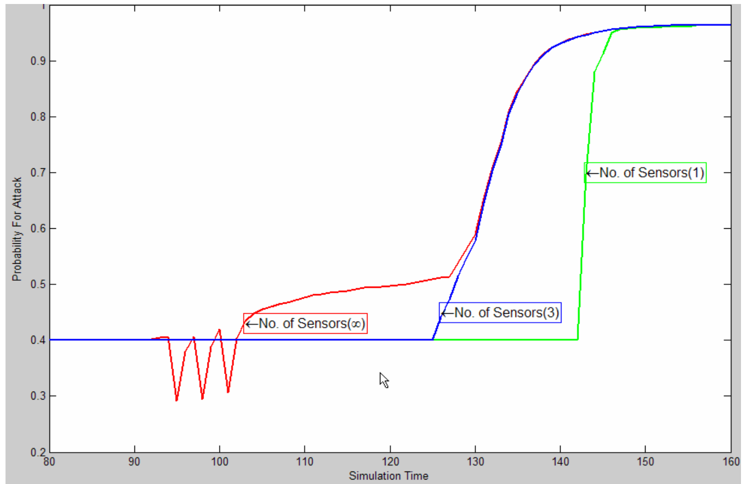


Figure 18. Attacking probability estimate over time given sensor configurations with one, three and infinitely many sensors

This work on bridge was a contribution that shows how high-level information (predictive state) can be used for a *proactive* multi-objective control in large-scale environments.

Chapter 6
Further Development

Our plan recognition models should be more flexible for partial changes. Additionally, considering that experts (knowledge and sensor estimate) could have significantly different opinions, a Robust Bayesian approach [53] could be considered as a further development. The result of the robust Bayesian recognition (recognition of plans according to different experts) could be a *family* of plan recognition estimates instead of *one* estimate distribution per agent.

To achieve better flexibility we propose research about suitability of Multi-entity Bayesian Networks (MEBN) for plan recognition. It is an extension of standard BN in their ability to encode repeated, parameterized argument structures called MEBN Fragments, [31]. In that manner we improve reusability and achieve better flexibility of plan recognition.

Furthermore, we propose an approach that takes agents' (simulated) capabilities into account. A *agent based simulation* that supports plan recognition with situation based knowledge, where CPTs are changed on-line, is one of the most probable future developments.

Bibliography

[1] © Network Inference 2004, "The Evolution of Integration Technology Towards an Enterprise Semantic Web", White House Conference 2004, SEMANTIC TECHNOLOGY SERIES, 2004, www.topquadrant.com/documents/sept8egov2004/ Jeff%20Pollock-Semantic-Tech-Evolution.ppt

[2] E. A. Smith, "Effects Based Operations: Applying Network Centric Warfare in Peace, Crisis, and War", ISBN: 1893723089, DoD-Ccrp, US, 2002.

[3] Marco Smedberg, "Om stridens Grunder", ISBN 91-7125-028-X, Page One Publishing AB, Stockholm, Sweden, 1994.

[4] Marco Smedberg, "Militär Ledning", M7743-858001, Historiska Media, Lund 2001.

[5] P. W. Phister, T. Busch, and I. G. Plonisch, "Joint synthetic battlespace: Cornerstone for predictive battlespace awareness", In *Proc. of the 8th Intl C2 Research and Technology Symposium*, 2003.

[6] S. J. Russell, P. Norvig, "Artificial Intelligence", ISBN 0-13-103805-2, Prentice-Hall, New Jersey 1995.

[7] R. A. Piccerillo and D. A. Brumbaugh, "Predictive battlespace awareness: Linking intelligence, surveillance and reconnaissance operations to effects based operations", In *Proc. of the C2 Research and Technology Symposium*, San Diego, US, 2004.

[8] 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.

[9] R. Suzić, "Knowledge representation, modelling of doctrines and information fusion", In *Proceedings of the CIMI conference*, Enköping, Sweden, May 20-22, 2003.

[10] R. Suzić, "Kunskapsrepresentation av doktriner och taktiskt uppträdande", Technical Report FOI-R--0865-SE, Swedish Defence Research Agency, Stockholm, Sweden, June 2003. (In Swedish)

[11] "BrigR A Strvplut Fu (förhandsutgåva)", M7741-122740, ATK, Sweden 2002. (In Swedish)

[12] K. Pallin, "Militärstrategisk doktrin", Bilaga 1 till 19 400:740754, Sweden, 2002.

[13] David L. Hall and James Llinas, editors, *Handbook of Multisensor Data Fusion*, chapter 2 Revisions to the (JDL) Data Fusion model by Alan N. Steinberg and Christopher L. Bowman, CRC Press, 2001.

[14] C. J. Matheus, M. M. Kokar, and K. Baclawski, "A Core Ontology for Situation Awareness". *In Proceedings of the Sixth International Conference on Information Fusion*, pages 545 –552, 2003.

[15] J. Cantwell, J. Schubert, P. Svensson, "Informationsfusion", FOA-R—00-01632-505—SE, Sweden, 2000.

[16] J. F. Sowa, "Knowledge Representation", ISBN, 0-534-94965-7, Brooks Cole Publishers Co, US, 2000.

[17] M. Eklöf, P. Hörling, P. Svan, R. Suzić, C. H. Yi," Information and Knowledge Management for Network Based Intelligence (NBI)", Swedish Defence Research Agency (FOI), User Report, FOI-R--1417--SE, Stockholm, Sweden, 2004.

[18] H. E. Eriksson and M. Penker, "UML Toolkit", ISBN 0-471-19161-2, John Wiley & Sons, US, 1998.

[19] M. Kokar, C. J. Matheus, Kenneth Baclawski, J. A. Letkowski, M. Hinman, J. Salerno, "Use Cases for Ontologies in Information Fusion", *In proceedings Fusion 2004*, Stockholm, Sweden.

[20] "Pansarkännedom steg 1, 2", Försvarsmedia, M7734-452543, Sweden, 1991.

[21] R. 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*, Antalya, Turkey, 2003.

[22] F. V. Jensen, "An introduction to Bayesian Networks", UCL press, London, ISBN: 1-85728-332-5, 1996.

[23] J. Doyle and Richmond , "Background to Qualitative Decision Theory", AI Magazine, American Association for Artificial Intelligence, US, 1999.

[24] K. B. Laskey, S.M. Mahoney and E. Wright "Hypothesis Management in Situation-Specific Network Construction", Proc. Seventeenth Conference on Uncertainty in Artificial Intelligence, 2001.

[25] E. K. Jones, N. Dennis and D. Hunter, "Hypothesis Management for Information Fusion", In IEEE Aerospace and Electronic Systems, ISSN 0885-8985, Vol. 18, No. 6, June 2003.

[26] G. Laskey, K. Laskey "Combat Identification with Bayesian Networks", Proc. Command and Control Research and Technology Symposium 2002.

[27] K.B. Laskey, D. Barbara, T. Shackleford, E. Wright, J. Fitzgerald, *Detecting Threatening Behavior Using Bayesian Networks*, In Proceedings of the Conference on Behavioral Representation in Modeling and Simulation, 2004.

[28] L. D. Hudson, B.S. Ware, S.M. Mahoney, K.B. Laskey, "An application of Bayesian networks to antiterrorism risk management for military planners", Technical Report, Department of Systems Engineering and Operations Research, George Mason University.

[29] K. B. Laskey , T. Levitt, "Multisource Fusion for Probabilistic Detection and Opportunistic Assessment of Terrorist Threats", presented at Aerosense 2002.

[30] Kevin P. Murphy, "Dynamic Bayesian Networks: Representation, Inference and Learning", University of California, Berkeley, 2002, US.

[31] K.B. Laskey, "MEBN: First-Order Bayesian Logic for Open-World Probabilistic Reasoning", http://ite.gmu.edu/~klaskey/publications.html, George Mason University Department of Systems Engineering and Operations Research, US, 2004.

[32] H. A. Kautz, "Research Statement: Efficient Real-World Knowledge Representation and Reasoning", www.cs.washington.edu/homes/kautz/kautz-research-statement.pdf

[33] P. Cohen, C. Perrault and J. Allen, "Beyond Question Answering", In Knowledge Representation and Natural Language processing. W. Lehnart and M. Ringle, Ed., Lawrence Erlbaum Associates, 1981.

[34] M. J. Huber and E. H. Durfee and M. P. Wellman, "The Automated Mapping of Plans for Plan Recognition," Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence (UAI), Seattle, Washington, July, 1994, pgs 344-351.

[35] Van-T. Vu, F. Brémond and M. Thonnant, " Video surveillance: human behaviour representation and on-line recognition", The *Sixth International Conference onKnowledge-Based Intelligent Information & Engineering Systems* (KES'2002), Podere d'Ombriano, Crema**,** Italy, 16, 17 & 18 September 2002.

[36] D. Fu, E. Remolina, J. Eilbert, "A CBR Approach to Asymmetric Plan Detection, Workshop on Link Analysis for Detecting Complex Behavior", (LinkKDD2003) August 27, 2003.

[37] F. Mulder, F. Voorbraak, "A formal description of tactical plan recognition", Journal of Information Fusion 4(1): 47-61 (2003)

[38] A. R. Pearce, C. A. Heinze, and S. Goss, "Enabling perception for plan recognition in multi-agent air-mission simulations". *Proc. Fourth International Conference on Multi-Agent Systems (ICMASS2000)*, pp 427-428, 2000.

[39] Bui, H. H., Venkatesh, S., and West, G. "Policy recognition in the Abstract Hidden Markov Model", Journal of Artificial Intelligence Research, 17: 451-499, 2002.

[40] S. Saria and S. Mahadevan, "Probabilistic Plan Recognition in Multiagent Systems", *International Conference on AI and Planning Systems (ICAPS)*, 2004.

[41] R. P. Goldman, C. W. Geib and Christopher A. Miller, "A New Model of Plan Recognition", In *Proc. Fifteenth Conference on Uncertainty in Artificial Intelligence*, 1999.

[42] P. Cohen, R. Schrag, E. Jones, A. Pease, A. Lin, B. Starr, D. Gunning and M. Burke, "The DARPA High-Performance Knowledge Bases Project", AI Magazine, Winter 1998, p. 30.

[43] L. Zadeh, "Fuzzy sets", *Information and Control*, 8:338–353, 1965.

[44] R. Suzić, "Representation and Recognition of Uncertain Enemy Policies Using Statistical Models", *In Proceedings of the NATO RTO Symposium on Military Data and Information Fusion,* Prague, Czech Republic, 2003.

[45] F.O. Karray, C. de Silva, "Soft Comupting amd Intelligent Systems Design", ISBN 0-321-11617-8, Pearson Education Limited, GB, 2004.

[46] S. Mulgund, G. Rinkus, C. Illgen, & J. Friskie, "OLIPSA: On-Line Intelligent Processor for Situation Assessment", *Second Annual Symposium and Exhibition on Situational Awareness in the Tactical Air Environment*, Patuxent River, MD, US, 1997.

[47] K.P. Murphy, "Bayes Net Toolbox for Matlab" , *Last updated on 29 May 2003*, http://www.ai.mit.edu/~murphyk/Software/BNT/bnt.html

[48] L. R. M. Johansson, "Information Acquisition in Data Fusion Systems", TRITA-NA-0328, ISSN 0348-2952, ISRN KTH/NA/R--03/28--SE, ISBN 91-7283-655-5, CVAP283, November 2003.

[49] L. R. M. Johansson and R. Suzić, "Bridging the Gap between Information Need and Information Acquisition", *Proceedings of the 7th International Conference on Information Fusion*, ISIF, Fusion, 2004.

[50] R. Suzić and R. Johansson, "Realisation of a Bridge between High-Level Information Need and Sensor Management Using a Common DBN", In *proceedings of 2004 IEEE International Conference on Information Reuse and Integration (IEEE IRI-2004*), November, 2004.

[51] Stefan Arnborg and Joel Brynielsson, "Bayesian games for threat prediction and situation analysis", In *Proceedings of the 7th International Conference on Information Fusion*. International Society of Information Fusion, 2004.

[52] M. Arulampalam, S. Maskell, N. Gordon and T. Clapp, "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking", *IEEE Transactions on signal processing*, Vol. 50, No. 2, February 2002.

[53] D. Rios Insua and F. Ruggeri (Eds.) *Robust Bayesian Analysis (Lecture Notes in Statistics)*, volume 152. Springer-Verlag, 2000.