Reasoning with uncertainty in the situational awareness of air targets

LTZE2 B.G.M.Mertens

21st July 2004

Man-Machine Interaction, Mediamatics,
Electrical Engineering, Mathematics and Computer Science,
Delft University of Technology,
Royal Netherlands Naval College,
Combat System Department,
The Netherlands

Graduation Committee:

Dr. drs. L.J.M.Rothkrantz,
Prof. dr. ir. E.J.H. Kerckhoffs,
Prof. dr. H. Koppelaar,
KLTZE ir. F. Bolderheij,
Prof. dr. ir. F.G.J. Absil.

Abstract

In combat simulations target classification and identification are very important. In this research area several studies about simulating identification have been done, most of them take a set of information like "the target is visually identified hostile" to start the simulation with. Mostly classification is not taken into account in identification problems.

In this report the input consists of basic sensor data and a priori knowledge. This will be combined into information which is necessary to evaluate the situation. Based on this information the complete situational awareness is evaluated.

To derive information out of sensor data, facts have to be derived in three areas, these are facts concerning position, identity and behaviour. Based on these derived facts a decision will be made about the classification and the identification of the target.

Two Bayesian reasoning models were designed for the decision processes of the targets classification and identification. These models are designed as much alike as possible. An implementation was made to test the models. In the implementation temporal aspects are not taken into account but the results were promising.

To conclude we conducted a literature survey to investigate the possibilities of temporal reasoning in this project.

Preface

This thesis has been written as a result of a research assignment that I have worked on during my graduation at the Royal Netherlands Naval College, Combat Systems Department and as part of my graduation project for the Delft University of Technology. The project was partially carried out at TNO Physics and Electronics Laboratory (TNO-FEL) and concerns the naval air defence simulation model SEAROADS II. The second part of this project was carried out at the Delft University of Technology and the Royal Netherlands Naval College. In the second part the system designed in the first part of the project was implemented and tested. Because real data is hard to get a simulation was used to acquire the necessary information. The simulation that was used in this part of the project was made for the STATOR project.

Acknowledgments

At first I would like to thank my supervisors drs.dr. L.J.M. Rothkrantz of Delft University of Technology, KLTZE ir. F. Bolderheij of the Royal Netherlands Naval College and R. Witberg of TNO-FEL for their guidance, support and advice during my graduation period.

From the OPSCHOOL I would like to thank LTZ 2 van Dijk for his operational knowledge.

And at last but not least I would like to thank my parents and boyfriend for their patience and support during my graduation period.

Den Helder, July 2004,

Bionda Mertens

Contents

List of Figures

Abbreviations & Acronyms

AAW	Anti-Air Warfare
ADCF	Air Defence and Command Frigate
ACO	Air Co-ordination Order
APAR	Active Phased Array Radar
ARM	Anti Radiation Missile
ASM	Air to Surface Missile
BBN	Bayesian Belief Network
BPA	Basic Probability Assignment
CM	Cruise Missile
CRC	Class, Responsibility and collaboration
DAG	Directed Acyclic Graph
DBN	Dynamic Bayesian Network
DM	Decision Making
DPC	Defence Planning Committee
EMCON	EMission CONtrol
EO	Electro Optic
ESM	Electronic Support Measures
HMM	Hidden Markov Model
IDCRITS	IDentification CRITeria
IFF	Identification Friend-or-Foe
IR	InfraRed
ISR	Identification Safety Range
KFM	Kalman Filter Model
LVO	Air Defence Officer (Lucht Verdedigings Officier)
NAC	North Atlantic Council
NATO	North Atlantic Treaty Organisation
OCL	Object Constraint Language
OPSCOOL	OPerational SCHOOL
OPSROOM	OPerationS ROOM
PARSER	See SEAPAR
ROE	Rules Of Engagement
RNlN	Royal Netherlands Navy
SA	Situational Awareness
SEAPAR	Scheduling and Evaluation of an Active Phased Array Radar
SEAROADS	Simulation, Evaluation, Analysis and Research On Air Defence Systems

SAM	Surface to Air Missile
SSM	Surface to Surface Missile
TBM	Theater Ballistic Missile
TBMD	Theater Ballistic Missile Defence
TE	Threat Evaluation
TG	Task Group
TIC	Target Identification and Classification
TNO	Netherlands Organisation for Applied Scientific Research (Nederlandse Organisatie voor Toegepast Natuurwetenschappelijk Onderzoek)
TNO-FEL	TNO Physics and Electronics Laboratory (TNO-Fysisch en Elektronisch Laboratorium)
UAV	Unmanned Air Vehicle
UML	Unified Modeling Language
WIC	War Identification Code

Part I

The problem definition

Chapter 1

Introduction

This report was written as a part of my graduation project for the Royal Netherlands Naval College and the Delft University of Technology. Because of the military aspect of this research project several military terms will be used frequently. These terms are explained in Appendix A. Two terms are significant to the understanding of this report so I will define them first.

Classification is the process in which the sort of target is determined, e.g. is it an airplane or a missile?

Identification is the process in which the intention of the target is determined, e.g. is it a friend or a foe?

1.1 Project description

The first part of this project was carried out at the Netherlands Organisation for Applied Scientific Research - Physics and Electronics Laboratory (TNO-FEL) and concerns the naval air defence simulation model SEAROADS II. This model was developed by TNO-FEL and was funded by the Royal Netherlands Navy. SEAROADS II simulates scenarios composed of an attack by fighters, Anti-Ship Missiles, or Tactical Ballistic Missiles and the defence of a single ship or a task group against this attack. I used SEAROADS II to develop a model for the classification and identification of air targets in a naval environment.

The second part of this project was carried out at the Delft University of Technology and the Royal Netherlands Naval College. In the second part the model designed in the first part of the project was implemented and tested. Because real data is hard to get a simulation was used to acquire the necessary information. The simulation that was used in this part of the project was made for the STATOR project.

In this report an overview is presented for the picture compilation of a target. There have been several studies about simulating identification, most of them take a set of information like “the target is visually identified hostile”. Mostly classification is not taken into account in identification problems.

This report will formalise the derivation of facts concerning position, identity and behaviour out of sensor information. Based on these derived facts a decision will be made about the classification and identification of the target. The role of Rules Of Engagement (ROE) in this decision process is made clear. Finally

the influence of time in this simulation will become clear and the benefits of temporal reasoning will be looked at.

1.2 Project goal

Because SEAROADS II is a simulation of combat vessels in an air defence scenario, this project is narrowed to Anti-Air Warfare (AAW).

My graduation project involves modeling and implementing a temporal reasoning system which is able to make a decision about the *classification* and *identification* of an air target. This is a challenging problem because most rules used on board combat vessels are vague.

Therefore we will start with a study of the classification and identification process as it is currently done on board Dutch combat vessels. We then conduct a literature study of reasoning models with and without temporal features which can be used. We will design a system that is able to make a decision about the classification and identification of an air target based on basic sensor data collected by a naval vessel. This system will be implemented and tested using simulated data, because real data is classified and therefore hard to obtain.

1.3 Report structure

In Chapter 2 an overview of the way a target is represented on board a combat vessel is presented and an introduction to the existing SEAROADS II system is given. Some possible solutions for the reasoning in the model are presented and discussed in Chapter 3. We also make a choice between the possible artificial intelligence techniques and explain how the chosen techniques could be used. In Chapter 4 an overview of the model is given. Then in Chapter 5 the architecture of the model is worked out in more detail. In Chapter 5 the model is formalised and in Chapter 6 the probabilistic and causal models are presented. As an extension of these models the time aspect is introduced by investigating some temporal reasoning techniques in Chapter 7.

After a thorough description of the model we describe the design of the prototype using Unified Modeling Language (UML) in Chapter 8. In Chapter 9 the prototype is presented. In Chapter 10 we will describe the test scenario's and their results.

Chapter 2

The existing situation

2.1 Situational awareness on board

On board a combat vessel a clear picture of all surrounding targets is essential. Therefore a team of experts evaluates all information gathered by sensors on board the vessel and data communication with allied forces. Based on this information together with guidelines and rules supplied by the government (ROE) a decision is made on several topics. These topics are classification, identification and attack-decision evaluation. Although the decision-making on board a combat vessel sometimes differs from the methods used in the simulation the former is the base from which the model is built. Therefore the decision process is investigated and analysed into gathered information elements and the usage of these elements in the actual evaluation. In this chapter an overview is given for all information that is necessary to draw sensible conclusions. An explanation of terms used in this chapter can be found in Appendix A. The decisions on board combat vessels can be represented in an OODA loop (Observe, Orient, Decide and Act). In Figure 2.1 these phases are translated to military terms [6] and [1]. In this report we take a look at the first three phases, Situational Awareness (SA), Threat Evaluation (TE) and Decision Making (DM).

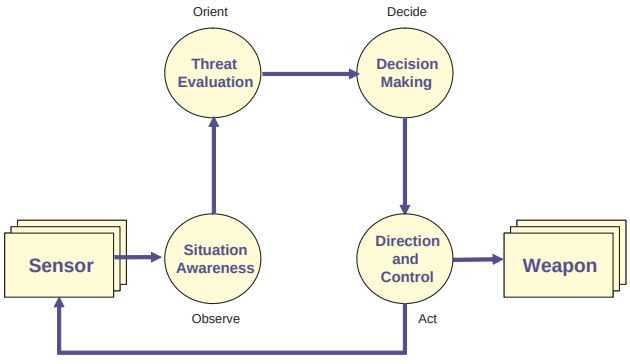

Figure 2.1: An overview of the process

2.1.1 Basic information about the situation

To make classification and identification possible the following information about the target has to be gathered from the environment. This information is gathered for all surrounding targets by sensors on board the vessel and the link 11 system which provides target information which is already processed by allied vessels. A list of direct measurable sensor data and a-priori knowledge which is necessary is given here:

- Target track;
- IFF[1] on board?
- IFF mode;
- Vesta[2] on board?
- Link 11[3] on board?
- ESM[4] signature;
- ROE in force.

2.1.2 Derived information

The data gathered by sensors gives raw information. By combining this raw information in the right way detailed information can be derived. On board the combination of information is done by operators which are trained to recognise certain patterns on their screen. But they also combine information on their screen with information in maps and a priori knowledge about the enemy. They derive information concerning position, identity and behaviour.

According to the position:

- Adherence to airlane;
- Adherence to air co-ordination order(ACO);
- In military speed/altitude domain;
- Flying in formation;
- Manoeuvring;
- Inside identification safety range (ISR).

According to identification:

- Visual identification friendly/hostile;
- ESM friendly/hostile;
- IFF.

[1] see Appendix A
[2] see Appendix A
[3] see Appendix A
[4] see Appendix A

According to behaviour evaluation:

- Hostile act;
- Hostile intent;
- Performs identification.

2.1.3 Decision making

In the process several decisions have to be made based on derived information and ROE. ROE are a list of directives to military forces (including individuals) that define the authorisation for, or limits on, the use of force during military operations. In Figure 2.2 the influence of ROE on the reasoning process is visualised. These decisions are made by OPSROOM (operations room) officers.

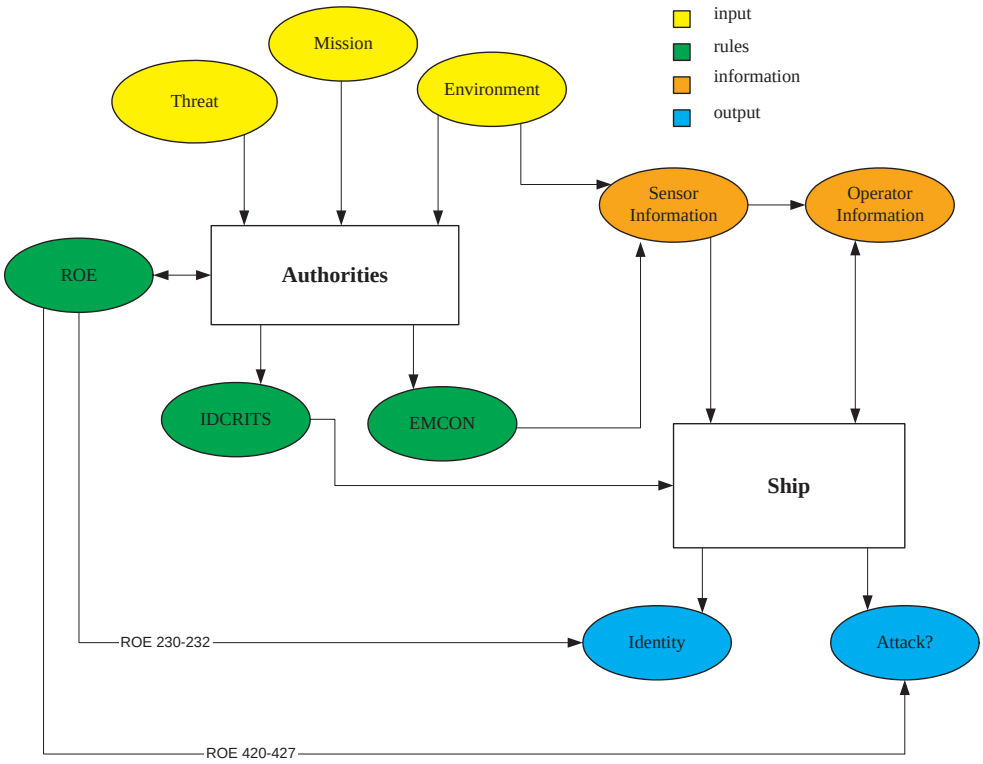

Figure 2.2: An overview of how ROE influence the decision process

The authorities decide based on the environment, the threat and the mission which ROE will be in force, which identification criteria (IDCRITS) have to be satisfied and which sensors may be used (EMCON). This information is sent to every allied ship in the operation so they can update their sensor configuration. The ship will receive sensor information which is dependent on the EMCON, this sensor information will be analysed and combined by operators which leads to facts about surrounding targets. Decisions about the identification of the target and the authority to attack the target may be done based on the IDCRITS and the ROE in force. The meaning of specific ROE mentioned in the figure can be found in Appendix B.

Classification

Classification is an important matter, because during the identification process as well as during the evaluation of the rules of engagement, the target's capability has to be considered. Sometimes a classification can lead to a direct identification. For example if the enemy is the only one with F-16's the classification of a target being an F-16 directly indicates that the target is hostile.

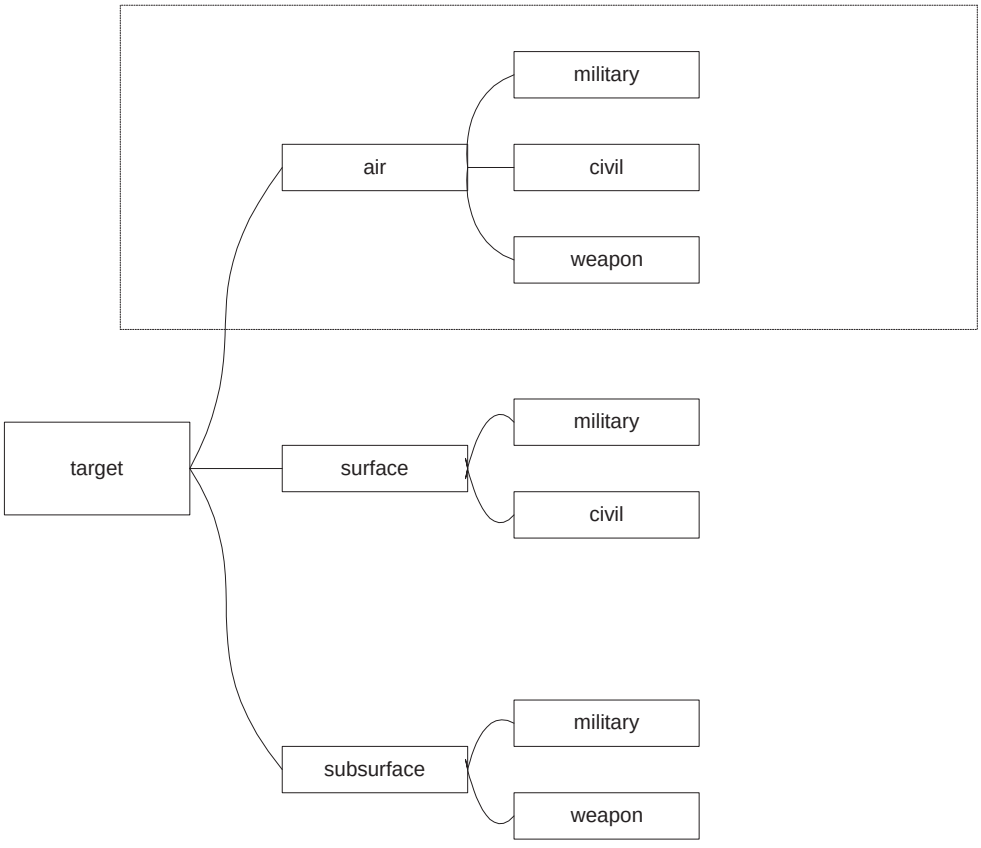

Figure 2.3: [2] An overview of the classical approach of classification

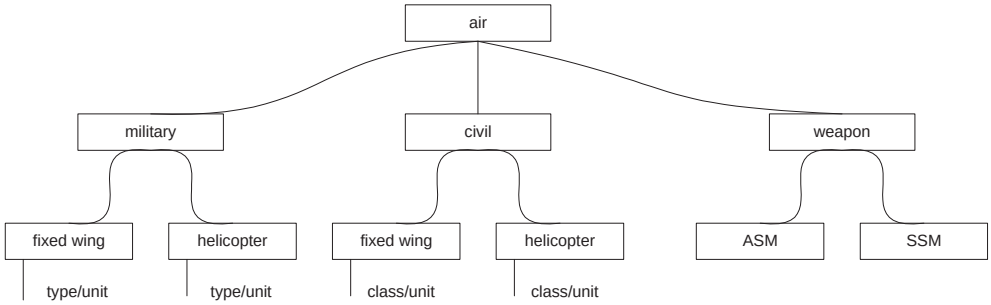

Figure 2.4: [2] An overview of the classical approach of classification of air targets

The classic way of dealing with a classification problem is shown in Figure 2.3 and Figure 2.4 [2].

It was very common to divide targets in civil and military targets. But nowadays the difference between civil and military is fading because of terrorism. Therefore the division made in both diagrams is not up to date anymore. In this report we will present an alternative approach, in Section 6.1 this approach is worked out in detail.

Identification

Some of the facts mentioned in Section 2.1.1 and 2.1.2 contain stronger information than others about identification[5] of the target. In the real situation some of these facts immediately lead to a positive identification. These facts are listed below:

- Visual identification
- IFF mode 4
- unique ESM
- link 11
- voice cross-tell
- 2 way communication + position check
- identification manoeuvres
- according to ACO

[5]see Appendix A

The other facts out of the list in Section 2.1.2 give a direction for the identification of the target but on their own they cannot be the base of a positive identification. It is possible to assign a pending identification to a target based on these facts. If more information is gathered the pending identification may be changed to a positive identification. These facts have to be combined by using IDCRITS[6]. Based on this combination an identification can be assigned to a target. IDCRITS can differ per operation, these criteria are based on the ROE. Before an identification is assigned to a target we have to check if all ROE are satisfied.

2.1.4 Reasoning with uncertainty

Human experts show remarkable skill in drawing conclusions from limited information. Typically, the evidence available to an expert is merely suggestive, vague and highly incomplete. Nevertheless experts are usually able to draw sensible conclusions. In a combat situation the enemy will try to mask as many features as possible which will reveal his identity. Therefore an operator is never entirely certain about a conclusion. In the classification and identification process there are three kinds of uncertainty.

The first one is the uncertainty about the sensors; the sensor information will never be exactly the same as the real situation. This uncertainty is left out of consideration in our model, because the sensors modeled in the simulation give exact values. In the definition of the model possible solutions for dealing with this uncertainty are given. The second one is uncertainty about the correctness of the information derivation. The last one is uncertainty about the consequence of a certain fact. The problem of handling uncertainty in knowledge-based systems has proven to be hard. Literature contains many approaches, in the next chapter the most commonly used approaches will be treated.

[6] see Appendix C

2.2 SEAROADS II: The existing system

The maritime section of the Operations Research and Business Management division of the TNO-FEL supports the Royal Netherlands Navy (RNlN) and other clients in the procurement and deployment of naval ships. For the Anti Air Warfare (AAW) aspects the medium-level simulation model SEAROADS is used in many maritime air defence studies. SEAROADS is an acronym for "Simulation, Evaluation, Analysis and Research On Air Defence Systems". It is used to quantify and analyse the air defence capability of one or more ships or land based units. The model simulates scenarios composed of an attack (by fighters, Anti-Ship Missiles (ASM), or Tactical Ballistic Missiles (TBM), which can be seen in Figure 2.5) and the defence of a single ship or a task group against this attack. This includes simulation of the attack, all relevant sensor and weapon systems, the threat evaluation and weapon assignment rules of all units, and the possible communication and co-ordination between the units. The development of SEAROADS was initiated and funded by the RNlN. It started as a single ship model, containing only hard kill weapon systems. Over the years the model has been expanded; the current version of SEAROADS includes hard kill and soft kill weapon systems and is able to handle a task group. The development is carried out by TNO-FEL, who is also the direct user of the model, so the model can easily be adapted to incorporate specific combat systems [35]. In the next sections topics in SEAROADS concerning this report are discussed.

2.2.1 Sensor information

Traditionally, warships are equipped with rotating radars, fire control radars and passive sensors like Electronic Support Measures (ESM) and Infrared (IR) Search and Track systems. In SEAROADS, these systems are modeled in detail. A detailed radar equation including aspects like clutter, ducting and multi-path is used for radar. A new system is the multi-function radar APAR (Active Phased Array Radar), for evaluation of this new system on the new Dutch Air Defence and Command Frigates (ADCF), this radar has also been included in SEAROADS.

Apart from functions as search and track, APAR can also perform Mid-Course Guidance and Terminal Illumination to guide own deployed missiles (SAMs). For the Royal Netherlands Navy and most other navies, the APAR is a new type of system. By using SEAROADS the benefits of this multi-function radar for naval AAW could be demonstrated. Detailed modeling of the scheduling on millisecond-level of the APAR functions is done by the model PARSER, providing input parameters for SEAROADS. An example of such input parameters are the number of Terminal Illuminations that one APAR-antenna face can execute simultaneously and the remaining search capability as a function of this number [35].

2.2.2 Classification and identification

Classification and Identification in SEAROADS are implemented as a delay. When a target is in sensor-range, after a set delay all information about the target is known and is equal to the information given in the operation order.

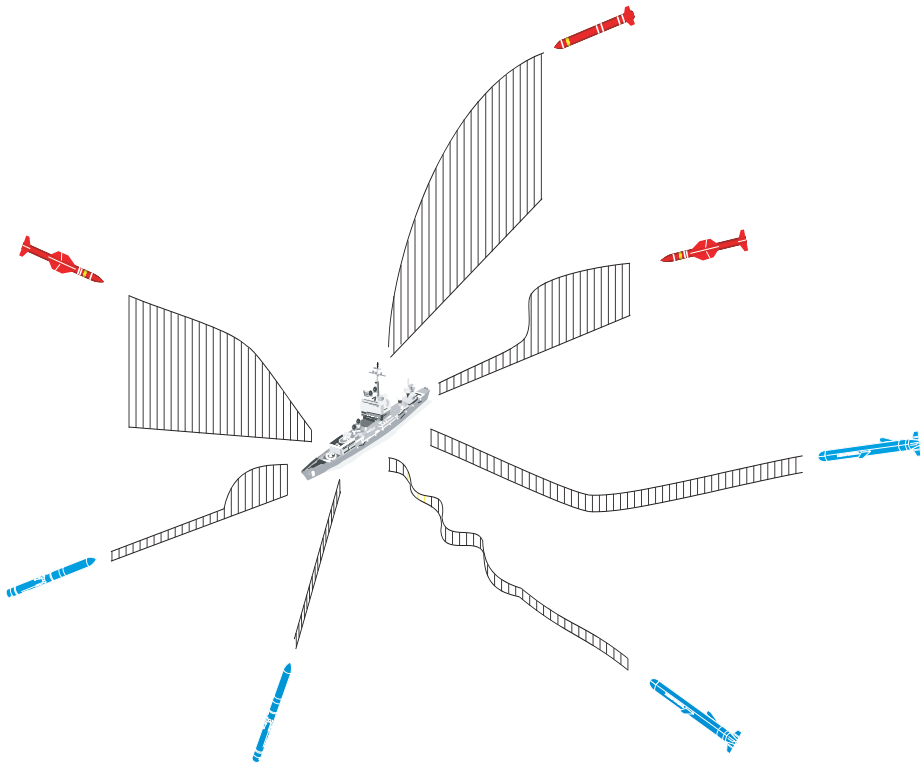

Figure 2.5: An overview of possible simulated attacks

The delay is set as the reaction time, this is in the existing model about 4 seconds [40].

Part II

The design of the model

Chapter 3

Possible reasoning models

3.1 Introduction

The reasoning models that are described in this chapter can be seen as tools to be used to reason about the information we get about the situation. From these tools we have to select the one that will fit our purpose best. The reasoning model we choose will determine the way the information from the simulated situation will be handled, how the information is rated and how these rates are combined into probabilities. In the end this will have an effect on the identification assigned to a target. To make a deliberate choice we will determine the requirements that our program must fulfill. Then a short description will be given for the most common used reasoning models, with advantages and disadvantages of each model. Finally we will choose the best model for this project and discuss the arguments that led to the choice.

3.2 Requirements

Our model has to be able to draw a conclusion for the classification and the identification of a target based on just little information. An enemy will try to conceal as much information as possible, so mostly the available information is very poor. The model has to draw a conclusion based on facts and confidence gained from the existing situation. A list with probabilities assigned to conclusions based on facts has to be assembled in consult with an expert. The program has to combine these probabilities with the confidence factor of the proven facts to a probability for a certain identification. These probabilities will lead to a decision about the identification of the target based on the ROE. Therefore the program will need the features listed on the next page.

- Collect all necessary information from the situation;
- Derive facts from the gained values;
- Rate the confidence in each fact;
- Assign probabilities for certain conclusions based on each fact;
- Combine the confidence and probabilities for all facts;
- Draw conclusions from the given information.

This process is applicable to classification and identification, with the difference that in the classification process an iterative deepening will take place. The reasoning process for the attack evaluation is slightly different, because the only facts that have to be considered are ROE. In ROE there is no uncertainty, they are in force or they are not. The only uncertainty is located in the input. The facts which are used to draw conclusions differ in these cases.

3.3 Reasoning models

Now the possible models will be described. For each model we will give a short description and sum up the advantages and/or the disadvantages of the model. In this section uncertainty will be represented as a probability. More detailed information about the most commonly used approaches can be found in the following books, papers and internet sites [9], [10], [11], [14], [21], [25], [28], [29], [32], [33], [43], [44] and [45].

3.3.1 Bayesian Belief Network

Bayesian belief networks are often used in reasoning systems. They have proven themselves in a number of applications. In a Bayesian Belief Network one can indicate the effect an event has on another event. One can say for example that the chance it will start raining given the fact that it is cloudy is 0.3. If one also adds the fact that it rained yesterday and specifies the influence that has on the probability that it will start raining a Bayesian network has been created. Bayesian networks can be visualised as directed graphs (see Figure 3.1). Given some evidence (some facts should be specified as being true) the probability of the events one is interested in (the query events) can be calculated. For example the query event can be the fact that it is raining given the facts that it is cloudy, but it did not rain yesterday, but the query event can also be the fact that it is cloudy given the fact that it is raining. One can also state that it is not completely certain an event has happened, but that it happened with a certain probability. This probability is then taken into account when the probability of the queried event is calculated.

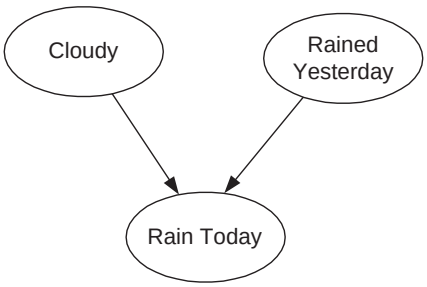

Figure 3.1: An example of a Bayesian Belief Network

Advantages: A lot of research has been done on Bayesian belief networks and they are being used in a lot of applications.

Disadvantages: Bayesian belief networks can be very time-consuming and probability inference in complicated belief networks has been shown to be difficult [4]. A lot of research on real-time inference algorithms has been done though and a couple of approximation algorithms have been developed. There is no real difference between uncertainty about the information and lack of information in the basic Bayesian networks.

Another disadvantage is that the probabilities used in the Bayesian Belief Network should be very precise and should in the ideal case be derived from datasets. This is not the case in our model where the probabilities have been derived by expert information and are just indications, not crisp numbers.

3.3.2 Certainty factors

The certainty factor model, which was introduced by Shortliffe and Buchanan, is a model that can be applied to expert system using a rule base for the reasoning process. Certainty factors can be used if one wants to state that for a rule "if fact then conclusion" the conclusion is not entirely certain, uncertainty in this model is handled by assigning (un)certainty factors to every rule and basic fact. This might seem a very interesting model, but there is one major drawback. Several researchers showed that the certainty factor model is inconsistent with the basic axioms of probability theory. The certainty factor model was very popular in expert systems in the 1980's but got so much criticism over the years that it has been abandoned and nowadays it is only considered to be interesting from a historical point of view [16]. In most applications the certainty factor model has been replaced by Bayesian belief networks, since the two models have a lot in common.

Advantages: The simplicity of this model is very appealing; it seems to be intuitively correct.

Disadvantages: The model cannot be theoretically justified. The results with MYCIN (the first certainty factor model) were promising and quite good, but research showed the model being very robust, variations of 0.2 in the certainty factors hardly affected the conclusion [32].

3.3.3 Markov models

Markov models are sequential model using a finite number of possible states that obey the Markov assumption. The Markov assumption states that future states are independent from the past, given the current state. In a transition matrix the probabilities (Si) of every possible transition from state m to state n are listed. The process can be only in one state at a time. Markov models are statistical models of sequential data that can be used to recognise a pattern in a sequence of data over a period of time. Markov models have successfully been applied to speech recognition, pattern recognition and target tracking applications. They are particularly useful to analyse a sequence of data that can be characterised as a signal. A Markov model can be visualised using a Bayesian network. An example is given in Figure 3.2 and Table 3.1.

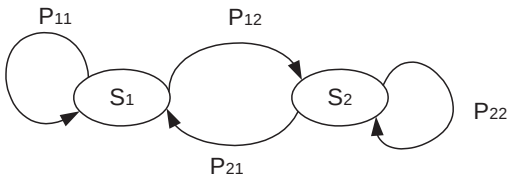

Figure 3.2: An example of a Markov model

	S_1	S_2
S_1	P_{11}	P_{12}
S_2	P_{21}	P_{22}

Table 3.1: A transition matrix

Advantages: Markov models have successfully been applied in a number of pattern recognition applications.

Disadvantages: To use Markov models one needs precise information about the probabilities of the state transitions.

3.3.4 Dempster-Shafer theory

The Dempster-Shafer theory is a mathematical model to model a person's belief in a fact. It provides a method to combine measures of belief in a fact induced by different pieces of evidence, the so-called Dempster's rule of combination. In the Dempster-Shafer theory evidence of different levels of abstraction can be represented easily and discrimination between uncertainty and ignorance can be made. In the Dempster-Shafer theory the probabilities of the facts do not all have to be specified exactly. So it is possible to reason with only partial information.

Advantages: One of the advantages of the Dempster-Shafer theory is that the degrees of belief for a question can be obtained from the probability distribution for another question [27]. Dempster's rule of combination gives the availability to combine different bodies of evidence.

Disadvantages: A theoretical justification for Dempster's rule is problematic, still Dempster-Shafer theory is considered to be an alternative. The Dempster-Shafer model gives an interval between the lower probability and the higher probability, a decision rule is essential to draw the right conclusion.

3.3.5 Fuzzy logic

Fuzzy logic is one of the approaches that could be used to represent uncertainty too. Fuzzy logic makes it possible to convert a crisp value into a fuzzy value and reason with that fuzzy value to come to a fuzzy conclusion. This fuzzy conclusion can then be defuzzified to get a crisp conclusion. For example, we can define the fuzzy set for temperature as T={HOT, TEPID, COOL} and radiator setting R={HIGH, NORMAL, LOW} as visualised in Figure 3.3.

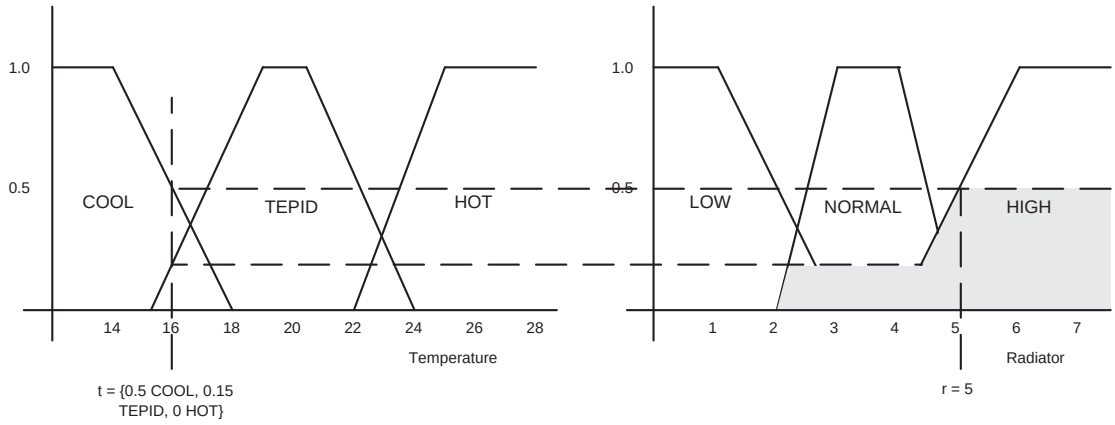

Figure 3.3: An example of a fuzzy set

Suppose we have two rules in our knowledge base:

if temperature is COOL then radiator is HIGH
if temperature is TEPID then radiator is NORMAL

And suppose we have a temperature reading of 16° , what value would the radiator get? To calculate the answer to that question we would use the reasoning method shown in Figure 3.3. The crisp temperature value of 16° can be fuzzified as t={0 HOT, 0.15 TEPID, 0.5 COOL}. Now we can reason with this fuzzy value and the rules in the rule base to get a fuzzy value for the radiator setting as shown in the figure. The fuzzy output value is r={0 LOW, 0.15 NORMAL, 0.5 HIGH}, this value would have to be defuzzified, which can be done in several ways. The method that is shown in the figure is called the First Maximum defuzzification method. In this method the fuzzy set with the highest value is observed, in this case radiator HIGH.

The first time the fuzzy set reaches the computed value the defuzzified output results in a radiator setting of 5, as shown in the figure. Fuzzy logic has proven to be particularly useful for small control applications. But it is usually not the best choice when it comes to bigger applications.

Advantages: Fuzzy logic is a nice method because it is intuitive and it makes it possible to define the rules in normal language.

Disadvantages: There is no completeness in the inference formalism and the mathematical basis for fuzzy logic is rather coarse [20].

3.4 Choosing a model

From the described models we have to choose the one that can implement the required functions best. If we look at the requirements given in Section 3.2 we see that reasoning with just a little information must be possible. Not all information will be available, the amount of information depends on the situation. The classic Bayesian model is therefore less appropriate, but there has been research in this matter and solutions are offered. Although the classic Bayesian mathematical model is the basis of Bayesian belief networks research has shown that these can handle situations in which not all information is available. Bayesian belief networks are an option for the implementation of the given requirements.

In this situation the past influences the future and it is not possible to assign any sensible probability to transitions like from assumed friendly to friendly, so a Markov model is not useful.

The mathematical basis for Fuzzy combinations is coarse. There is not one clear combinational rule, but several different ways are offered to combine the given facts. Lots of testing is necessary to find the best fuzzy combination algorithm for this problem. Therefore a fuzzy solution is rejected.

In a Dempster-Shafer model the outcome is presented as an interval, there has to be another decision process to draw a final conclusion, this will slow the process down and is not desirable. But in Dempster-Shafer's model there is an explicit difference between uncertainty about the information and lack of information. Another advantage of Dempster-Shafer's model in this case is that the probability of occurrence of a hypothesis does not imply any probability of the inverted case, and last but not least Dempster's rule of combination gives a strong basis for the combination of several facts to one conclusion. Because of these advantages Dempster-Shafer's theory is an option for the implementation of our requirements.

Based on the previous discussion, we decided that both Dempster-Shafer and Bayesian belief networks are suitable for this situation, a closer look is given to them both. Therefore a more detailed description of the Dempster-Shafer model and the Bayesian belief model are given in the next sections. A final decision is made afterward.

3.4.1 The Dempster-Shafer model

Dempster-Shafer theory has been developed by Glenn Shafer based on earlier work of Arthur Dempster, who attempted to model uncertainty by a range of probabilities rather than as a single probabilistic number [7]. Dempster-Shafer has its own extensive terminology which is explained in Appendix D. The Dempster-Shafer theory assumes that there is a fixed set of mutually exclusive and exhaustive elements called the environment (Θ) or frame of discernment.

In the identification case for example:

$$\Theta = \{hostile, suspect, neutral, assumedfriendly, friendly, unknown\}$$

Each subset of Θ can be seen as a possible answer to the question "What is the identification of the target?".

In Dempster-Shafer theory a degree of belief is used in stead of probabilities, this degree of belief has to be seen analogous to the mass function of a physical object. The evidence measure (m) is analogous to the amount of mass. A priori a set of evidence measures is assigned, this is called the basic probability assignment (BPA). A fundamental difference between Dempster-Shafer theory and probability theory is the treatment of ignorance. The Dempster-Shafer theory does not force belief to be assigned to ignorance or refutation of a hypothesis. Where probability theory assigns 50% chance to the target being friendly even if there is no evidence at all about the identification of the target Dempster-Shafer theory has another solution. The mass is assigned only to those subsets of the environment to which you wish to assign belief. Any belief that is not assigned to a specific subset is considered nonbelief and is just associated with the environment Θ. Belief that refutes a hypothesis is disbelief, which is not nonbelief.

Now let's look at the case in which additional evidence becomes available. We have to combine the evidence to produce a better estimate of belief in the evidence. The way the combination is done using Dempster's rule of combination will be made clear using an example. Given two evidence measures, for fact 1 the mass-distribution for the possible conclusion (X) can be given as:

$$m_1(X) = \begin{cases} 0.3 & \text{if} \quad \text{X=}\Theta \\ 0.7 & \text{if} \quad \text{X=\{hostile,neutral\}} \\ 0 & \text{else} \end{cases} \tag{3.1}$$

For fact 2 the mass-distribution for the possible conclusion (Y) can be given as:

$$m_2(Y) = \begin{cases} 0.4 & \text{if} \quad \text{Y=}\Theta \\ 0.6 & \text{if} \quad \text{Y=\{hostile\}} \\ 0 & \text{else} \end{cases} \tag{3.2}$$

This evidence can be combined using Dempster's rule of combination, given with the following formula:

$$m_1 \oplus m_2(Z) = \sum_{X \cap Y = Z} m_1(X)\, m_2(Y) \tag{3.3}$$

Table 3.4.1 shows the masses and product intersections for the identification arranged in a table.

Table 3.2: Confirming evidence

m1 / **m2**	{**hostile, neutral**} **0.7**	Θ **0.3**
{**hostile**} **0.6**	{hostile} 0.42	{hostile} 0.18
Θ **0.4**	{hostile, neutral} 0.28	Θ 0.12

Each set intersection is followed by its numeric mass product. The entries in the table are calculated by cross-multiplying mass products of rows and columns. This leads to a new evidence measure:

$$m_1 \oplus m_2\,(Z) = \begin{cases} 0.12 & \text{if} \quad \text{Z=}\Theta \\ 0.60 & \text{if} \quad \text{Z=\{hostile\}} \\ 0.28 & \text{if} \quad \text{Z=\{hostile, neutral\}} \\ 0 & \text{else} \end{cases} \tag{3.4}$$

But instead of restricting belief to a single value there is a range of belief in Dempster-Shafer's theory. The belief ranges from a 0.60 that the identification of the target is hostile to (0.60+0.12+0.28=1) that the identification of the target might be hostile. In evidentional reasoning the evidence is said to induce an evidential interval. The lower bound is called the support and the upper bound is called the plausibility. For this example the evidential interval is [0.6,1].

The plausibility can be calculated in a way similar to the calculation of the belief, in that case the evidence which contradicts with the given conclusion is used. For example if there had been any evidence for the target not being hostile, the plausibility would be less than 1.

Let's look at an other case to make it clear:

$$m_1\,(X) = \begin{cases} 0.3 & \text{if} \quad \text{X=}\Theta \\ 0.7 & \text{if} \quad \text{X=\{hostile,neutral\}} \\ 0 & \text{else} \end{cases} \tag{3.5}$$

$$m_2\,(Y) = \begin{cases} 0.4 & \text{if} \quad \text{Y=}\Theta \\ 0.5 & \text{if} \quad \text{Y=\{hostile\}} \\ 0.1 & \text{if} \quad \text{Y=\{neutral\}} \\ 0 & \text{else} \end{cases} \tag{3.6}$$

Table 3.4.1 shows the masses and product intersections for the identification arranged in a table.

Table 3.3: Confirming evidence of the case study

m1 **m2**	{**hostile, neutral**} **0.7**	Θ **0.3**
{**hostile**} **0.5**	{hostile} 0.35	{hostile} 0.15
{**neutral**} **0.1**	{neutral} 0.07	{neutral} 0.03
Θ **0.4**	{hostile, neutral} 0.28	Θ 0.12

Each set intersection is followed by its numeric mass product. The entries in the table are calculated by cross-multiplying mass products of rows and columns. This leads to a new evidence measure:

$$m_1 \oplus m_2\,(Z) = \begin{cases} 0.12 & \text{if} \quad Z=\Theta \\ 0.50 & \text{if} \quad Z=\{\text{hostile}\} \\ 0.10 & \text{if} \quad Z=\{\text{neutral}\} \\ 0.28 & \text{if} \quad Z=\{\text{hostile, neutral}\} \\ 0 & \text{else} \end{cases} \tag{3.7}$$

In this case the belief ranges from a 0.50 that the identification of the target is hostile to (0.50+0.12+0.28=0.9) that the identification of the target might be hostile. For this example the evidential interval is [0.5,0.9].

In Figure 3.4 some common evidential intervals are shown. In this overview Bel indicates the belief and Pl indicates the plausibility.

Evidential Interval	Meaning
[1;1]	Completely true
[0;0]	Completely false
[0;1]	Completely ignorant
[Bel,1]	Tends to support
[0;Pl]	Tends to refute
[Bel;Pl]	Tends to both support and refute

Figure 3.4: Some common evidential intervals

3.4.2 The Bayesian Belief Network

Bayesian Belief Networks are a network-based framework for representing and analysing models involving uncertainty. Bayesian Belief Networks (BBN) come from the cross disciplines of probability, artificial intelligence, and decision analysis. Bayesian Belief Networks exploit conditional independence relationships to create natural and compact domain models, thereby supporting useful reasoning patterns. The key feature of Bayesian belief networks is that they enable us to model and reason about uncertainty. Typically in Bayesian Belief Network modeling, we assign a Bayesian belief value to each uncertain event, such as the belief value of "It is cloudy"is assigned as 0.55, the belief value of "John is late" is 0.2, etc. All these probabilities for the uncertain events come from peoples subjective judgments which are determined by collecting empirical, historical or statistical data. In our case these probabilities are decided by domain experts. In BBN modeling, the probability representations of uncertainties are assigned as the prior belief values to each node in a certain Bayesian Belief Network.

A Bayesian Belief Network consists of the following:

- A set of variables and a set of directed edges between variables;
- Each variable has a finite set of mutually exclusive states;
- The variables together with the directed edges from a directed acyclic graph (DAG).

The general probabilistic inference problem is to find the probability of an event given a set of evidence. This can be done in Bayesian Belief Networks with sequential applications of Bayes rule.

Having entered the probabilities we can now use the Bayesian belief network to do various types of analysis. The most important use of Bayesian belief networks is in revising probabilities in the light of actual observations of events. The totality of such episode-specific information is called evidence (e) and can be used to update the probability of conclusion (u) into $P(u)$ using $P(u \mid e)$ and $P(e)$. There are two possible ways to combine evidence: the first one are deterministic nodes, a deterministic node has its value specified exactly by the values of its parents, with no uncertainty. The second one are noisy logical relationships, such as *noisy and* and *noisy or*, the noisy gates are a generalisation of logical gates, where uncertainty is represented as a leak ($P(L)$). There are two possible methods to combine evidence using noisy logical relationships, first if both facts have to be true before the conclusion may be drawn the *noisyand* mode has to be used. Otherwise if every piece of evidence can lead to the conclusion the *noisyor* mode is used [16]. In the case of noisy and, the total belief can be calculated using the following formula:

$$p\,(u) = p\,(e_1)\,p\,(u\,|e_1\,) * p\,(e_2)\,p\,(u\,|e_2\,) * \ldots * p\,(e_n)\,p\,(u\,|e_n\,) * p\,(L) \qquad (3.8)$$

In the case of noisy or, the total belief can be calculated using the following formula:

$$p\,(u) = 1 - ((1 - p\,(e_1)\,p\,(u\,|e_1\,))\,(1 - p\,(e_2)\,p\,(u\,|e_2\,)) \ldots (1 - p\,(e_n)\,p\,(u\,|e_n\,))\,(1 - p\,(L))) \qquad (3.9)$$

The more episode-specific evidence we gather, the closer we come to the ideals where the conclusion agrees with the actual cause (e.g. the target's behaviour can be explained when knowing the target is friendly). The event for which we want to determine its conditional probability given the evidence is called the query.

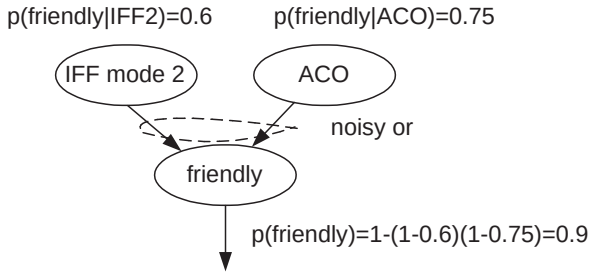

Figure 3.5: Decision making with a Bayesian network

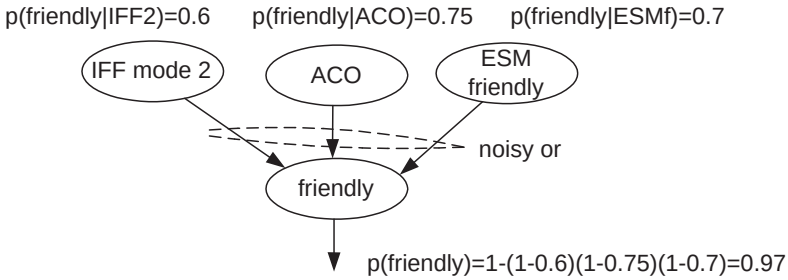

Figure 3.6: More evidential information in a Bayesian network

To explain the way evidence is combined in our Bayesian networks an example is shown in Figure 3.5 and 3.6. It becomes clear that the more evidence is gathered the more we are sure about the conclusion.

When we enter evidence and use it to update the probabilities in this way we call it propagation. Here, we see how the evidence effects the conditional probability distribution for this particular network. In general, there are four types of reasoning:

1. Causal reasoning is the pattern of reasoning that reasons from a cause to its effects;
2. Evidential reasoning is the reasoning from effects to its possible causes;
3. Mixed reasoning combines both causal and evidential reasoning;
4. Inter-causal reasoning involves reasoning between two different causes that have an effect in common.

If we look at the problem from the ships point of view we use evidential reasoning because we use the observed events to reason about its cause. For example the decision of a plane being hostile based on the visual observation of a hostile plane. But if we look at this problem from the pilot's point of view we already know that the plane is hostile and we use causal reasoning to explain the observed event. In our case we look at the environment from the ship's point of view and we use evidential reasoning.

3.5 Final choice

Regarding the last two sections we see that the problems with the classic Bayesian belief networks can be worked out. Therefore both methods seem to be equally suitable.

But if we take another look at the requirements we see that because of the requirement to make a real-time decision Dempster-Shafer is not suitable for this situation. There are a lot of facts which have to be combined into one solution, this will take a lot of time to compute. Bayesian belief networks take less time to compute and are because of the graphical representation easy to explain to the user.

Based on these arguments we have chosen to use Bayesian belief networks in this model.

Chapter 4

Overview of the entire system

4.1 Introduction

In the previous chapters the knowledge is described that can be used to reason about the current situation. More detailed information about classification and identification can be found in the following books, papers and internet sites [5], [15], [17], [23], [26], [37] and [38]. The complete system consists of three parts, first we gather all necessary information, then we derive as much facts as possible and to conclude we reason with these derived facts to decide on the classification and identification of the target. These three parts are shown in Figure 4.1 and will be worked out in the following chapters. The reasoning process will be modeled in two phases. In the first phase we will design an overall model to reason with the available information. In the second phase we will make clear how the necessary facts can be derived from the sensor information. In this chapter the first phase of the reasoning model will be explained.

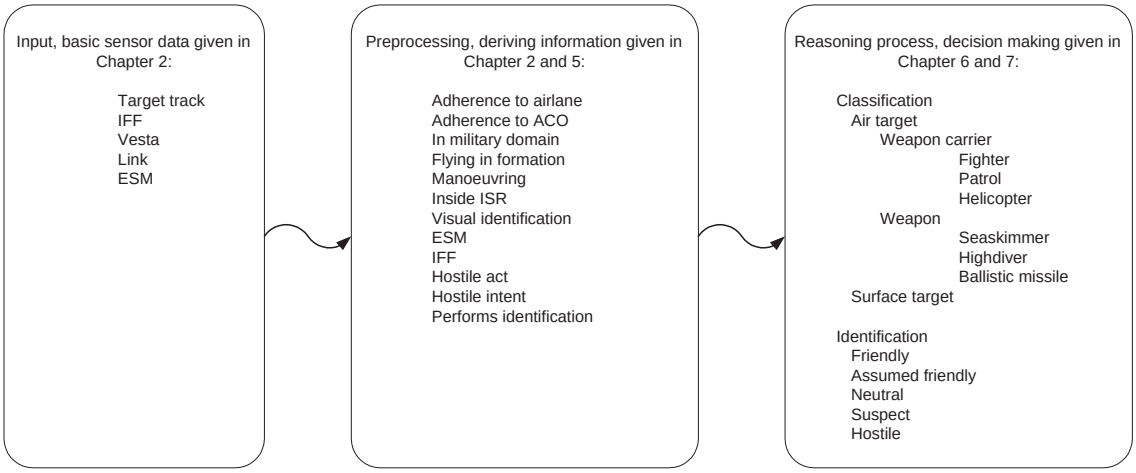

Figure 4.1: An overview of the entire system

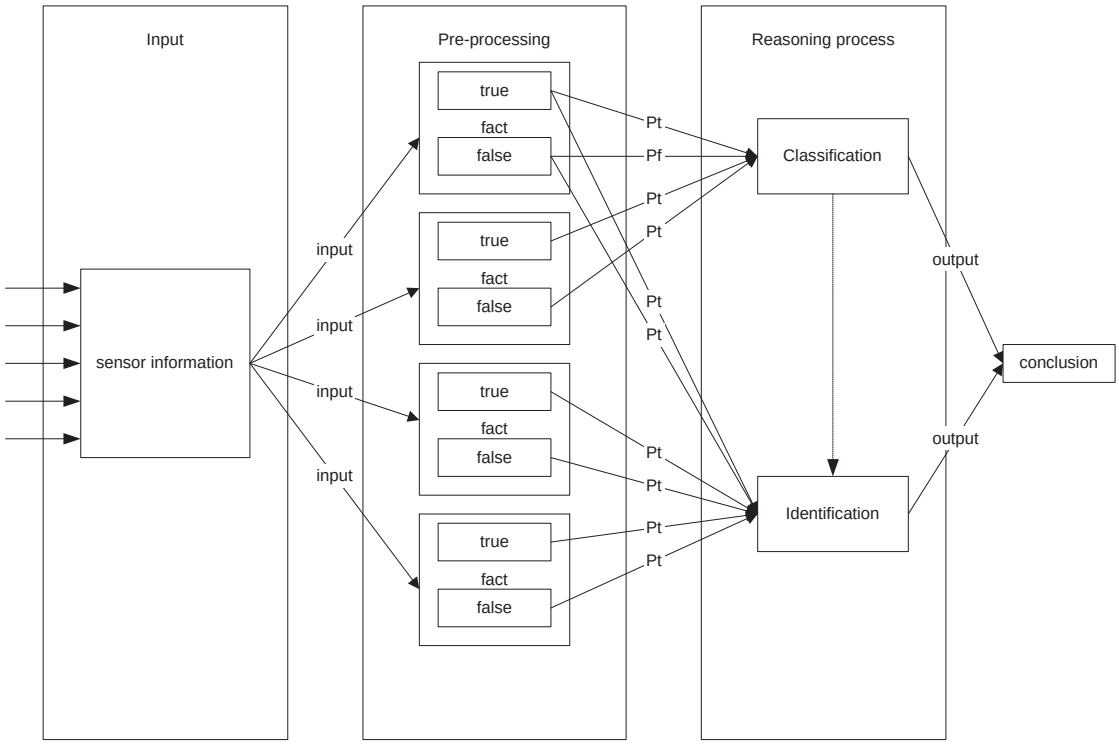

Figure 4.2: An overview of the reasoning process

4.2 The reasoning model

To make the data flow in the model clear an overview of the reasoning process is given in Figure 4.2. In this figure the overall structure of the model can be seen. Sensor information gives a representation of the existing situation of surrounding targets. For each target an evaluation is made based on this diagram. The received sensor data is in some cases compared to databases to evaluate if the facts are true or false. Some of these facts can be used to reason about the classification, others about the identification. For the identification reasoning process the classification conclusion can be used as input in some cases. The following example will make clear that classification can be important for the identification of a target.

If a ship is approaching and classification is done correctly, the classification will be deepened like, it is a surface target, it is a ship, it is a frigate, it is an M-frigate, it is HNLMS Van Galen. If all this information is collected it is easy to identify the target as a friendly unit, because HNLMS Van Galen is a Dutch frigate.

A more detailed overview of the classification is given in Figure 4.3 and of the identification in Figure 4.4.

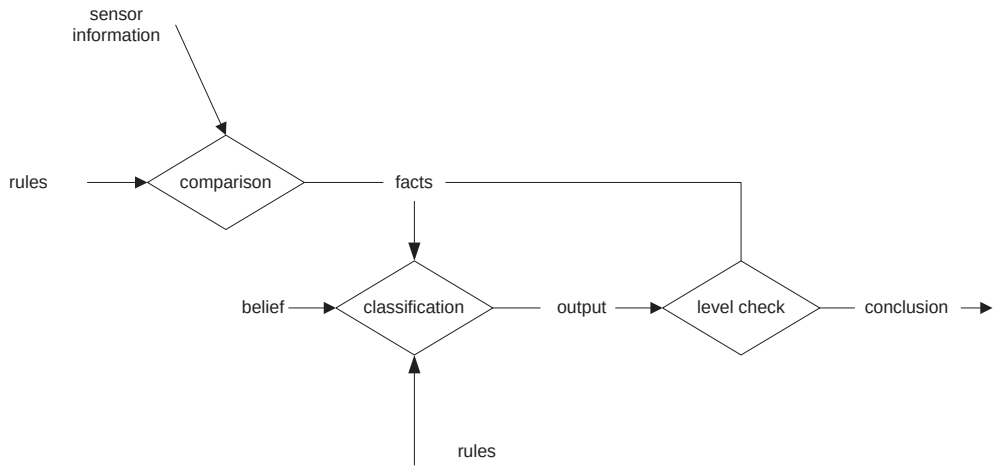

Figure 4.3: A detailed overview of the classification reasoning process

In the classification process the sensor information is compared to several databases. For all targets the facts which influence the classification are evaluated. The first time the diagram is passed the highest level is evaluated; air target or surface target. Each time the diagram is passed the system will try to deepen the information level. If the target is classified up to class/type name the classification has reached the deepest level.

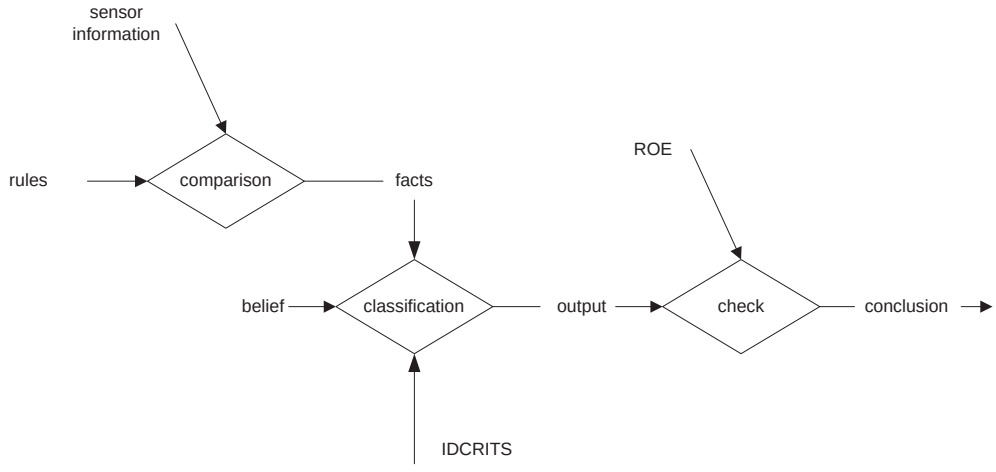

Figure 4.4: A detailed overview of the identification reasoning process

In the identification process the sensor information is compared to several databases. For all targets the facts which influence the identification are evaluated. Using these facts together with the belief values and the IDCRITS Bayesian belief networks can be generated, these are given in Chapter 6. All targets are evaluated in sequence, first the target that will reach the ship first, then the next and so on. In this way a new target is compared to other targets to evaluate if the targets are in formation. If a target is evaluated an output is generated, but before the output is made public the system has to check if all demands of the ROE have been covered. These ROE are partially covered by IDCRITS, but by using Bayesian belief networks with a *noisy or* combination not always a high probability for a certain conclusion implies that the ROE are satisfied. It may be possible to gain a lot of substantial evidence which leads to a high probability but the ROE are not satisfied. In that case a pending identification should be assigned.

Chapter 5

Formalisation of the preprocessing

5.1 Introduction

To make the model work in a simulation the derivation of facts has to be formalised. In this chapter first an overview is presented of all facts that have to be derived from the sensor information and how this derivation can be done is explained shortly. For each part of the reasoning process all facts which may influence the decision are mentioned and the reasoning process is formalised.

5.2 Derivation of information

In the real situation the decision making process is done by humans, they are experts in deriving facts from given information. In our model rules are necessary to do the same job. As mentioned in Section 2.1.1 and Section 2.1.2 certain information is necessary for the classification and identification of a target. Here follows a short description how these facts can be derived using sensor data:

- *Adherence to airlane:* compare the target track to the references of civil airlanes to determine if the target is moving in an airlane. This will be measured in the model as follows:
  - The altitude of the target has to be part of a fuzzy set of the limits of the airlane.
  - The position of the target has to be part of a fuzzy set of the limits of the airlane.
  - The heading of the target has to be part of a fuzzy set of the limits of the airlane.
  - The velocity of the target has to be part of a fuzzy set of the limits of the airlane.

- *Adherence to ACO:* if the target is moving in an airlane, analyse the track to determine if the target is moving according to a known flight plan. This will be measured in the model as follows:
  - The altitude of the target has to be part of a fuzzy set of the flight-plan's altitude.
  - The position of the target has to be part of a fuzzy set of the flight-plan's position at that time.
  - The heading of the target has to be part of a fuzzy set of the flight-plan's heading.
  - The velocity of the target has to be part of a fuzzy set of the flight-plan's velocity.
- *In military domain:* compare the target position with the military air domain.
- *In formation:* compare tracks of neighbouring targets, if moving in same direction with little separation the targets are in formation. This will be measured in the model as follows:
  - The distance of the target to the neighbouring target has to be part of a fuzzy set around zero.
  - The heading of the target has to be part of a fuzzy set of the heading of the neighbouring target.
  - The velocity of the target has to be part of a fuzzy set of the velocity of the neighbouring target.
- *Manoeuvring:* if the current track differs from the track history the target is manoeuvring.
- *Inside ISR:* compare the target range to the ISR.
- *Visual identification:* if the target is in visual range of an ally, identification takes place. The visual features lead to a type of target.
- *ESM identification:* if the target is in ESM range of an ally, identification takes place, if the ESM signature matches a known radar type.
- *IFF mode 2:* if IFF response, mode 2.
- *IFF mode 3:* if IFF response, mode 3.
- *IFF mode 4:* if IFF response, mode 4.

- *Hostile act*[1]*:* penetrating NATO secure area, intentionally impeding NATO/NATO-led operations, breaching or attempting to breach the security of a NATO/NATO-led military installation or restricted area. This will be measured in the model somewhat different. In the model the following actions are called a hostile act:
  - there has been an attack; or
  - the target enters the ISR without performing identification; or
  - the target is aiming an illumination radar at the vessel.
- *Hostile intent*[2]*:* capable, prepared and intention to inflict damage. This will be measured in the model as follows, a target has a hostile intent if it is closing in on the vessel and aiming a radar device at the vessel.
- *Performs identification:* compare manoeuvring to known identification sequence.

The fuzzy sets mentioned in the list above are a very simple way to determine with uncertainty if the target satisfies the given facts. For example take the boundaries in altitude for a certain airlane as between 4 and 5 kft. In that case the fuzzy set of the altitude will look like Figure 5.1

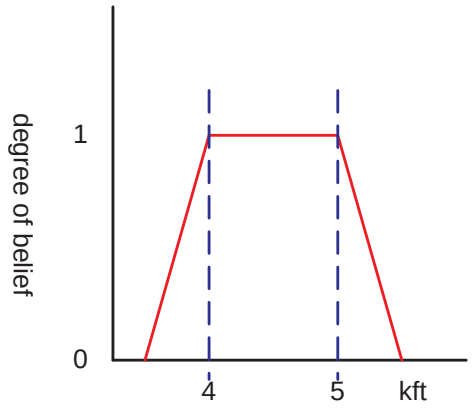

Figure 5.1: Fuzzy set for the altitude of an airlane

It is easy to be seen that if the altitude of the target is below 3.5 or above 5.5 kft the target is not in the airlane but if the altitude for example is 3.75 kft the probability that the target is in the airlane is 50 %.

If the ESM is unique for friendly/hostile targets information is gained, if the ESM is non-unique identification depends on the parties owning that type of vehicle. All facts that are evaluated can have three outcomes. The fact is true, the fact is false or there is not yet enough information about the fact.

[1]see Appendix A
[2]see Appendix A

Chapter 6

The Reasoning Model

6.1 Classification

The classic way of dealing with the classification problem is addressed in Section 2.1.3. In our model targets are divided in a number of mutual exclusive groups in several layers. If a target is detected it is classified in the first layer containing two groups:

- air target
- surface target.

In the next iteration an air target will be classified in the second layer if possible, a surface target is not worked out in this report. In the second layer a little more information can be given about the type of target, a weapon is discerned from a weapon carrier. In the final layer the sort of weapon or platform will be identified. There is no difference between civil and military targets in the classification. If necessary the difference between civil and military can be made based on the identification.

In the classification process we will use iterative deepening to evaluate the target gradually. First we examine the target being an air target or a surface target. How the classification tree is organised is worked out in Figure 6.1 and 6.2. The classification in the way shown in Figure 6.1 might be very difficult, because the UAV group is not mutually exclusive with the two other defined groups and the same for the ARM and CM missile. There are many different UAV's, different in size, speed, mission and it can be fixed wings as well as helicopters. The ARM and CM come in many different forms, they have no real distinctive features. Therefore the second division may be better, but from literature only there cannot be made a sensible conclusion about the feasibility of these two versions. For this report we will use the second tree, because the Bayesian models will be more complete.

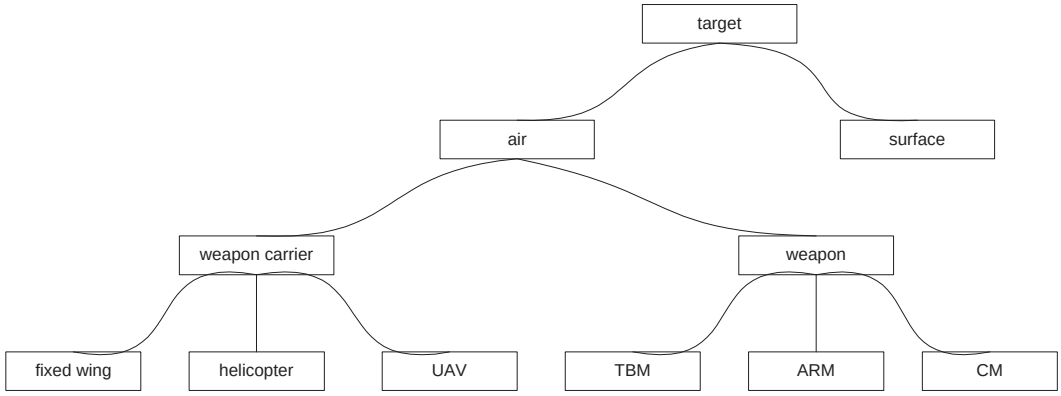

Figure 6.1: Classification by way of propulsion and weapon type

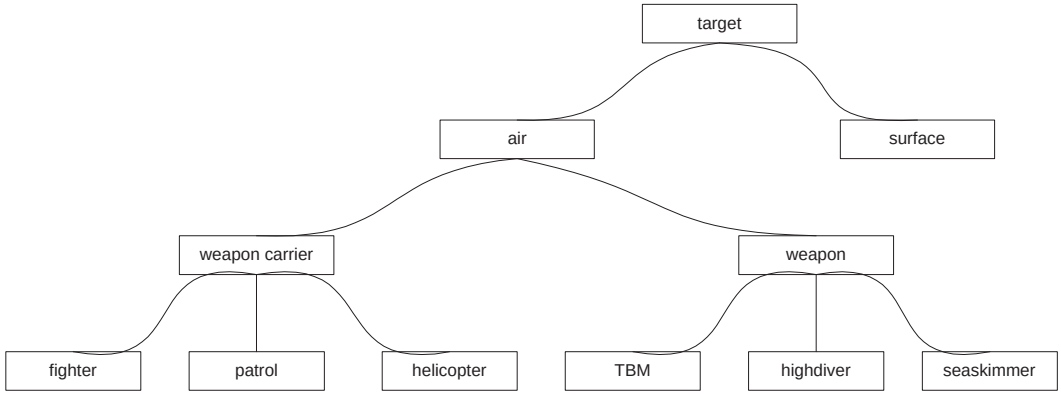

Figure 6.2: Classification by threat

The Bayesian belief networks for the different layers make clear which information can be used to distinguish between the different groups. This information can be found in the following book and internet site [8] and [42], further information about distinguishing features we gathered by interviewing experts at the OPSCHOOL. In Figures 6.3 and 6.4 The Bayesian belief networks for the first layer are given, in these figures a noisy or is used to complement the evidence. The numbers displayed in the figures are gathered by interviewing several experts at the OPSCHOOL. The interviewed experts gave similar beliefs to the same relation, these answers were combined into the used values.

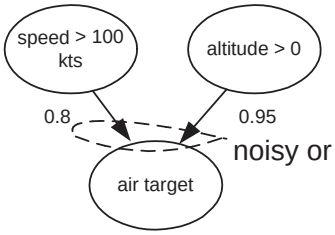

Figure 6.3: Bayesian belief model of an air target

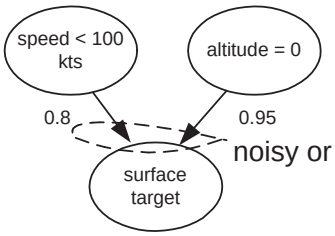

Figure 6.4: Bayesian belief model of a surface target

In the second layer there has to be made a difference between a weapon or a weapon carrier. In Figure 6.5 and 6.6 the Bayesian belief networks are shown for this examination. In the following iteration more information will be given about the target.

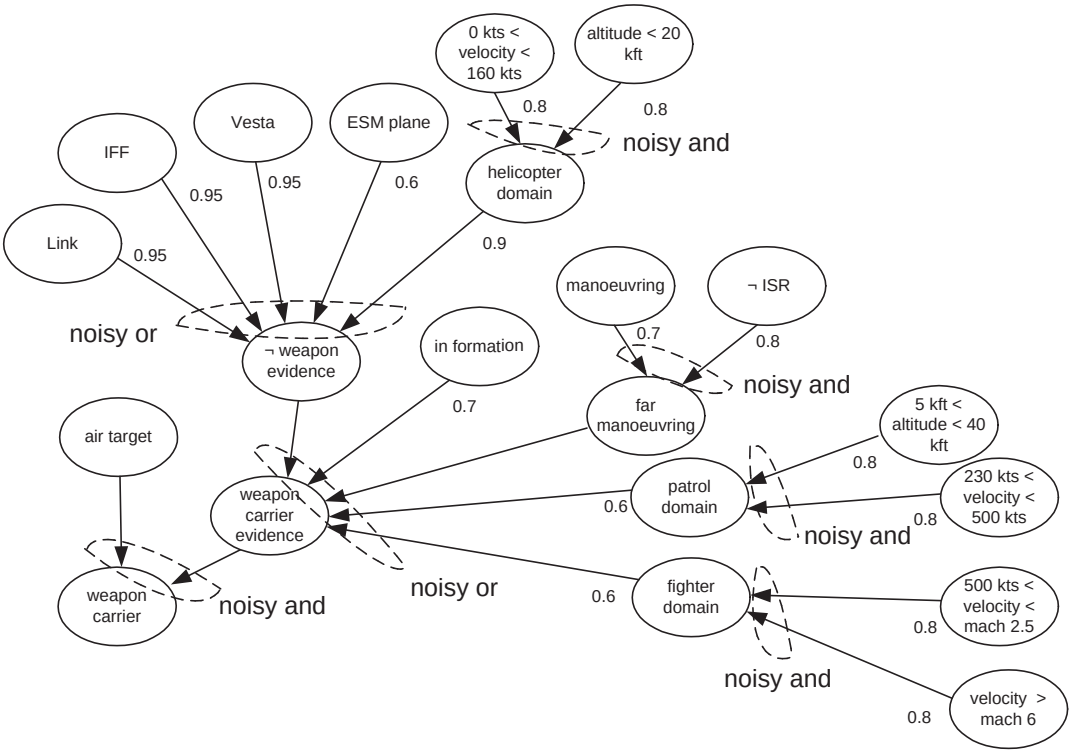

Figure 6.5: Bayesian belief model of a weapon carrier

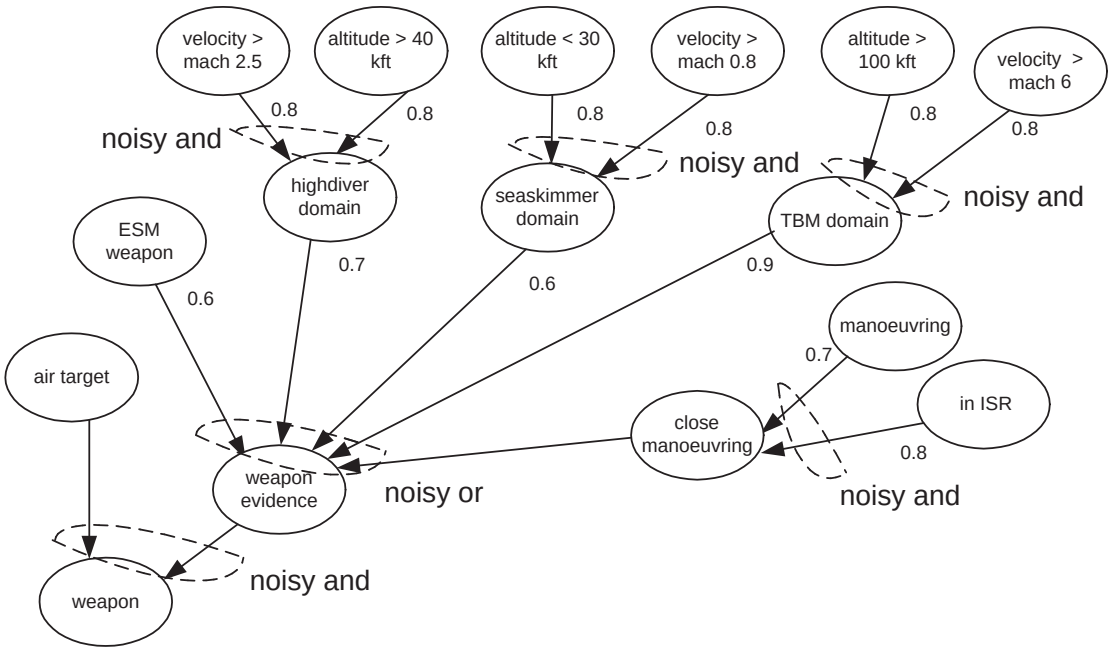

Figure 6.6: Bayesian belief model of a weapon

In the third and last layer there are six groups, but depending on the outcome of layer two they are divided in two groups. If in level two the probability of a weapon carrier is the highest, there are three possible outcomes for level three, it could be a helicopter, a patrol aircraft or a fighter. The Bayesian belief networks for this decision can be found in Figure 6.7, 6.8 and 6.9. Based on all gathered information every option is evaluated and the one classification which has the highest probability will be chosen in this evaluation the classification history will be used as a guideline in the new classification.

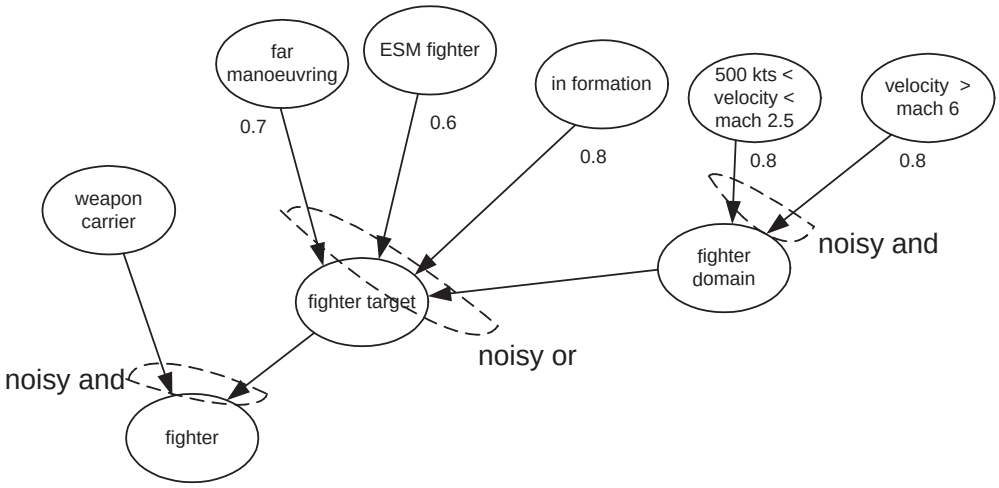

Figure 6.7: Bayesian belief model of a fighter

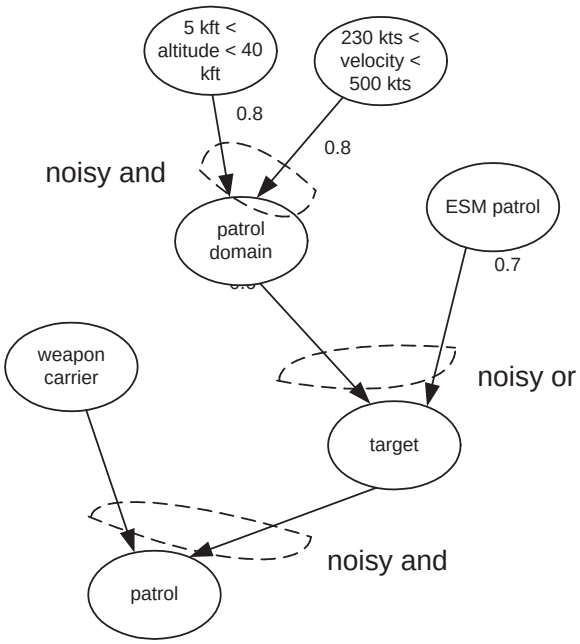

Figure 6.8: Bayesian belief model of a patrol aircraft

If in level two the probability of a weapon is highest there are also three possible outcomes for level three, the target could be a seaskimming missile, a highdiving missile or a TBM. In Figure 6.10, 6.11 and 6.12 the Bayesian belief networks for this decision are illustrated.

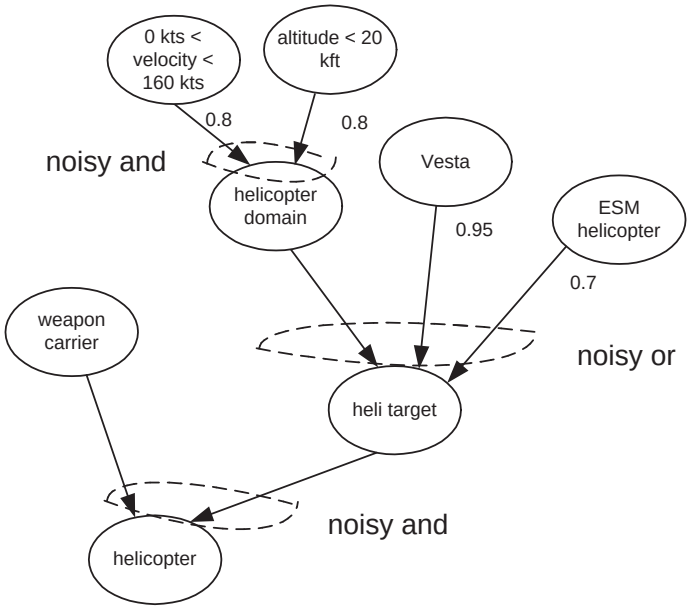

Figure 6.9: Bayesian belief model of a helicopter

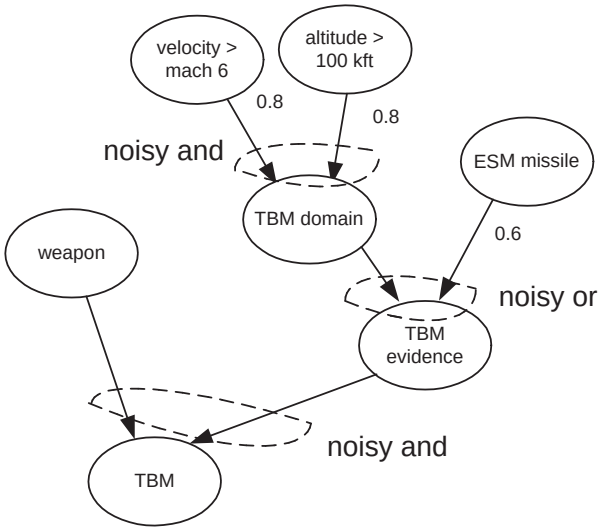

Figure 6.10: Bayesian belief model of a TBM

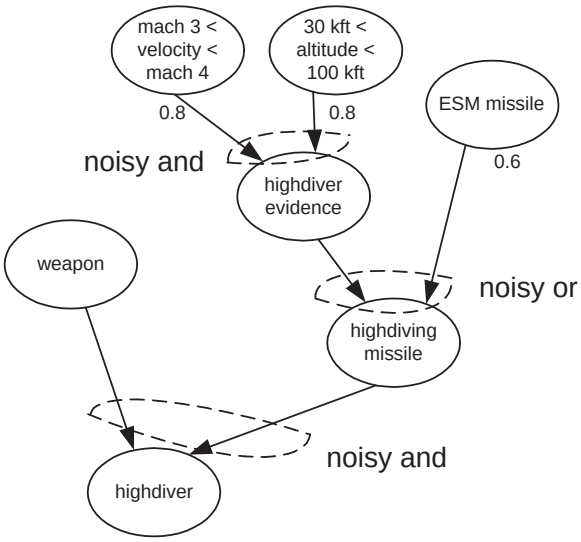

Figure 6.11: Bayesian belief model of a highdiving missile

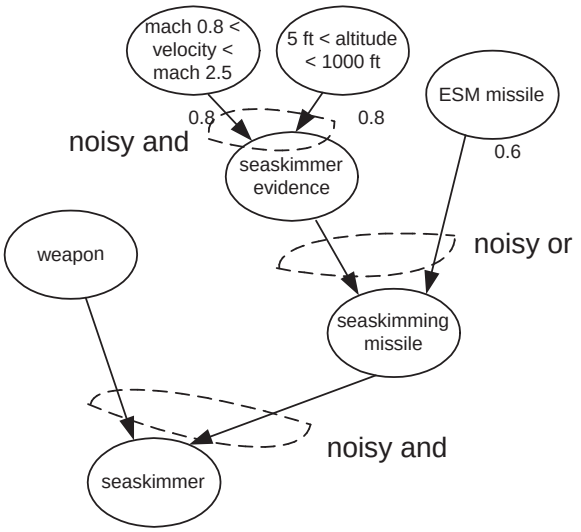

Figure 6.12: Bayesian belief model of a seaskimming missile

6.2 Identification

In the identification process all possible identifications are evaluated and the one that has the highest probability will be chosen if all constraints are satisfied. These constraints are the ROE, e.g. if the ROE in force tells to visually identify a target before calling it hostile or friendly, than how strong the belief may be that the target is hostile it is identified suspect until the target is visually identified hostile. In the following Figures 6.13 until 6.17 the Bayesian belief networks are given for the evaluation process. If the probability of all identities is almost equal, the status will be set to unknown.

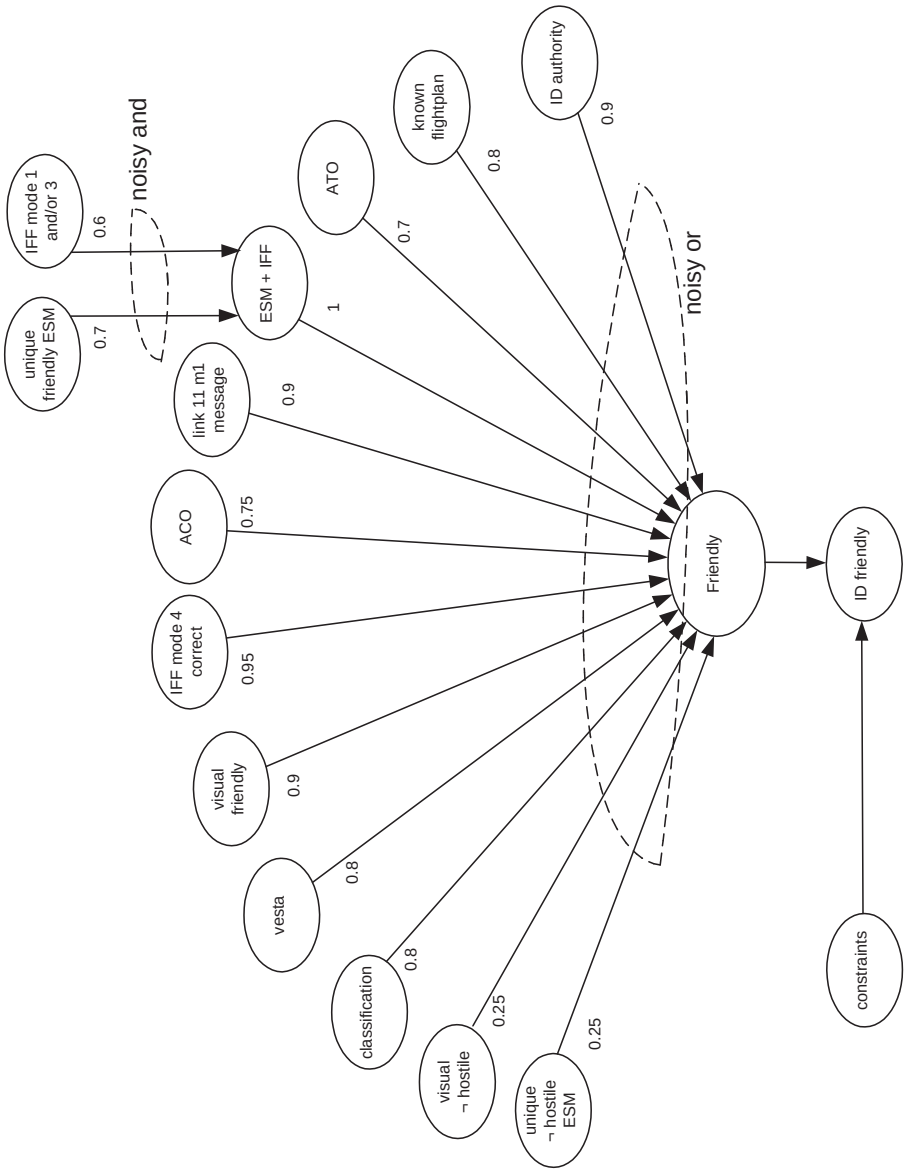

Figure 6.13: Bayesian belief model for a friendly identification

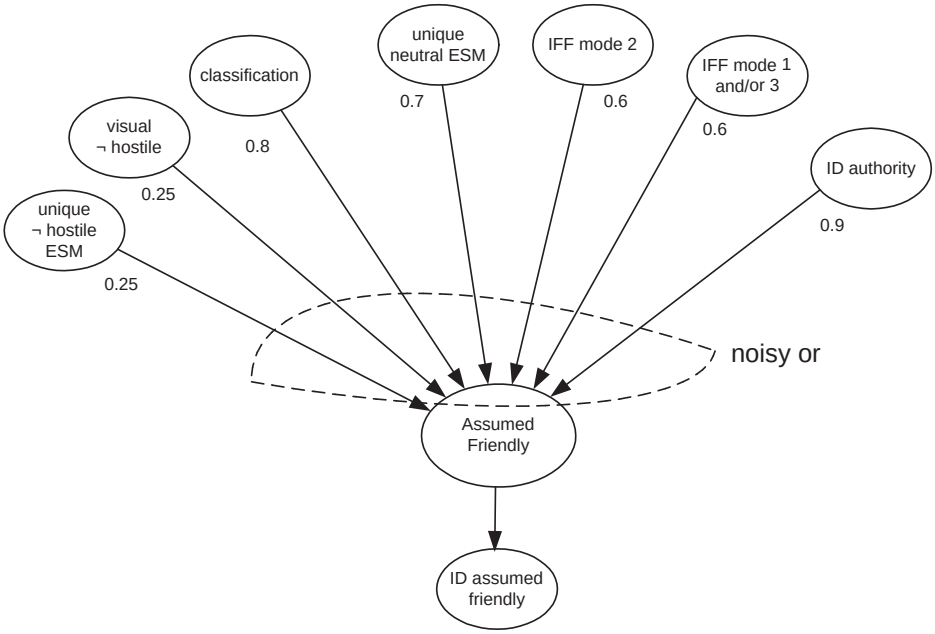

Figure 6.14: Bayesian belief model for an assumed friendly identification

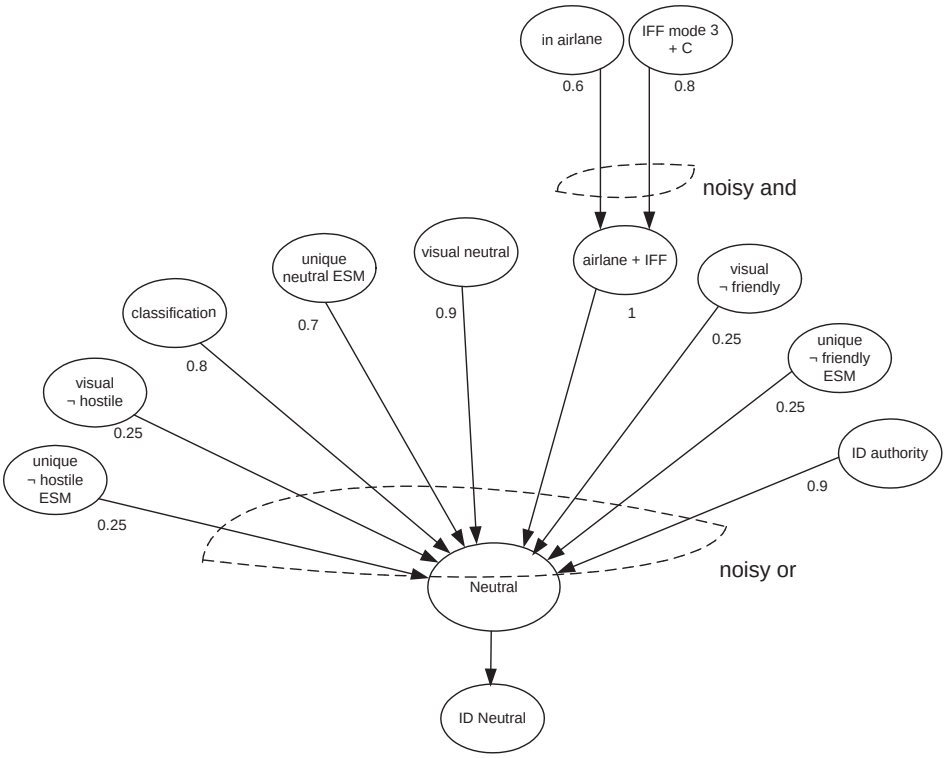

Figure 6.15: Bayesian belief model for a neutral identification

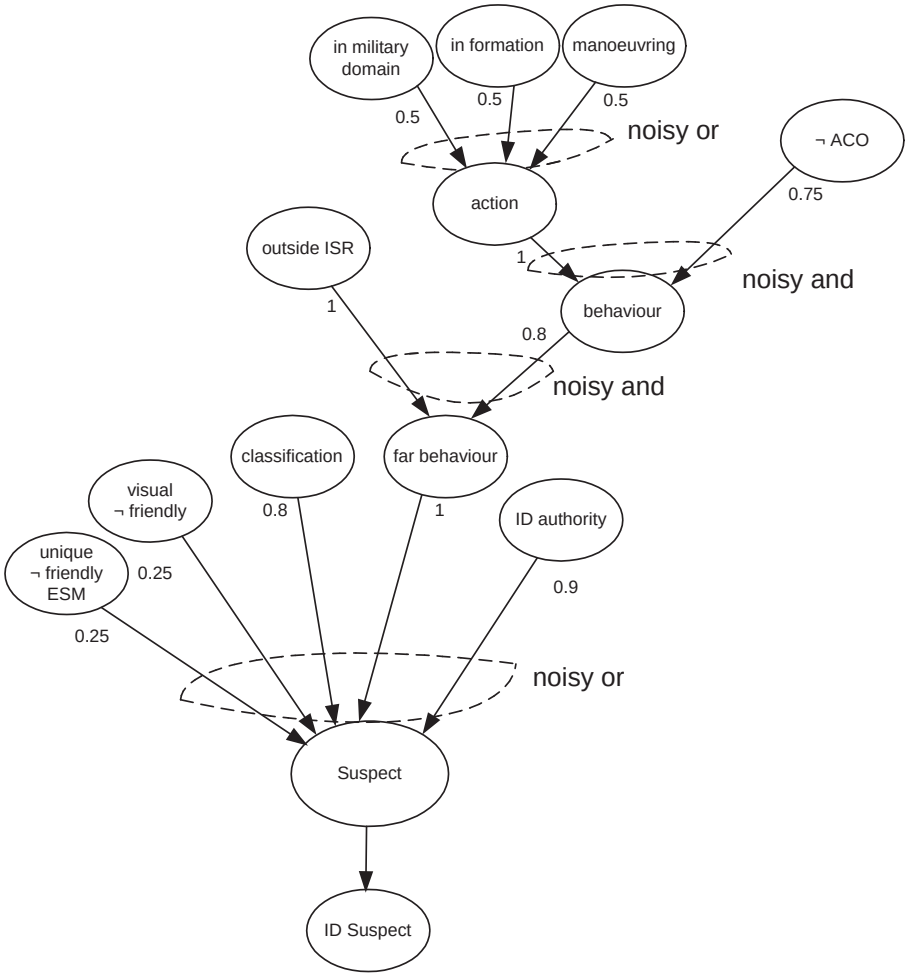

Figure 6.16: Bayesian belief model for a suspect identification

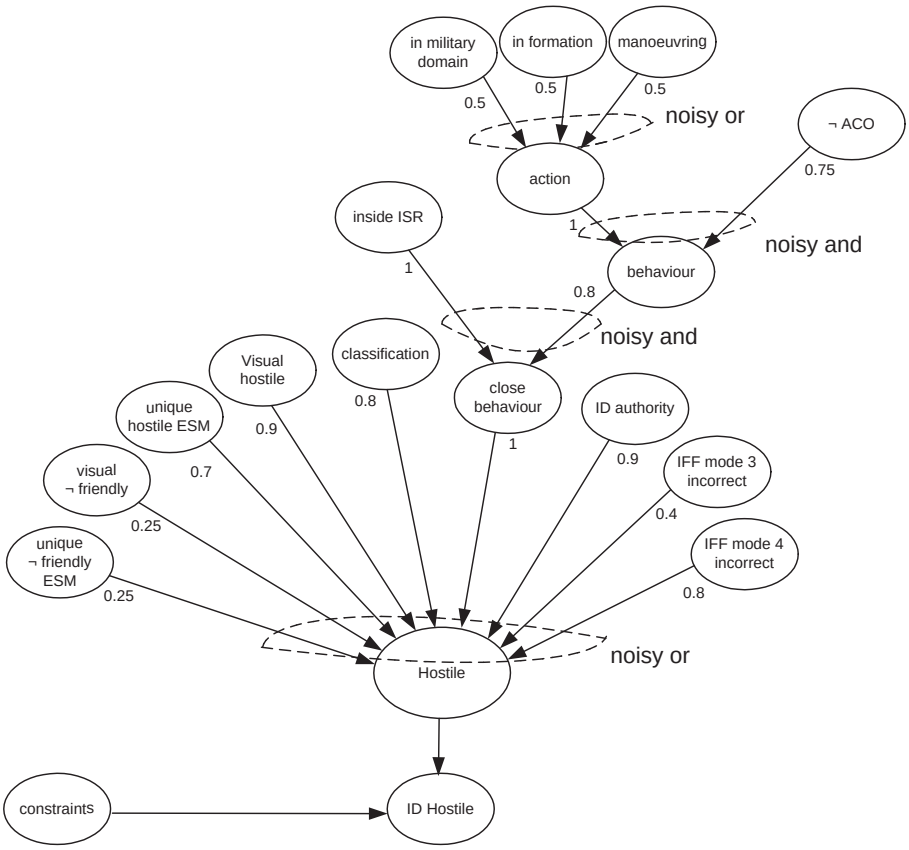

Figure 6.17: Bayesian belief model for a hostile identification

6.3 Uncertainty of the input data

In sections 3.4.2 and 5.2 we have already indicated that in real life we can not always be completely sure about the data we get from the sensors or about the information that we derive from sensor data. For each fact that is an input for a Bayesian belief network we could calculate the probability that the fact is true and use this probability in calculations that are done by the Bayesian belief network as described in the formulas for *noisy and* and *noisy or* in section 3.4.2. However because we use sensor data that is stored in an XML file for our prototype we *can* be sure of the facts that are input for the Bayesian belief networks and the probability of those facts will always be 1 for our prototype.

Next to the kind of uncertainty that is caused by wrongly deriving or observing facts we could also take the uncertainty into account that is caused by information that is not observed at all. This uncertainty could be represented by introducing a leak in the *noisy and* and *noisy or* gates as described in section 3.4.2. Initially we would set this leak to 0.95, because this is the standard confidence interval in Gaussian probability density functions. However in our simulation the entire situation will be known and therefore no information will be missed. Because of this we set the leak to 1.0 for all noisy gates.

Chapter 7

Temporal reasoning

7.1 Introduction

As a result of the previous study an obvious question was, is it possible to reason about the classification and identification of a target in time? Before we are able to answer this question, we must be sure about the meaning of reasoning in time. Reasoning about events that depend on time is called temporal reasoning and is something humans can do fairly easy. However, it is difficult to formalise temporal events so that the computer can make temporal inferences. Temporal reasoning is a variation of the reasoning processes mentioned before. Instead of one set of facts on one certain timepoint temporal reasoning is meant to recognise processes and events in time. In this thesis there are a number of processes which may be reasoned with in time, because a number of observations is available in time. These processes are for example, the sensor readings, the decision process and the processes of deriving and combining information. In this chapter we will make clear in which processes temporal reasoning may be useful and introduce a number of possible methods to reason in time. A choice will be made about the method most promising for this situation.

7.2 Reasoning in time

As said before there are a couple of processes in which temporal reasoning may offer additional information. These processes are:

- Getting sensor data;
- Deriving information;
- Decision making.

We will take a look at these processes and explain the benefits of and problems with reasoning in time.

First the sensor information, in a lot of civil situations it is obvious that sensor information will give information about the target. If we look at an airport, the incoming planes would like the air traffic controller to know as exact as possible what kind of plane is coming and in which position the plane is at the moment. Because there is a limited set of approaches possible to each landing strip we expect to see a pattern in the sensor readings. While the plane is coming closer more detailed information can be given and one of the approaches gets more probable in time. In military situations we expect a little different situation. In a military environment we expect very limited information about approaching targets.

Hostile forces will try to give as little as possible information about themselves and sometimes try to give false information to mislead their opponents. Furthermore as long as we do not know what kind of target is approaching there are no strict rules about how the target will approach for example. So sensor readings will not be very predictable in time and reasoning in time will not give much more information.

Second we take a look at the process of deriving and combining information. As explained in Chapter 5 sensor data can be used to obtain more detailed information. For example, the heading and speed of a target may be derived from positions of the target in time. Therefore we have to take a look at each derived fact and determine if it is possible to use temporal reasoning. The facts we have to evaluate are:

- air target;
- surface target;
- weapon evidence;
- weapon carrier evidence;
- highdiver evidence;
- seaskimmer evidence;
- TBM evidence;
- patrol evidence;
- helicopter evidence;
- fighter evidence;
- behaviour:
  - in airlane;
  - according to ACO;
  - in military domain;
  - in formation;
  - manoeuvring;
  - in ISR;
  - visual;

  - ESM friendly / hostile;
  - hostile act / intent;
  - performs ID.

Most of these facts are not related in time, but others can't be evaluated at one timepoint, an example of a fact which can't be evaluated without the time aspect is manoeuvring. It is obvious that we are not able to tell whether a target is moving according to one position. Other facts like air target are absolutely not related to time, changes in the altitude of a target over time does not give us any additional confidence that a target is an air target.

If we evaluate all these facts we see that:

- air target, time related, the longer the target has an altitude the higher the probability it is an air target;
- surface target, time related, the longer the target has no altitude and a low velocity the higher the probability it is a surface target;
- weapon carrier evidence, not time related;
- weapon evidence, not time related;
- highdiver evidence, time related, the longer the speed and altitude match the highdiver pattern, the more likely it is a highdiving missile;
- seaskimmer evidence, time related, the longer the speed and altitude match the seaskimmer pattern, the more likely it is a seaskimming missile;
- TBM evidence, time related, the longer the speed and altitude match the TBM pattern, the more likely it is a ballistic missile;
- patrol evidence, time related, the longer the speed and altitude match the patrol pattern, the more likely it is a patrol plane;
- helicopter evidence, time related, the longer the speed and altitude match the helicopter pattern, the more likely it is a helicopter;
- fighter evidence, time related, the longer the speed and altitude match the fighter pattern, the more likely it is a fighter;
- behaviour:
  - in airlane, time related, the longer the target stays in the airlane, the more likely it is an airliner;
  - according to ACO, time related, the longer the target follows the ACO, the more likely it is the expected plane;
  - in military domain, time related, the longer the target stays in the military domain, the more likely it is a military target;
  - in formation, time related, the longer the targets stay in formation, the more likely this is no coincidence;
  - manoeuvring, time related, the target has to move a couple of times within a certain timespan;

- in ISR, not time related;
- visual, not time related;
- ESM friendly / hostile, not time related;
- hostile act / intent, time related, if it once performed a hostile act it stays hostile;
- performs ID, time related, the target has to perform a couple of actions in a preset sequence in time.

There are also some facts which can be derived from other facts in time, like the heading and speed can be derived from the positions of a target in time, but which can also be obtained directly from the sensor data. This means that these facts can also be derived if the sensors are malfunctioning or have been switched off.

The way a target moves (its behaviour) is the most distinctive feature between different sorts of targets. From the list above it shows that the behaviour of a target is time related, thus it may be useful to evaluate the behaviour in time.

Finally the decision will get more reliable in time because there will be more information available when the target has been followed for some time or the target has moved closer. The decision process will take care of the processing of this information into a proper decision, in which the process could take the decision at an earlier time point into account. In comparison to the benefits of temporal reasoning in the evaluation process the benefits in the decision process will be quite small.

7.3 Temporal reasoning methods

In time some theories about temporal reasoning have evolved, in this section we will give a short overview of the main methods with their advantages and disadvantages. We will explain which method is useful in this situation. More detailed information about the most commonly used approaches can be found in the following papers [13] and [31].

7.3.1 Hidden Markov Models

A hidden Markov model is just like a regular Markov model in that it describes a process that goes through a sequence of states. The difference is that in a regular Markov model, the output is a sequence of state names, and because each state has a unique name, the output uniquely determines the path through the model. In a hidden Markov model, each state has a probability distribution of possible outputs, and the same output can appear in more than one state. Hidden Markov models are called hidden models, because the true state of the model is hidden to the observer. In general, when we see that the output of a hidden Markov models is a particular symbol, we can not be sure what state that symbol came from.

To explain the general Hidden Markov basics we use the following notation:

Set of states	$S = \{s_1, \ldots, s_N\} = \{1, \ldots, N\}$, where N are the number of states;
Output alphabet	$K = \{k_1, \ldots, k_M\}$, where M is the number of observation symbols in the alphabet;
Initial state probabilities	$\Pi = \{\pi_i\}, i \in S$
State transition probabilities	$A = \{a_{ij}\}, i, j \in S$
Symbol emission probabilities	$B = \{b_{ijk}\}, i, j \in S, k \in K$
State sequence	$X = (X_1, \ldots, X_{T+1})$
Output sequence	$O = (o_1, \ldots, o_T)$

There are three fundamental questions that we want to know about Hidden Markov Models:

1. Given a model $\mu = (A, B, \Pi)$, how do we efficiently compute how likely a certain observation is, that is $P(O|\mu)$?
2. Given the observation sequence O and a model μ, how do we choose a state sequence $(X_1, \ldots, X_{T+1})$ that best explains the observations?
3. Given an observation sequence O, and a space of possible models found by varying the model parameters $\mu = (A, B, \Pi)$, how do we find the model that best explains the observed data?

The first question is about which model is the best, the second one lets us guess what path was probably followed through the Markov chain, and this hidden path can be used for classification, for instance in applications to part of speech tagging. The third question can be used to estimate the unknown parameters in the model.

Hidden Markov models are useful when one can think of underlying events probabilistically generating surface events. One widespread use of this is tagging - assigning parts of speech (or other classifiers) to the words in a text. We think of there being an underlying Markov chain of parts of speech from which the actual words of the text are generated.

7.3.2 Kalman filtering

Kalman filtering assumes that each state variable is real-valued and distributed according to a Gaussian distribution. That each sensor suffers from unbiased Gaussian noise and each action can be described as a vector of real values, one for each state variable. That the new state is a linear function of the previous state and action. These assumptions taken together, allow prediction and estimation to be implemented by some matrix calculations, even with a large number of state variables.

A linear system is a process which can be described by the following two equations:

$$x(k+1) = Ax(k) + Bu(k) + w(k) \tag{7.1}$$

$$y(k) = Cx(k) + z(k) \tag{7.2}$$

Here A, B and C are matrices, k is the time index. The state is represented by x and the input is given in u. The output is given as y. As mentioned above the Kalman filter assumes that process and measurement suffer white noise, these are given as w and z, where w is process noise and z is measurement noise. These equations are called the state-space process equations.

Based on these equations, the problem of finding a minimum variance estimate of the quantity x_k (the state) is the Kalman filter problem. The state contains all the information regarding the system at a certain point in time. This information should be the least amount of data one is required to know about the past behaviour of a system in order to predict its future behaviour. The Kalman filter is a computational scheme to reconstruct the state of a given state-space model in a statistically optimal manner, generally expressed as the minimum variance of the state reconstruction error conditioned on the acquired measurements. The conventional Kalman filter as it was originally derived by Kalman [12] is a recursive method to compute the minimum variance estimate of the state vector using the state error covariance matrix $P(k|k-1)$ given by:

$$P(k|k-1) = E[(x(k) - \hat{x}(k|k-1))(x(k) - \hat{x}(k|k-1))^T] \tag{7.3}$$

If we work out this equation we find a recursion for $P(k|k-1)$ [30]. In that way we find the solution to the optimum filter using the Kalman filtering technique is embedded in a non-linear equation called the Ricatti Equation . This equation can be solved recursively or iteratively. Utilising this recursive property is one of the main features of the Kalman filters since only the previous measurement needs to be stored to update the algorithm. The theoretical foundations used from statistics are quite complex. There has been a lot of research to provide simpler derivations and implementations of the same 'optimal statistical state observer'.

7.3.3 Dynamic Bayesian networks

Dynamic Bayesian networks is a term which can be explained in many different ways.

1 Some say a dynamic Bayesian network is a network which is dynamic in time [22], so the actual structure of the network may change over time.

2 Others say a dynamic Bayesian network is a regular Bayesian network in which some nodes have connections to nodes in another timeslice [3] and [18], as depicted in Figure 7.1.

3 Some state that a dynamic Bayesian network is a regular Bayesian network where some nodes have a temporal character [19] see Figure 7.2.

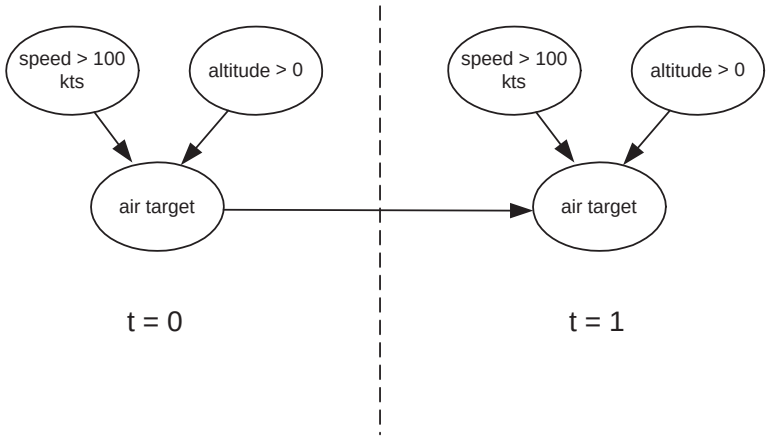

Figure 7.1: An example of intertimeslice connections in a DBN

Explanations 2 and 3 can be used in our model, if we take another look at the first part of this section we learn there are two sorts of behaviour in time we want to capture in the model. We can give an example of both methods in our model, the first is if we once measured an altitude of a target we can say that in the next time slice it is very well possible that the target has approximately the same altitude again and will still be an air target. On the other hand we have temporal information like the target is manoeuvring which we want to inject into a node.

7.4 Conclusion

Both methods described in the previous sections can be implemented as a Dynamic Bayesian network, how that can be done will be explained shortly, more detailed information can be found in [18].

As explained in Section 7.3.1 a hidden Markov model has states which contain a probability distribution for a certain set of outputs possible from that state. This is very similar to the nodes in a Bayesian network. The possible transition from one state to the next in a hidden Markov model is given by a

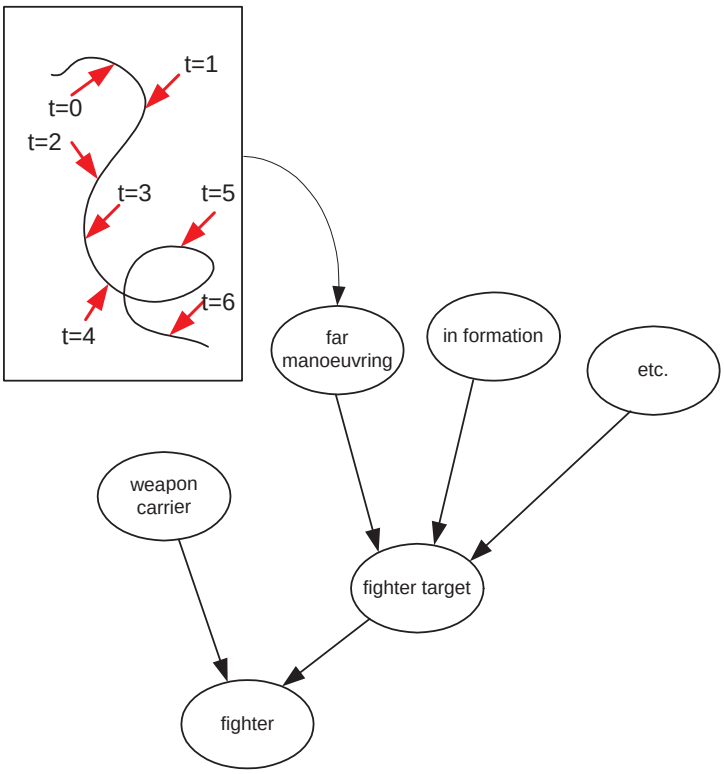

Figure 7.2: An example of temporal input in a DBN

probability, which can be modeled as a conditional probability in the dynamic Bayesian network.

A Kalman filter is modeled as a current state which can be given as a matrix with all information of this time point. This matrix can be seen as an overview of conditional connections between variables in time. In this way all zero's in the matrix mean there is no connection and therefore there is no relation between the two nodes in the dynamic Bayesian network.

Furthermore we saw that the temporal aspects in our model could be represented in a Dynamic Bayesian network in two ways depending on the kind of relation. Therefore the choice will be to use Dynamic Bayesian networks to implement temporal relations in our model.

Part III

The implementation

Chapter 8

The UML model of the prototype

8.1 Introduction

After the reasoning process has been described we make the model concrete using Unified Modeling Language (UML). This method is commonly used to design Object-Oriented software. The abstraction level of the model will be slightly higher than the program itself. Because the more detail is added to the design, the more the model has to change during implementation. For those people who have never seen an UML model before, a short introduction will be given in the next section, more information can be found in [24] and [39]. Four useful diagrams for showing the functionality and structure of the model are introduced in this section. In this chapter we will also show the diagrams that were created for this project and the way they should be interpreted is explained shortly.

8.2 UML overview

It is possible to draw several diagrams with UML, that all explain one aspect of the model. Because this report does not use all possible UML diagrams, not all diagrams will be worked out in this section. The following diagrams are used in this report.

Use Case diagram:

A use case diagram is a description of the program from the user's point of view. A use case diagram consists of actors, which can both be a real user or any external process, and use cases. The use case represents tasks, which can be performed by the actor. An example of a use case is given in Figure 8.1.

Class diagram:

A class diagram shows the classes that are part of the program and the relationships between those classes. It can show inheritance between classes for

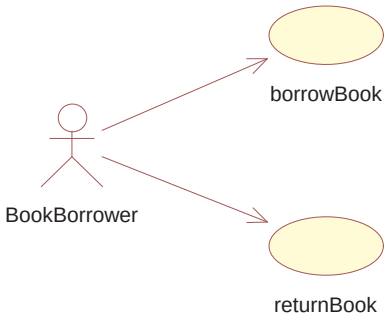

Figure 8.1: An example of a use case diagram

example. It can also show the methods and the attributes of the classes.

To clarify the meaning of our classes even more we also create some CRC cards for some of the classes. CRC stands for Class, Responsibilities and Collaborations. In a CRC card the functionality (or responsibility) of a class can be described in natural language and the classes that this class collaborates with to perform its tasks can be specified. CRC cards are not part of UML but provide a nice way to explain the functionality of a class. In a class diagram an arrow from one class to another mean that the class at the end of the arrow is an attribute of the class at the beginning of the arrow and is used by that class. If there is an * at the end of the arrow this means that the class at the beginning of the arrow can contain more than one instance of that class. Inheritance can also be shown in a class diagram. An arrow with a closed head from the child class to the parent class shows this. This is illustrated in the Figure 8.2, where Child is extended from Parent and uses multiple instances of the Helper class.

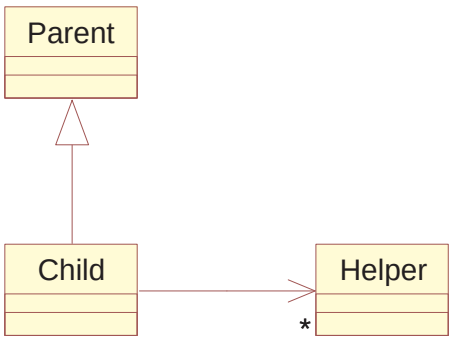

Figure 8.2: An example of a class diagram

Collaboration diagram:

Collaboration diagrams are a kind of interaction diagrams. They show the interaction and communication between instances of classes. The instances are represented by a rectangle and the messages that are exchanged between Arrows between those rectangles represent those instances. Collaboration diagrams provide a good way to get a clear understanding of how the different elements of a program co-operate to provide the required functionality.

Sequence diagram:

A sequence diagram is another kind of interaction diagram. It contains almost the same information as a collaboration diagram and it can be argued that drawing a sequence diagram is not necessary when a collaboration diagram has already been created. However because the information is presented differently in a sequence diagram as it is in a collaboration diagram it can still be useful to create a sequence diagram.

Where a collaboration diagram is very good at showing the relations between the class instances, a sequence diagram concentrates on the time order of the communication messages between the instances. In a sequence diagram the class instances are represented by rectangles, just like in a collaboration diagram. But where the instances can be placed anywhere in a collaboration diagram in a sequence diagram they are usually placed in a horizontal line. Every instance has a lifeline, which is represented by a vertical line that goes down from the instance rectangle. When one instance calls a method of another instance an arrow is drawn from the lifeline of the calling instance to the lifeline of the called instance. If due to this call the called instance calls a method of another instance (or one of its own methods), another arrow is drawn from lifeline to lifeline below the previous arrow to indicate that this call has occurred later than the previous call. An example of a sequence diagram is shown in Figure 8.3

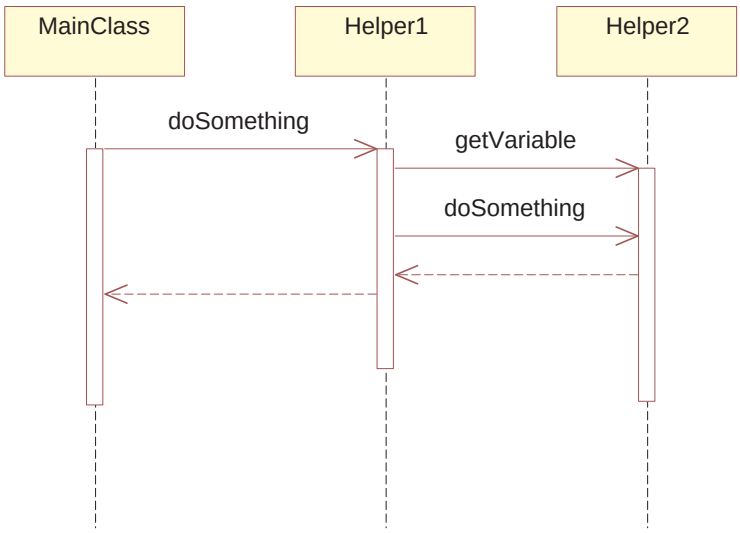

Figure 8.3: An example of a sequence diagram

8.3 The UML Model

The TIC program consists of three packages, which are shown in Figure 8.4. In the javabayes package all classes for the in- and output to and from Bayesian belief networks can be found. In the gui package all classes for the graphical user interface can be found and in the main package all rules for the reasoning process can be found.

Figure 8.4: Package overview of the entire model

8.3.1 The use case diagram

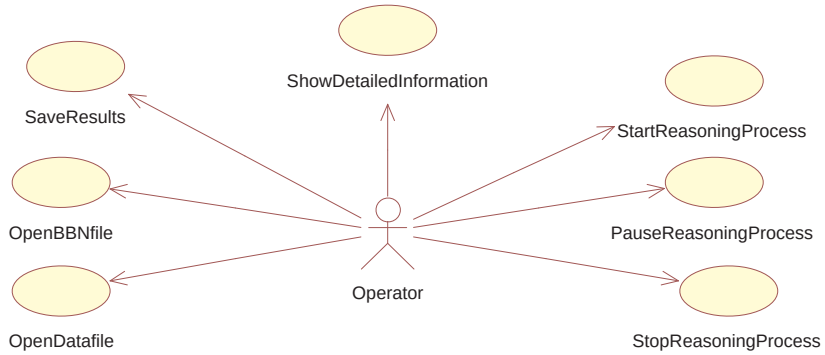

Figure 8.5: Use case diagram of the entire model

The use case for this project is quite simple because there is just one actor in the system, which is the operator. The operator can ask for the following actions to be performed:

- load a Bayesian belief network file;
- load a situation data file;
- start the reasoning process;
- pause the reasoning process;
- stop the reasoning process;
- show detailed information;
- save reasoning results.

The diagram showing these actions is visualised in Figure 8.5

8.3.2 The class diagram

A full class diagram may be difficult to interpret because it gives a lot of information about the contents of the classes from which the functionality of the class is not directly evident. Therefore we created a class diagram with empty classes. Also we have created some Class, Responsibility and Collaboration (CRC) cards that describe the responsibilities of the classes in natural language. We will show the class diagrams of the gui and the main model first in Figure 8.6 and 8.7, then the CRC cards in Figure 8.8.

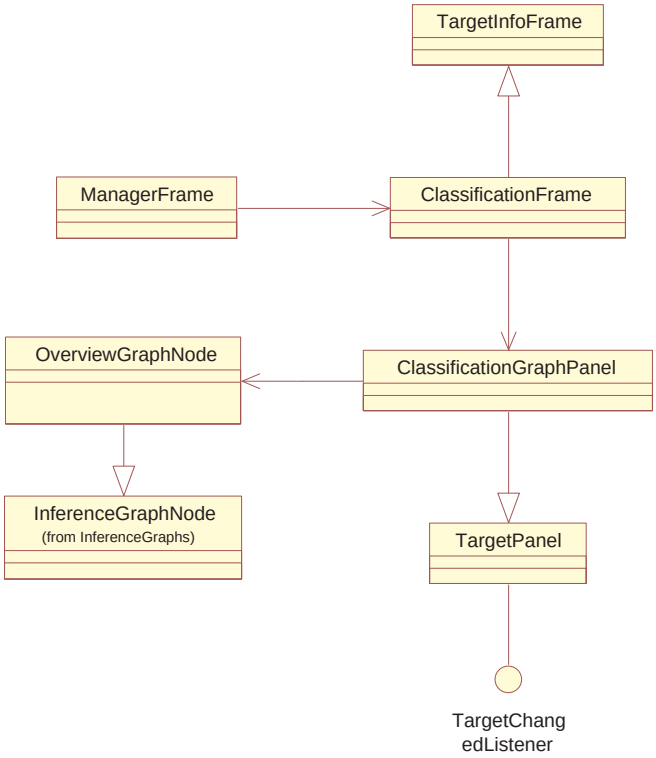

Figure 8.6: Class diagram of the gui

The class diagrams of the program are shown in Figure 8.6 and 8.7.

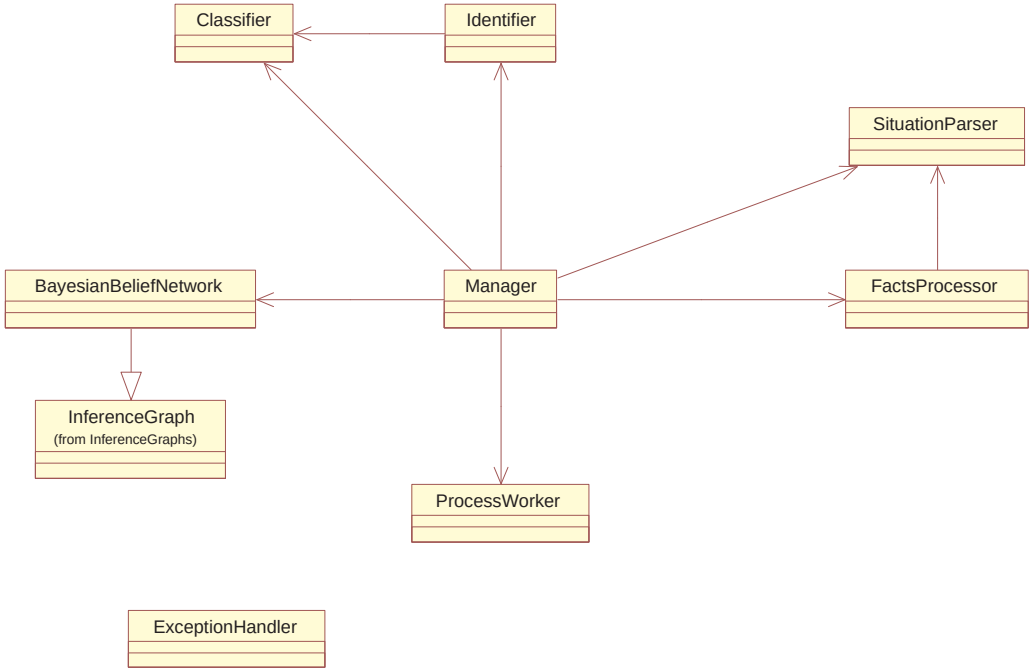

Figure 8.7: Class diagram of the main model

The CRC-cards

Not all classes are explained in a CRC-card, but all classes that are important to the understanding of the functionality of the program are shown on the next page.

Manager	
Responsibility	Collaborators
This class updates all the information about the target.	SituationParser FactsProcessor Classifier Identifier BayesianBeliefNetwork ProcessWorker

SituationParser	
Responsibility	Collaborators
This class parses all necessary values out of an XML file. Values that are not available are if possible replaced by initial values.	

FactsProcessor	
Responsibility	Collaborators
This class derives per target facts from the information gathered by the SituationParser.	BayesianBeliefNetwork SituationParser

ProcessWorker	
Responsibility	Collaborators
This class updates all nescessary information for the classification and indentification of the target.	Classifier Identifier ManagerFrame FactsProcessor

BayesianBeliefNetwork	
Responsibility	Collaborators
This class adds information to the Bayesian belief network and reads information out of the bayesian belief network.	InferenceGraphNode

Classifier	
Responsibility	Collaborators
This class combines all available influencing facts into a conclusion about the target's classification.	BayesianBeliefNetwork

Identifier	
Responsibility	Collaborators
This class combines all available influencing facts into a conclusion about the target's identification	BayesianBeliefNetwork Classifier

ManagerFrame	
Responsibility	Collaborators
This class manages all possible actions in the user interface.	Manager

InferenceGraphNode	
Responsibility	Collaborators
This class represents a node in the Bayesian belief network	

ExceptionHandler	
Responsibility	Collaborators
This class handles exceptions	

Figure 8.8: The CRC cards

8.3.3 The collaboration diagram

To get a clear understanding of how the program works we need to know how the classes communicate with each other. This can be shown in a collaboration diagram. The collaboration diagrams that were created for our model are shown in Figures 8.9, 8.10, 8.11, 8.12 and 8.13.

There are a couple of things in the diagrams that need explaining. First of all the arrow with only half of a head that is shown in Figure 8.11 and goes from the Manager to the ProcessWorker with the label *run*. This is not a printer error, this arrow indicates an asynchronous procedure call. This means that a new thread is started in which the ProcessWorker starts reasoning so that the control is given back to the Manager immediately. This enables the Manager to update the user interface while the ProcessWorker is reasoning. This will be seen in the sequence diagram as well.

Furthermore some Classes have methods within themselves, these methods are not shown in these collaboration diagrams. In the Classifier for instance the combination of all classification probabilities into a decision per target is done. The same occurs in the Identifier.

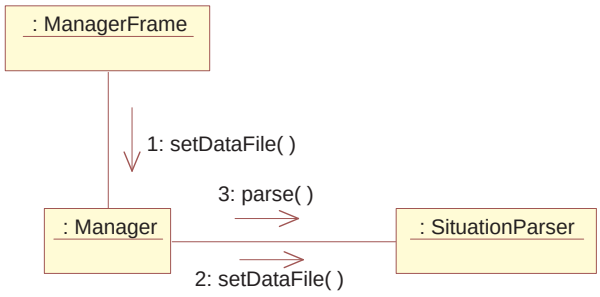

Figure 8.9: Collaboration diagram of the open data file action

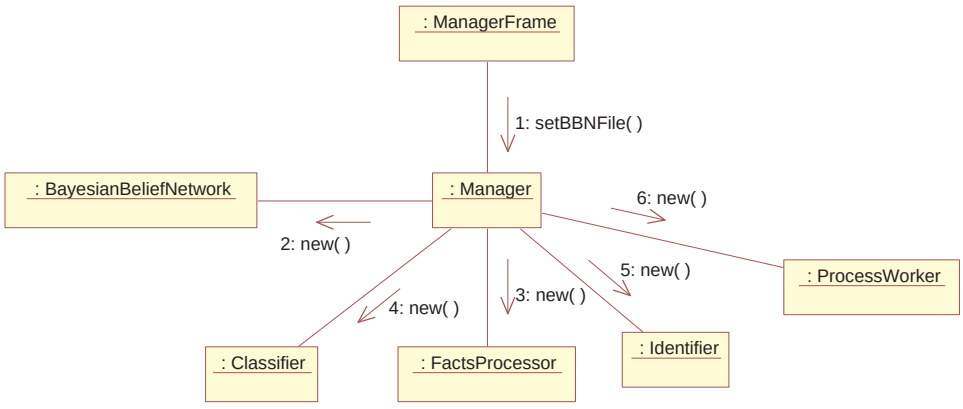

Figure 8.10: Collaboration diagram of the open BBN file action

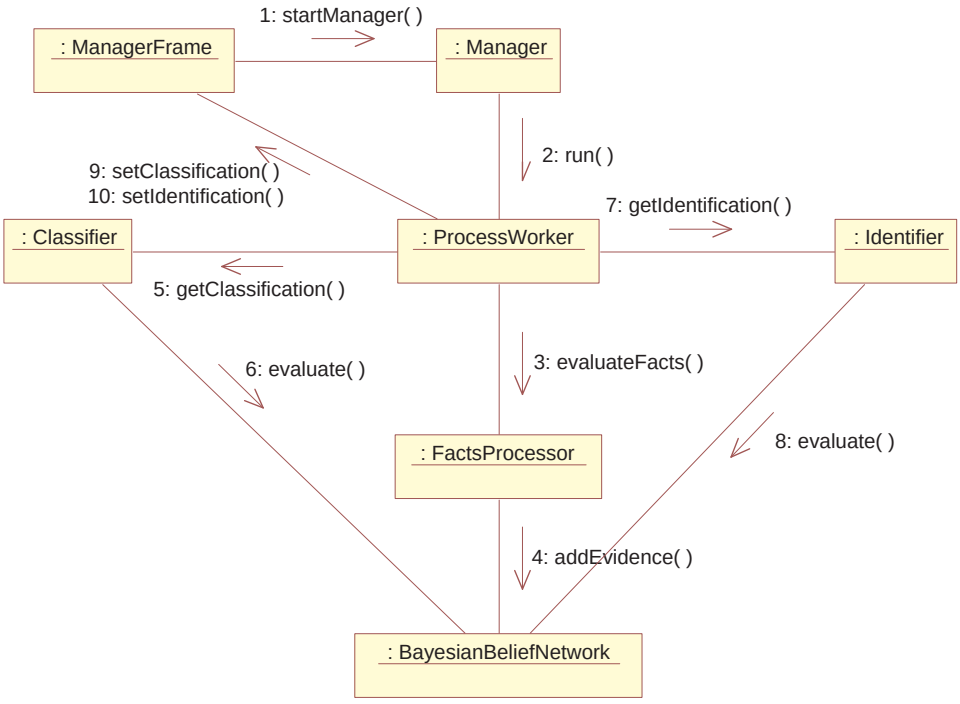

Figure 8.11: Collaboration diagram of the start reasoning process action

8.3.4 The sequence diagram

Because the Collaboration diagram is not designed to show the order of the messages between the classes in time, we have also created a sequence diagram. This diagram is better in showing time relations between the different method calls. The sequence diagrams are shown in Figures 8.14, 8.15 and 8.16.

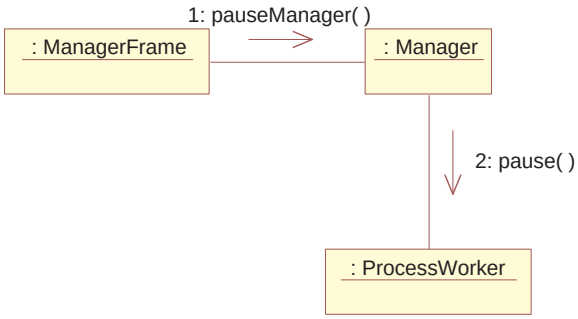

Figure 8.12: Collaboration diagram of the pause reasoning process action

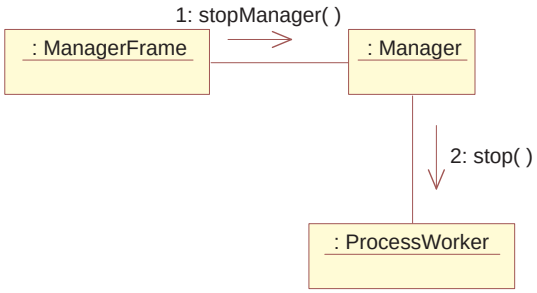

Figure 8.13: Collaboration diagram of the stop reasoning process action

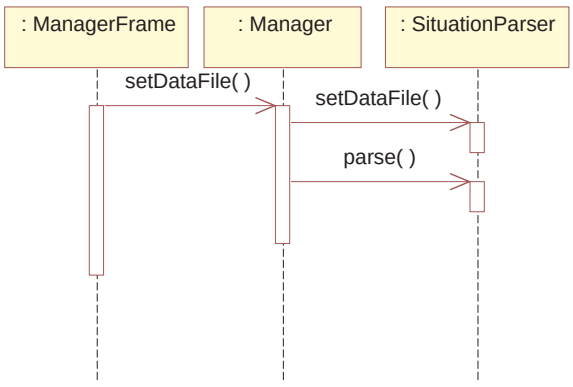

Figure 8.14: Sequence diagram of the open data file action

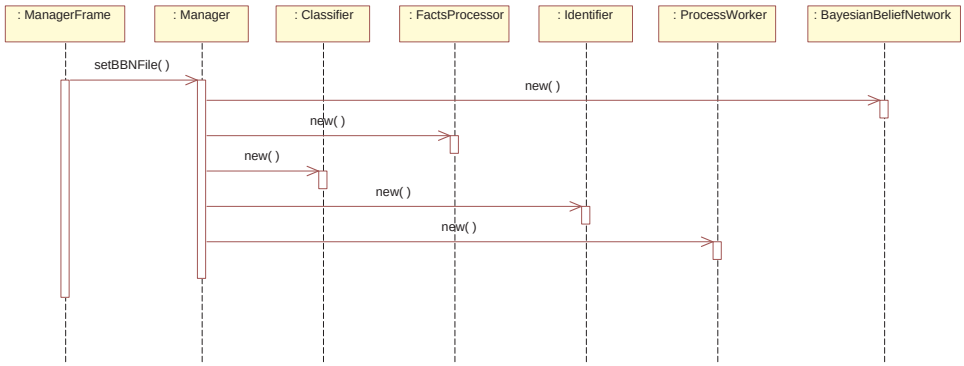

Figure 8.15: Sequence diagram of the open BBN file action

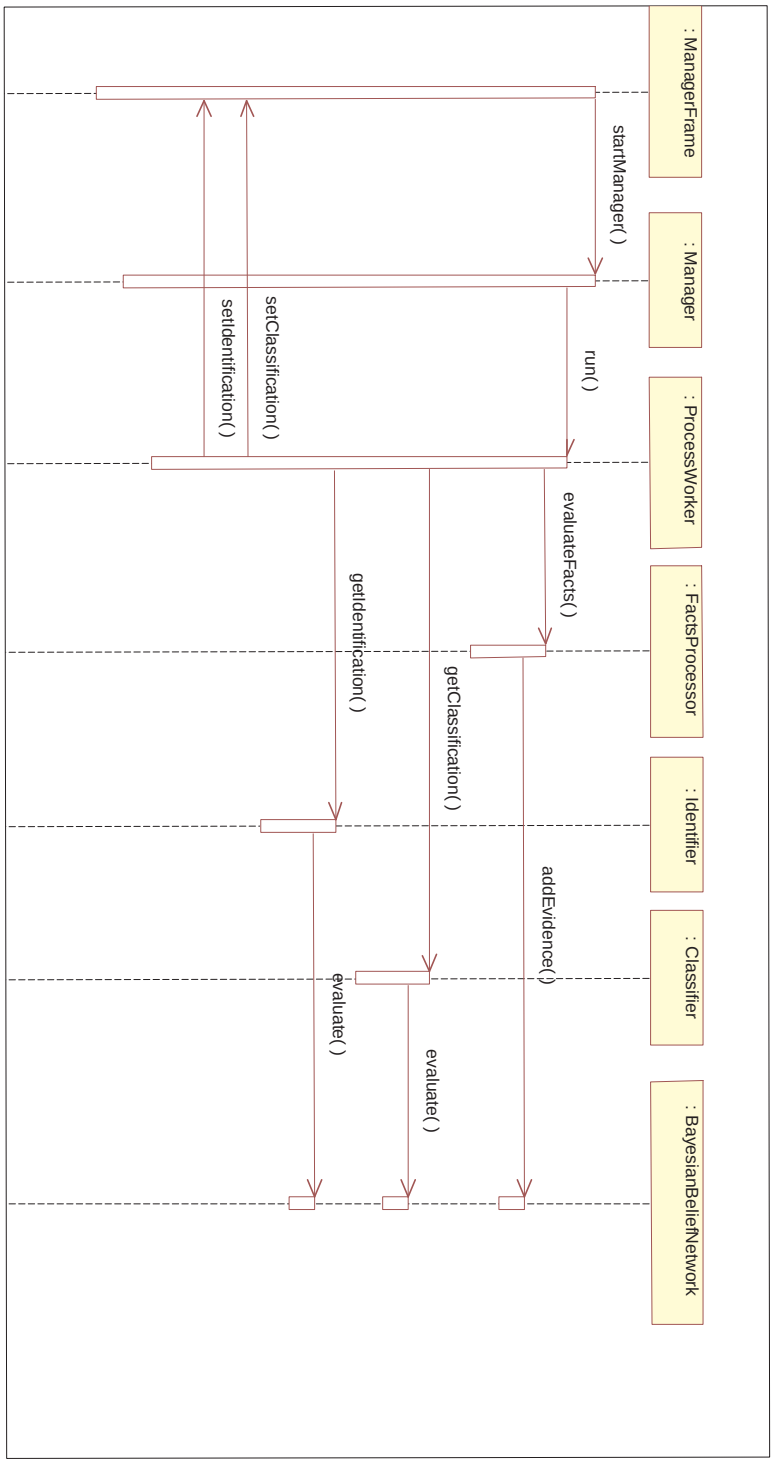

Figure 8.16: Sequence diagram of the start reasoning process action

Chapter 9

TIC (Target Identification and Classification)

9.1 Introduction

When we started implementing the prototype we had to choose the tools we would use to implement the Bayesian belief networks. We intended to implement the prototype in JAVA, so we searched for a tool which could help us to implement Bayesian belief networks in Java. First we thought about GeNIe 1.0 (Graphical Network Interface), which is a software package that can be used to create decision theoretic models intuitively using a graphical click-and-drop interface. GeNIe is the graphical interface to SMILE, a robust inference engine which has been thoroughly tested in the field since 1998. During a series of tests we experienced some problems creating large Bayesian belief networks. Therefore we searched for another tool and found a solution in JavaBayes, which is a set of tools for the creation and manipulation of Bayesian networks. The system is composed of a graphical editor, a core inference engine and a set of parsers. The graphical editor allows you to create and modify Bayesian networks in a friendly user interface. as an extra advantage JavaBayes is an open source program to which we could add some new abilities. During the implementation of our prototype a new version of GeNIe was released which looks promising, but we did not use it because we already made some implementations using JavaBayes.

9.2 Prototype

In Figure 9.1 the overall architecture of the system can be seen. The input is an XML file which contains all available information at different timepoints, these files look like Figure 9.2. In the TIC program we implemented the Bayesian belief networks given in Chapter 6 these are ordinary Bayesian belief models without a temporal aspect. We first wanted to examine the models in a simple way and if they work the temporal relations can be added afterward. The program makes a decision for each timepoint independently based on the sensor data available at that timepoint.

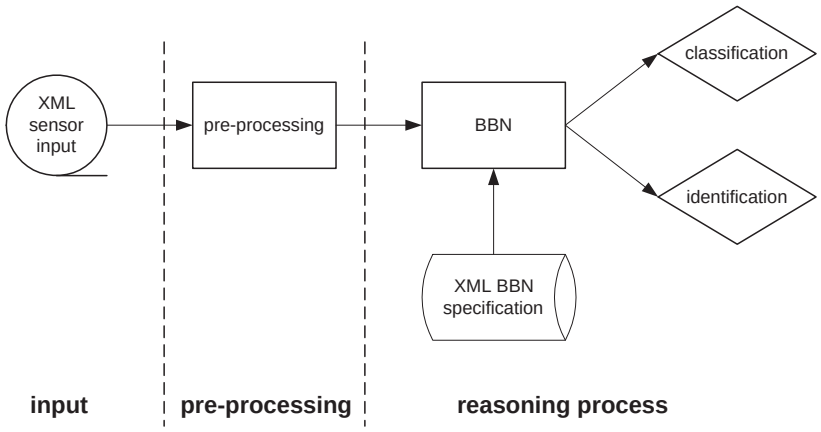

Figure 9.1: The architecture of the entire system

By looking at the decisions in time we might already see some temporal relations. In Figure 9.3 a part of the overall Bayesian network can be seen, this is a combination of all Bayesian networks shown in Chapter 6.

9.3 JavaBayes

The user interface of JavaBayes looks like Figure 9.3, we can see a part of the Bayesian belief network for this project. In this user interface it is possible to add new nodes and conditional dependencies. We can change all properties of a node in an interface like Figure 9.4. In this interface we added the possibility to combine the node's parents by using the *noisy and* or *noisy or* method. It is possible to add evidence to the network by making some nodes observed.

A network build in JavaBayes can be saved in several formats which make it possible to use the network in another environment. We have chosen to save the network in an XML format. In the TIC program we use parts of the JavaBayes code to import the XML file and add evidence to the network. The JavaBayes code can then be used to evaluate certain nodes in the network. This makes us able to add and extract information to the network in real time.

```
<?xml version="1.0"?>
<data>
   <environment>
      <airlane>
         <altitude>
            <minimum> 20000 </minimum>
            <maximum> 22000 </maximum>
         </altitude>
         <velocity> 400 </velocity>
         <heading> 240  </heading>
         <position>
             <!-- the start and end positions should be given in the same
               format as the position tag of the target -->
            <start>  </start>
            <end> </end>
         </position>
      </airlane>
      <airlane>
         ...
      </airlane>
   </environment>
   <!-- the time attribute can be given as a numeric of type long -->
   <situation time="345435435">
      <target number = "1">
         <track>
            <position>
            <x-pos> 104.45    </x-pos>
            <y-pos> 75.89  </y-pos>
             <!-- altitude is given in meters -->
            <altitude> 3000 </altitude>
            </position>
             <!-- velocity is given in meters/second -->
            <velocity> 300 </velocity>
            <heading> 350  </heading>
         </track>
         <iff>
            <mode>4</mode>
             <!-- The value of the tag correct is either TRUE or FALSE. -->
            <correct>TRUE</correct>
         </iff>
         <esm>
            <frequency> 1.0e+010    </frequency>
            <prf> 1.0e+004    </prf>
            <pulse_length> 1.0e-006 </pulse_length>
         </esm>
             <!-- The value of the tag link is either TRUE or FALSE. -->
         <link>FALSE</link>
      </target>
      <target>
         ...
      </target>
   </situation>
   <situation time="343535347">
      ...
   </situation>
</data>
```


Figure 9.2: An example of the XML file with sensor information about the environment

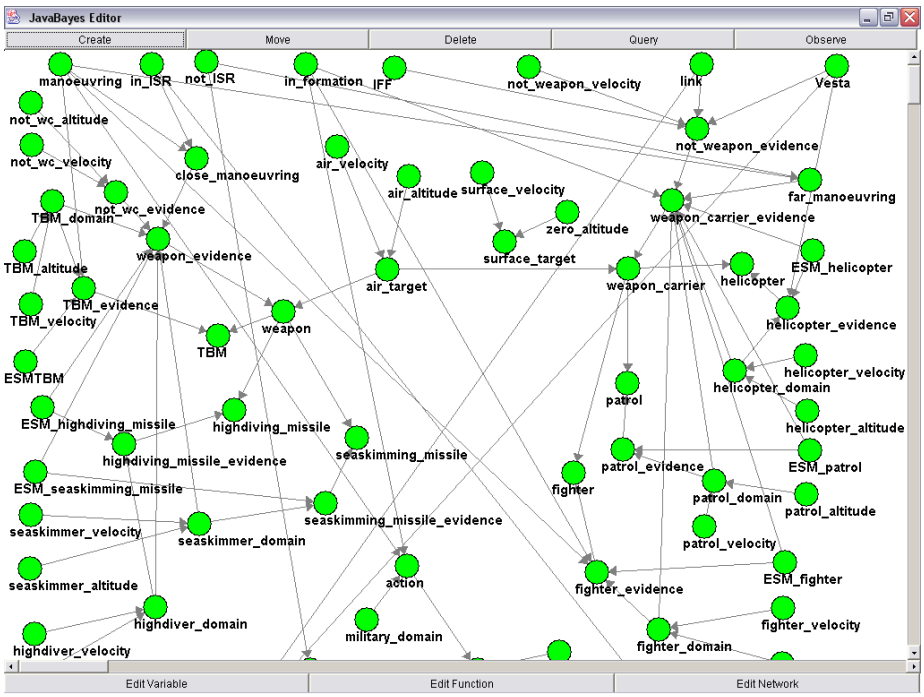

Figure 9.3: The user interface of JavaBayes

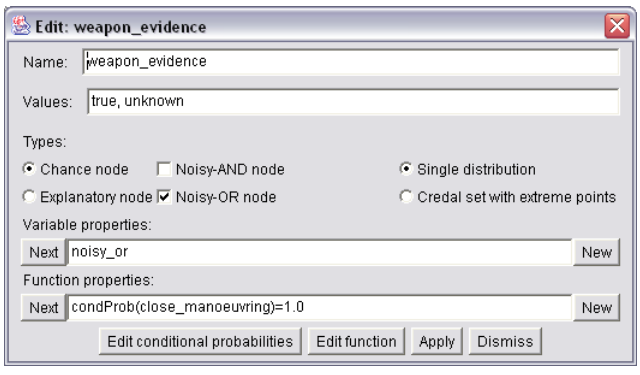

Figure 9.4: The user interface of JavaBayes

Part IV

Results, conclusions and recommendations

Chapter 10

Test scenario's

10.1 Introduction

In this chapter we will describe the results of the tests that were performed with the TIC. First we will show some pictures of the user interface of the TIC program and we will make some remarks about the way in which the tests were performed.

The system was tested by executing a scenario in which a ship may encounter all possible targets in a way that has been designed to test the program, to see if the program produces good results and satisfies the requirements that have been defined in section 3.2. The scenarios that were used are described in section 10.3, after which the test results are given and explained in section 10.4.

10.2 The user interface

The user interface enables us to view the decisions in an easy way, this can be seen in Figures 10.1, 10.2 and 10.3. In the user interface we are able to select the Bayesian belief network and the xml file. If these files are selected, we can start the reasoning process, pause it and stop it. We can get more detailed information about a decision as is shown in 10.3.

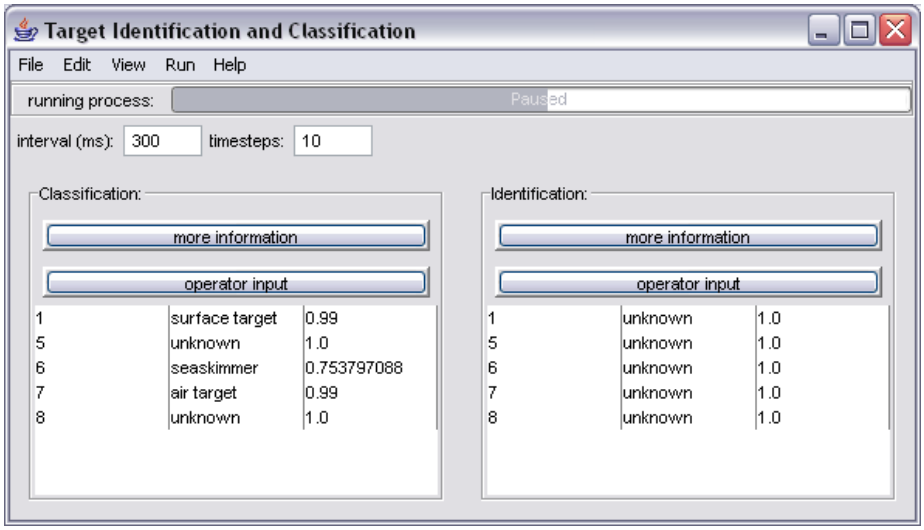

Figure 10.1: The user interface of the TIC program

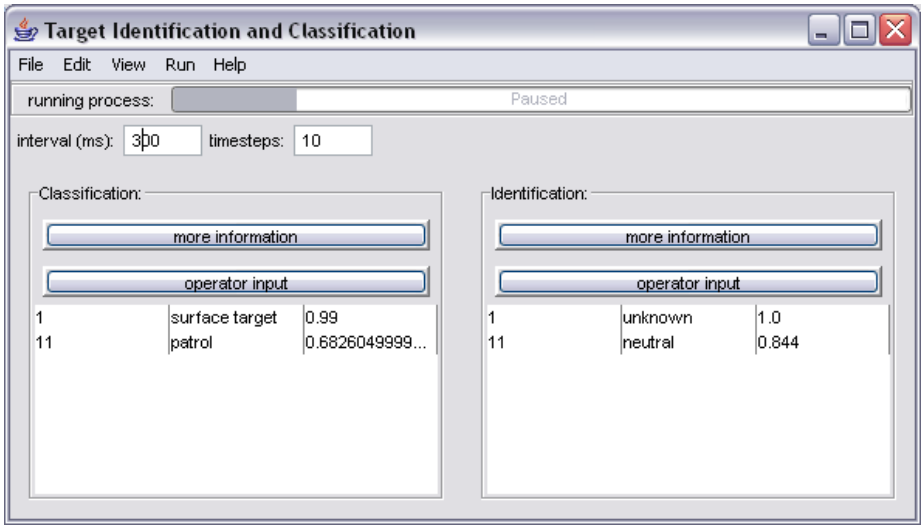

Figure 10.2: The user interface of the TIC program

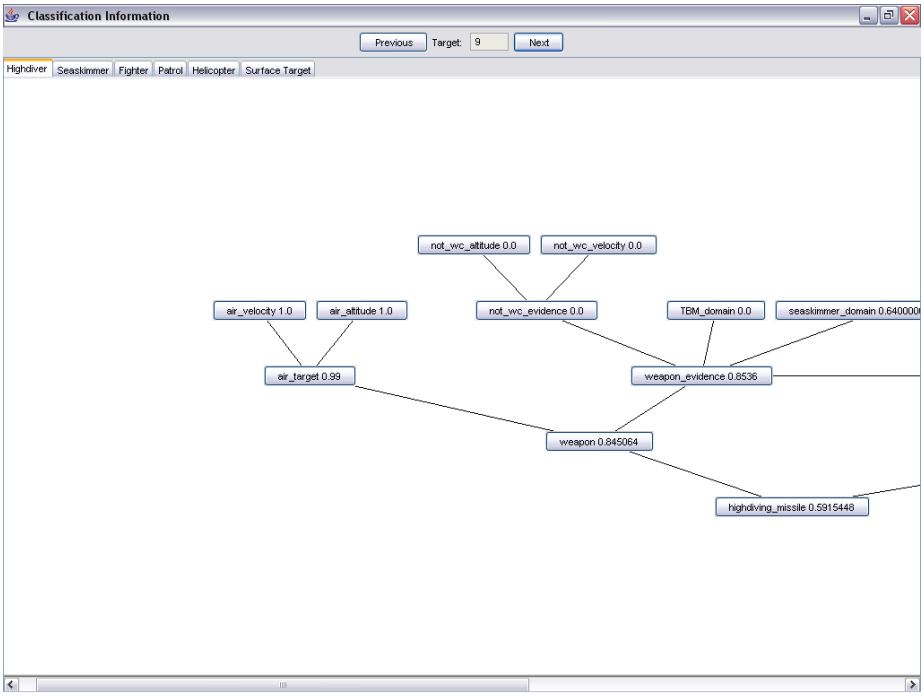

Figure 10.3: The user interface of the TIC program

10.3 The test scenario

The model that was used in this test scenario was developed in the STATOR project at the Royal Netherlands Naval College. This program gives an XML file as output in which all information from the ship's sensors about targets in the neighbourhood.

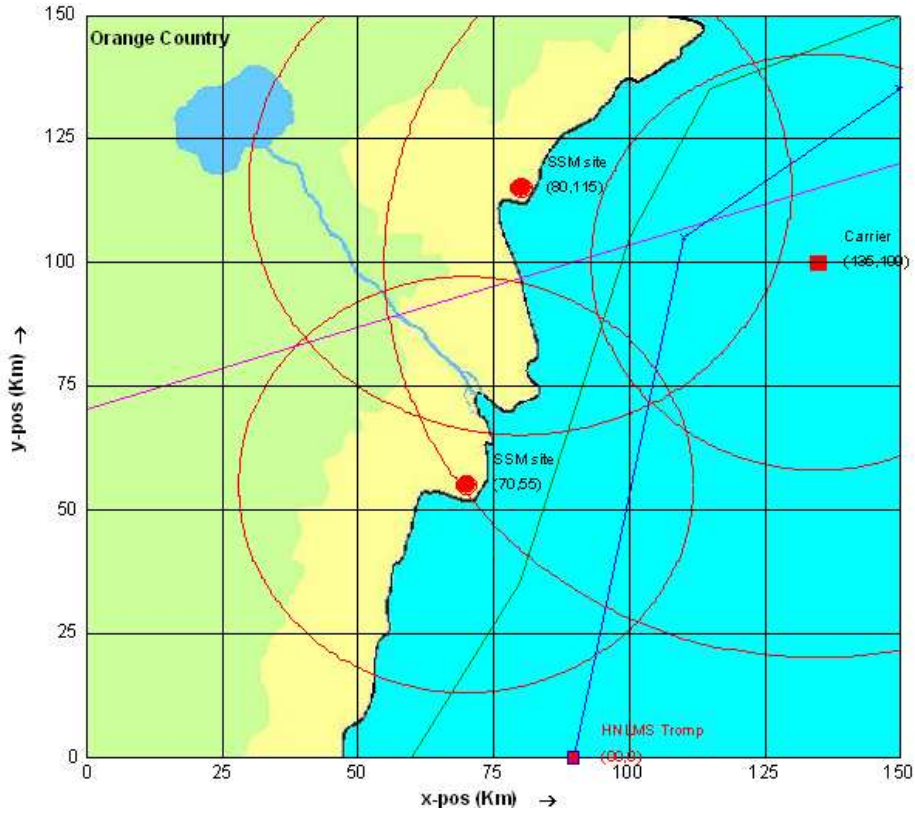

Figure 10.4: The test scenario

As can be seen in Figure 10.4 In this scenario a ship sails a certain track in which some targets may approach the ship. In our scenario the ship first reaches a missile site which fires four sea skimming missiles, second the ship reaches a missile site which fires two sea skimming missiles with way points and in the end an airliner flies across the ship in an airlane. This will be split up in three separate scenario's.

10.4 The test results

scenario 1

The ship reaches the first missile site and encounters four seaskimming missiles. In Figure 10.5 the probability distribution in time can be seen. Here the first of four missiles is approaching the ship. In the figure the evolution of evidence in time can be seen, first the sensors give information about the altitude and velocity of the target. For the range of altitude and velocity of this target there are two sorts of targets which are equally likely, namely a seaskimming missile and a fighter. The decision displayed will be *airtarget*, because the probability of *weapon* and *weapon carrier* are equally likely too. Some time later, the target switches its radar on. This new information makes it possible to decide that the target is probably a seaskimming missile.

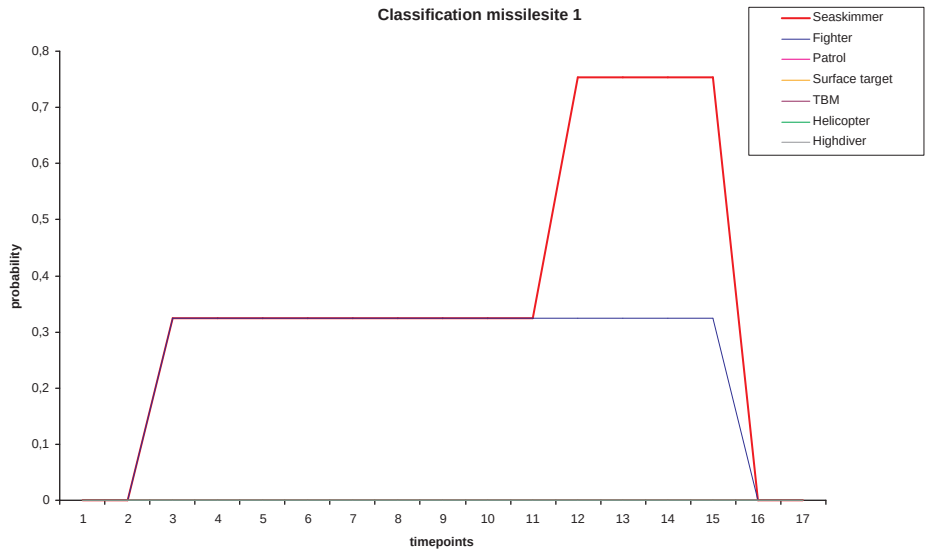

Figure 10.5: The probability distribution over the possible decisions for missile site 1

scenario 2

The ship reaches the second missile site and encounters two seaskimming missiles with waypoints in their track. In Figures 10.6 and 10.7 the probability distribution in time can be seen. The difference between these two figures is the database used to determine what platform may use the detected radar. In the first figure the radar is thought to be of a seaskimming missile, in the second figure the radar is thought to be of a highdiving missile. In the last case there is some conflicting evidence, the target is moving with a velocity and in the altitude range of a seaskimming missile but regarding the radar it could be a highdiving missile.

In these figures we see first two pop ups before we continuously detect the target, that is because of the sort of radar which is used. In this scenario we have a priori knowledge about the position of a missile site along the track. We expect a threat out of that direction and use a special radar to check for a longer rage with smaller bundle in that direction ones in a while. So we are able to detect the missiles before they enter our air surveillance radar range.

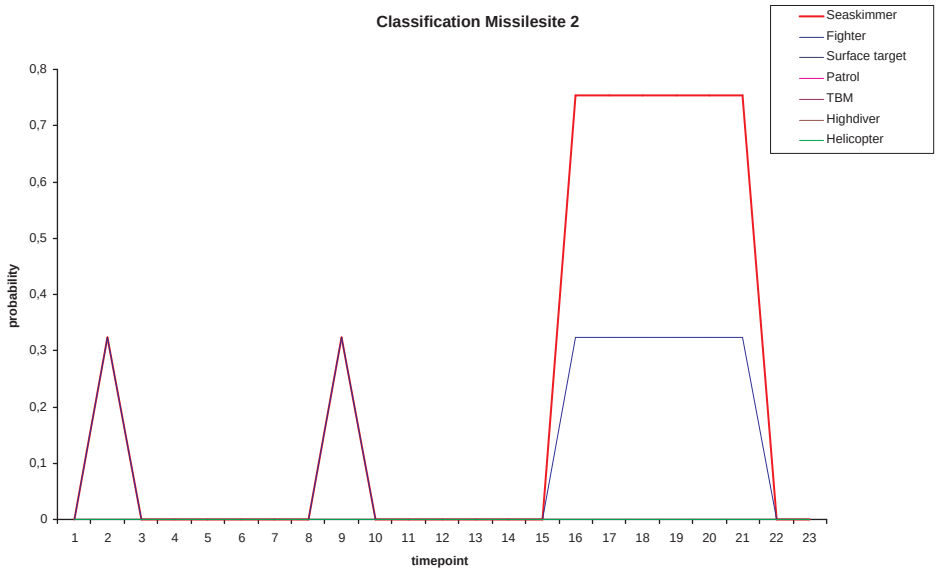

Figure 10.6: The probability distribution over the possible decisions for missile site 2

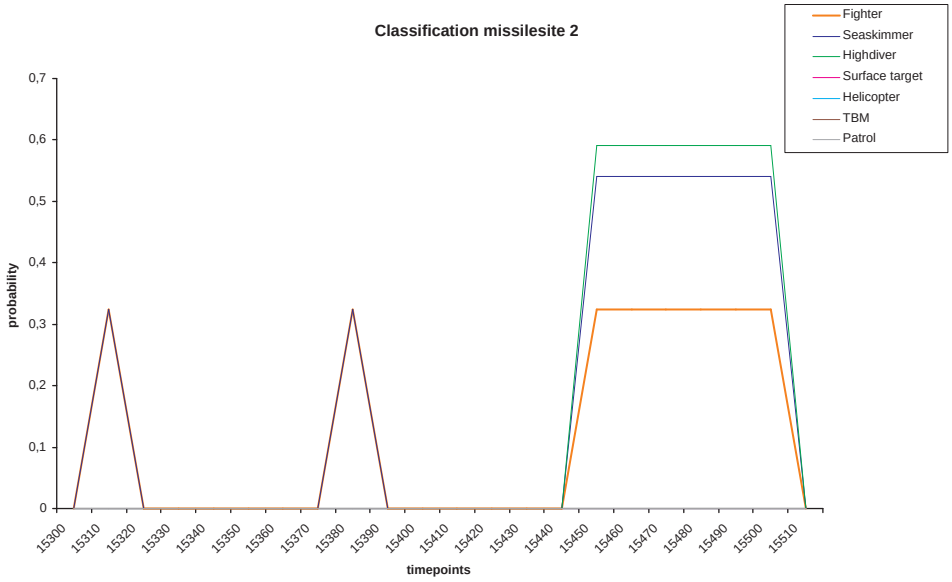

Figure 10.7: The probability distribution over the possible decisions for missile site 2 with conflicting evidence

scenario 3

The ship encounters an airliner which is flying in an airlane. The classification of this target can be seen in Figure 10.8. This aircraft transmits an IFF signal in mode 3, this makes us able to identify the target this can be seen in Figure 10.9. In the first two scenario's we see no identification figures, because we are not able to identify a target based on velocity and altitude only.

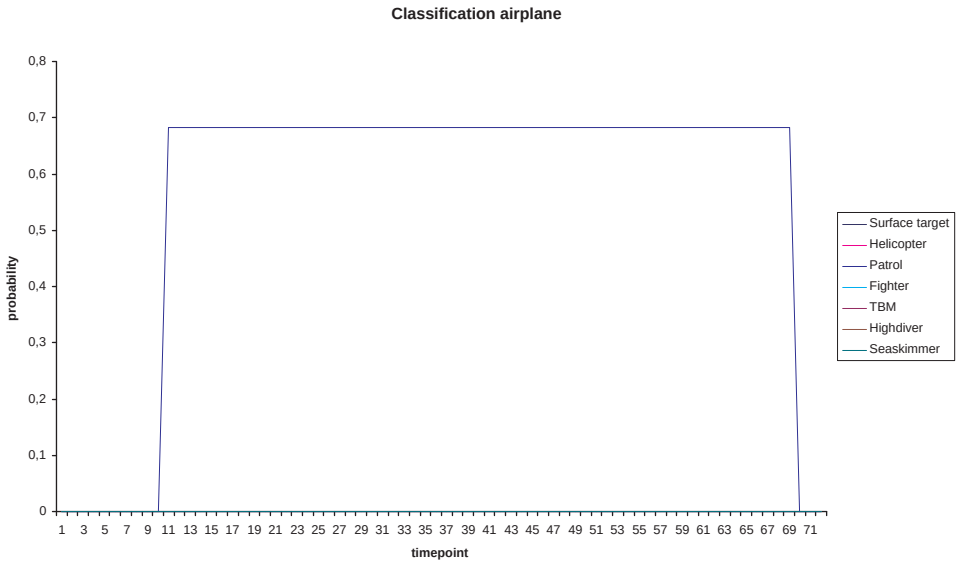

Figure 10.8: The probability distribution over the possible classification decisions for the airplane

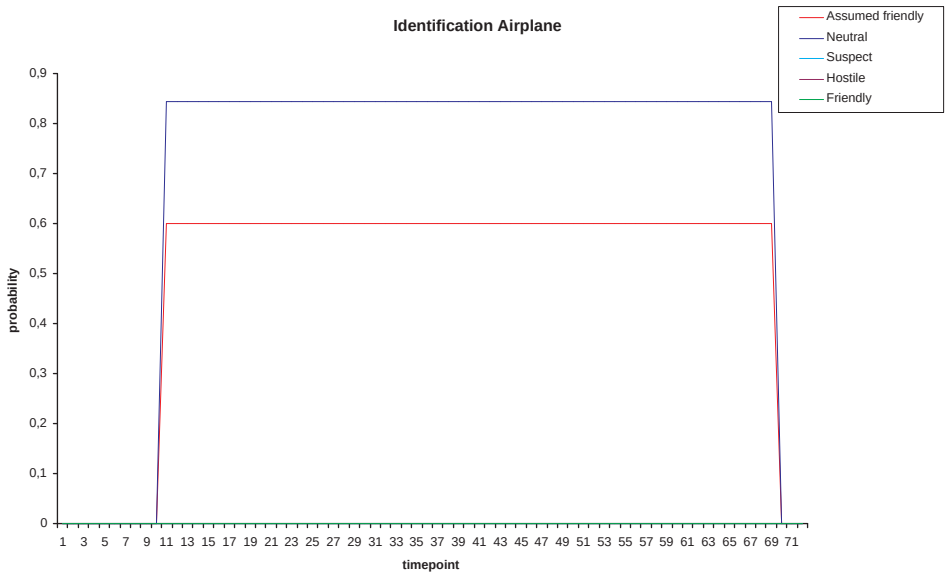

Figure 10.9: The probability distribution over the possible identification decisions for the airplane

10.5 Evaluation of the test results

For all three scenario's we want to know if the TIC program is able to classify and identify the targets right. During the execution of the test scenario's we realised that the TIC program was able to classify most of the targets correctly if there was enough information available. First we tested the program using the velocity, altitude and heading of each target. We saw that the program was not able to make a correct decision. Because the velocity and altitude domain of a fighter and a seaskimming missile are almost identical, the probabilities of both options become equal. The program decides with a maximum likelihood theorem, and then displays the decision air target, because that is the only thing that is certain. If the program gets some more specific information like a radar that switches on during the approach the program becomes able to draw the right conclusion. The same can be seen in the third scenario, the airliner has a slightly higher velocity than we would expect of a patrol aircraft so the probability stays quite low. The decision is made based on the ESM signature which is obviously a civil one.

For the identification it became clear that more information is necessary than for the classification to make a good decision. This could be directly deduced from the Bayesian belief networks in Section 6.2. Therefor we only see a proper identification in the third scenario. In that case we have an IFF transmission and we know the target is flying in an airlane and we have an ESM signature of the target which is obviously a civil one. This leads to a *neutral* identification, because the IFF mode 3 we see a lower probability for an *assumed friendly* identification.

In the second scenario a number of situations occur where temporal reasoning could improve the results. We already discussed the first two peaks but if we add the temporal relation that the belief in the target being an airtarget is amplified by observing this fact more than once, and keeping the belief if no new information is received. The figure would look smoother, we expect the figure to look like Figure 10.10 in stead of Figure 10.11.

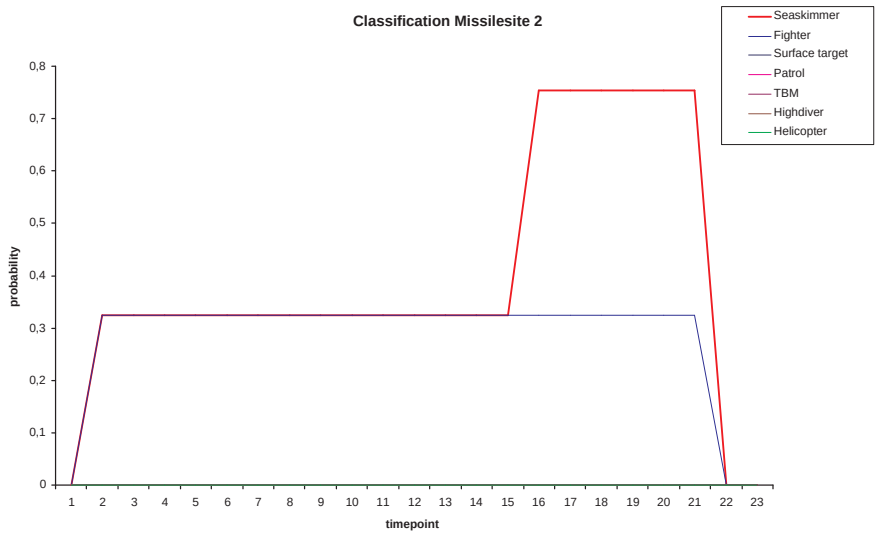

Figure 10.10: The probability distribution over the possible decisions for missile site 2 with temporal relations

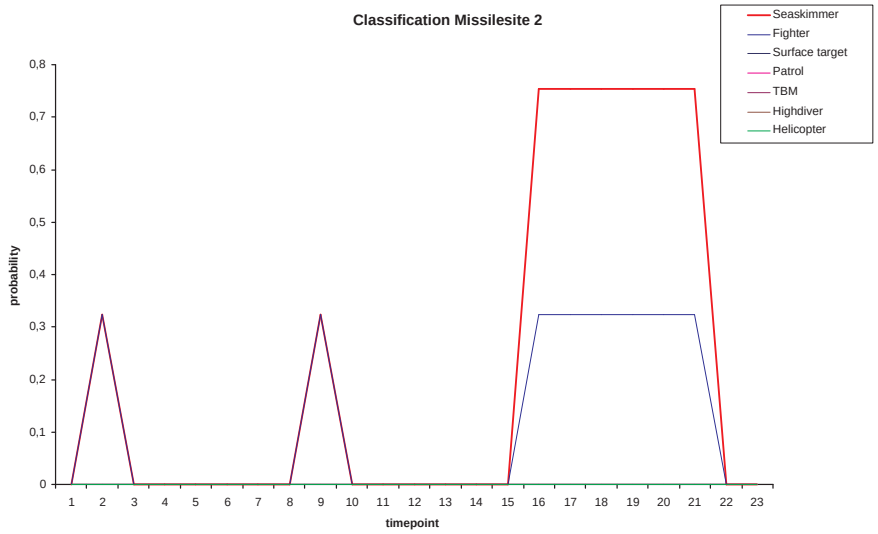

Figure 10.11: The probability distribution over the possible decisions for missile site 2 without temporal relations

Chapter 11

Conclusions and recommendations

This report shows that it is possible to design and implement a model that is able to make a decision about the classification of an air target and its identification based on sensor data in a maritime environment.

To design a true model for the situational awareness based on sensor information, I first investigated the current models in the combat simulation SEAROADS II build by TNO. This investigation showed that classification and identification were not accurately simulated. To improve this simulation, better knowledge about the way classification and identification is done on board combat vessels was necessary; this knowledge was gathered at the OPSCHOOL.

Further information about how to model uncertainty was very important. After a global study of the most common approaches Dempster-Shafer and Bayesian Belief Networks were selected as promising possibilities. A deeper study into these two theorems showed Bayesian Belief Networks best for this problem. Later a study was done to investigate the possibilities of temporal reasoning in the model. The most commonly used methods were investigated and Dynamic Bayesian networks showed to be useful for this model.

To design reliable Bayesian Belief Networks expert knowledge was necessary, which was again gathered at the OPSCHOOL. Based on this knowledge a model is designed which can be split into two complementing parts: the classification and the identification. In the classification the target type is specified, therefore two possible Bayesian Belief models were designed. In the identification the model determines whether the target is a friend or a foe.

The Bayesian belief models were implemented using JAVA. In this implementation the temporal aspect is not taken into account. To test the models some challenging scenario's were carried out. These tests show that for a proper classification of the target more information is needed than the velocity and the altitude of the target. For the identification of a target even more complex information is needed about the target's behaviour. In these tests I saw how temporal reasoning could smoothen the decisions.

I think it is important to build a prototype which takes the temporal aspects into account and to perform tests to determine the real benefits of temporal reasoning.

This model may be used in several applications:

- in a naval combat simulation;
- in a threat evaluation program;
- as a decision support system on board combat vessels.

For this last application some changes have to take place in the situation on board combat vessels. Most operators do not trust automatic systems. This model should be used as a support to make the right decision, not to make decisions on its own. But before the system is ready to be used in such a critical environment a lot of tests should be done to guarantee the reliability. In some cases the model might need some fine tuning.

Bibliography

[1] Bolderheij F. and Genderen, van P., *Mission Driven Sensor Management*, Royal Netherlands Naval College and Delft University of Technology, (2004).

[2] Bussé É. and Roy J., *Fusion of identity declarations from dissimilar sources using Dempster-Shafer theory*, Optical Engineering, Vol. 36 No.3, (March 1997).

[3] Burns B. and Morrison C.T., *Temporal Abstraction in Bayesian Networks*, AAAI Spring Symposium, Palo Alto, California, (2003)

[4] Cooper G., *Probabilistic inference using belief networks is NP-hard*, Artificial Intelligence 42, 393-405, (1990).

[5] Dall I.W., *Threat assessment without situation assessment*, Information Technology Division, Defence Science and Technology Organisation, Australia.

[6] Delft, van J.H. and Passenier P.O., *Functions and tasks in current CIC*, J.H. van Delft and H. Schuffel, editors, Human Factors research for future RNLN Combat Information Centers. TNO-TM, Soesterberg, The Netherlands, (1995).

[7] Dempster A.P., *Upper and Lower Probabilities Induced by multivalued Mapping*, Annals of Mathematical Statistics, Vol. 38. No. 2, (1967).

[8] Endres G. and Gething M., *Jane's Aircraft recognition guide*, HarperCollins Publishers, (2002).

[9] Fries T.P., *Consensus Development in Fuzzy Intelligent Agents for Decision-Making*, Department of Computer Science, Coastal Carolina University, Conway, (2001).

[10] Guo H. and Hsu W.H., *A Survey of Algorithms for Real-Time Bayesian Network Inference*, AAAI/KDD/UAI-2002, Joint Workshop on Real-Time Decision Support and Diagnostic Systems, Laboratory of Knowledge Discovery in Database Department of Computing and Information Sciences, Kansas State University, (July 2002).

[11] Giarratano J. and Riley G., *Expert Systems principles and programming*, PWS Publishing Company, (1993).

[12] Kalman R.E. , *A new approach to linear filtering and prediction problems*, Transactions of the ASME, Journal of Basic Engineering, vol. 82, pp. 34-45, (1960).

[13] Knight B., Ma J., Cowell D. and Petridis M., *Theory and Models for Temporal Reasoning, Applications of Artificial Intelligence in Engineering X*, Computational Mechanics Publications, pp. 39-46, (1995).

[14] Lambalgen M. van, *Fuzzy logic*, handout, University of Amsterdam, (June 16, 2000).

[15] Lecours M.and Bussé É., *Une règle de décision non-ad hoc pour L'identification Automatique des Cibles par Fusion des Données de Plusieurs Capteurs*, Laboratoire de Radiocommunications et de traitment du Signal, Rapport annuel d'activité 1997-1998, (1998).

[16] LucasP.J.F., *Certainty-factor-like structures in Bayesian belief networks*, Utrecht University, Information and Computing Sciences (May 22, 2000).

[17] Mouthaan Q.,*Towards an intelligent cockpit environment: a probabilistic approach to situation recognition in an F-16*, Delft University of Technology, Faculty of Information Technology and Systems (June 2003).

[18] Murphy K.P., *Dynamic Bayesian Networks: Representation, Inference and Learning*, University of California, Berkeley, (2002)

[19] Nodelman U., Shelton C.R. and Koller D., *Learning Continuous Time Bayesian Networks*, Stanford University, Proceedings of the Nineteenth International Conference on Uncertainty in Artificial Intelligence, pp. 451-458, (2003)

[20] Pan H., McMichael D., *Fuzzy Causal Probabilistic Networks - A new ideal and practical inference engine*, Cooperative Research Centre for Sensor Signal and Information Processing, Technology Park Adelaide, (May 22, 1998).

[21] Parsons S., Bigham J., *Possibility theory and the generalised Noisy OR model*, Department of Electronic Engeneering, Queen Mary and Westfield College.

[22] Vladimir Pavlović V., Rehg J.M., and Cham T., *A Dynamic Bayesian Network Approach to Tracking Using Learned Switching Dynamic Models*, Compaq Computer Corporation Cambridge Research Lab.

[23] Petterson G., *Multi-source integration and temporal situation assessment in air combat*, Defense Research Establishment Div. of Command and Contral Warfare Technology, Sweden, (1999).

[24] Pooley R. and Stevens P., *Using UML*, Addison-Wesley (1999).

[25] Russel S., Norvig P., *Artificial Intelligence A modern approach*, Prentice Hall, (1995).

[26] Rijn H.M. van, *Omgaan met onzekerheid bij identificatie van luchtdoelen*, Royal Netherlands Naval College, (1998).

[27] Shafer G., *A mathematical theory of evidence*, Princeton University Press, (1976).

[28] Smets Ph., *Imperfect Information: imprecision - uncertainty*, Uncertainty Management in Information Systems. From Needs to Solutions. A. Motro and PH Smets, Kluwer Academic Publishers (1997), 225-254, Université Libre de Bruxelles, (July 1999).

[29] Smets Ph., *What is Dempster-Shafer's model?*, Advantages in the Dempster-Shafer Theory of Evidence, Yager R.R., Fedrizzi M. and Kacprzyk J., Wiley (1994), 5-34, Université Libre de Bruxelles, (1994).

[30] Verhaegen M. and Verdult V., *Filtering and System Identification: An Introduction*, Delft University of Technoligy, Faculty of Information Technology and Systems, (October 2002).

[31] Vila L., *A Survey on Temporal Reasoning in Artificial Intelligence*, AI Communications, (1994)

[32] Voorbraak F., *Reasoning with uncertainty in AI*, Reasoning with uncertainty in Robotics, Intern. Workshop proceedings, pages 52-90Department of Mathematics, Computer Science, Physics and Astronomy, University of Amsterdam, (1995).

[33] Yu Y. and Johnson B.W., *Bayesian Belief Network and Its Applications*, University of Virginia, Center for Safety-Critical Systems Department of Electrical and Computer Engineering, (May 2002).

[34] *Allied Maritime Tactical Instructions and Procedures*, ATP-1(C), Vol. 1, (1983, Change 6).

[35] TNO FEL, *Anti-Air Warfare research using SEAROADS.*

[36] *Koninklijke Marine; Richtlijnen van Tactische Aard deel 3: Luchtverdediging*, RITA-3, (August 1995).

[37] *NATO Above Water Warfare Manual*, ATP-31(A), (1983, Change 10).

[38] *Report ANNCP WG IX CP 13*, (June 2002).

[39] *Response to the UML 2.0 OCL RfP*, (2002).

[40] TNO FEL, *User Documentation SEAROADS II.*

Internet

[41] Information, Decision and Control conferences *http://idc.cssip.edu.au*

[42] American Missiles, *http://www.fas.org/man/dod−101/sys/missile/index.html*

[43] Dempster-Shafer Theory of Evidence, *http://www.ccl.umist.ac.uk/teaching/material/5005/node30.html*

[44] AI Topics, *http://www.aaai.org/AITopics/*

[45] Bayesian networks, *http://www.ai.mit.edu/ murphyk/Bayes/bnintro.html*

Appendix A

Terminology

Classification: Classification is the process in which a target is analysed. The classification will be done in several layers, first we have to determine whether the target is a surface, subsurface or air target. In this case that part is quite easy, because the simulation only contains air, and surface targets. Than we try to analyse what kind of air target we are dealing with; e.g. an airbus, a fighter or a missile. If we know what type of target it is, it may be possible to be more specific, like what type of fighter is it? Eventually even the name of the fighter can be given in ideal circumstances.

Identification: Identification is the evaluation process of the target. We have to determine whether a target is hostile, friendly or neutral. As long as the identification of a target is not completely clear pending identities may be used, these are suspect, assumed friendly and unknown. After the target has been in track for more than two minutes a real identification has to be assigned.

Rules of Engagement: ROE are directives to military forces (including individuals) that define the authorisation for, or limits on, the use of force during military operations. Formulation of ROE is influenced by a variety of factors. ROE first must be lawful. International law defines the lawful limits for the use of force during military operations. National law may further limits the use of force by member states in certain types of military operations or in certain situations. Within the legal framework, the north Atlantic Council/ Defence Planning Committee (NAC/DPC) provides political direction for the conduct of military operations, including authorisations for, and limitations on, the threat or use ROE do never limit the inherent right of self-defence. ROE are not used to assign tasks or give tactical instructions. With the exception of self defence, during peacetime and operations prior to a declaration of counter aggression, ROE provide the sole authority to NATO or NATO-led forces to use force. Following a declaration of counter aggression, ROE generally limit the otherwise lawful use of force [34].

Hostile intent: Rule 421[1] permits attack against designated forces that demonstrate hostile intent against NATO or NATO-led forces. The existence of hostile intent must be judged on the basis of both:

a) The threatening unit's capability and preparedness to inflict damage; and

b) Evidence, including intelligence, which indicates an intention to attack.

The weight of evidence and intelligence indicating intention to attack must demonstrate a clear and substantial threat. The military capability and preparedness to inflict damage can be taken to exist when certain tactical events occur. These may include manoeuvring into weapon launch positions, whether NATO or NATO-led forces are presently in range or not, the deployment of remote targeting methods, are the use of shadowers or tattletales to provide picture compilation. Additionally, evidence may come from non-tactical events such as increased indications of enemy mobilisation and/or warlike gestures, revealed in public or gained from intelligence sources; increased movements of ammunition from stockpiles to airfields, dockyards or army depots; and the requisitioning of land, air and sea transportation [34].

Hostile act: While self-defence permits the use of force to defend NATO or NATO-led forces against an imminent or actual attack, rule 422[2] permits attack against designated forces that commit are directly contribute to a hostile act against NATO or NATO-led forces. The actions outlined below are not all-inclusive; depending on the circumstances; other actions may, due to their purpose, be considered hostile acts. Similarly, the examples outlined below do not, of themselves, constitute hostile acts. The status of the crisis, the political situation at the time and if indeed a hostile act has occurred. Specific examples of hostile acts include, but do not limited to:

a) One or more maritime, air or land units conduct mine laying operations which impose limitations or restrictions upon the movement of NATO or NATO-led forces;

b) Military aircraft penetrating a NATO secure area and refusing to comply with interception instructions;

c) Intentionally impeding NATO or NATO-led military operations; and

d) Breaching, or attempting to breach, the security of a NATO or NATO-led military installation or restricted area [34].

IFF: Identification Friend or Foe is a system which interrogates a transponder in the target which may or may not answer with the right code. Based on this answer and the mode the transponder is in information about the target's identification can be gained.

Vesta: Vesta is a part of a helicopter guidance system which enlarges the radar contact of the helicopter on screen to make sure it is not lost in seaclutter. This system is only used by a few Western European countries.

[1] see Appendix B
[2] see Appendix B

Link 11: Link 11 is a data communication system which is used to synchronise the target information of several allied vessels on screen.

ESM: Electronic Support Measures is a set of passive sensors which give information about the target's radar configurations.

Airlane: An airlane is a civil track in which all civil air traffic should fly. An airlane is bounded by altitude, position, heading and speed.

ACO: Air Co-ordination Order is a flight plan of a civil airplane.

Military domain: The military domain is all air space not known as an airlane.

ISR: The Identification Safety Range is an area around the ship in which the identification of each target entering this range has to be known.

Perform identification: A pilot can be asked to perform identification, this is a sequence of actions the pilot has to perform in the right order. This sequence may be changed and only friendly forces are informed about it.

Appendix B

ROE [34]

B.1 Self-defence

Self-defence: It is universally recognised that individuals and units have an inherent right to defend themselves against attack or an imminent attack. In exercising this right, individuals and units will act in accordance with national law. ROE do not limit this right. Self defence is the use of such necessary and proportional force, including deadly force, by NATO or NATO-led forces to defend themselves against attack or an imminent attack. The following definitions apply:

a) Necessary means the use of force is indispensable for securing self-defence.

b) Proportional means a response commensurate with the perception of the level of the threat posed. Any force used must be limited to the degree, intensity, and duration necessary for self-defence.

c) Imminent means that the need to defend is manifest, instant and overwhelming.

d) Attack is the use of force against NATO or NATO-led forces and personnel.

Extended self-defence: In keeping with the principles of the alliance, within the general concept of self-defence, NATO or NATO-led forces also have the right to use that force which is necessary and proportional to defend other NATO or NATO-led forces and personnel in the vicinity from attack or imminent attack. In circumstances during peacetime and operations, prior to a declaration of counter aggression, and where the use of force is not justified by self-defence, force may only be exercised within the constraints of and permissions authorised by ROE. Because national laws differ, there will not always be consistency between the nations as to where the right to use force in self-defence and extended self defence ends and the use of force authorised by ROE begins. In cases of inconsistency, ROE within a given operation shall not be interpreted as limiting the inherent right of self-defence.

B.2 Identification of suspected targets

230 Positive identification is to be established visually.

231 Identification is to be established visually or by at least two of the following means:

- IFF
- Electro-optic
- Electronic warfare support measure
- Track behaviour
- Flight plan correlation
- Thermal imaging
- Acoustic intelligence
- Other secure active/passive systems.

232 Identification is to be established visually or by one or more of the following means:

- IFF
- Electro-optic
- Electronic warfare support measure
- Track behaviour
- Flight plan correlation
- Thermal imaging
- Acoustic intelligence.

B.3 Other secure active/passive systems

420 Attack on designated force(s) or targets is prohibited.

421 Attack against designated force(s) or designated target(s) demonstrating hostile intent (not constituting an imminent attack) is authorised.

422 Attack against designated force(s) or designated target(s) which commits or directly contributes to a hostile act (not constituting an actual attack) is authorised.

423 Attack against designated force(s) or designated target(s) which have previously attacked or directly contributed to an attack, is authorised.

424 designated commanders are authorised to judge whether an attack is the first of a series and, in this event, to attack all units constituting a continuing threat.

425 Attack on designated military installation(s), facility(ies), equipment, and unit(s) which are engaged in or support military activity that threatens designated force(s), person(s) or property is authorised.

426 Attack on designated force(s) or designated target(s) which substantially contribute to the conduct of hostile military operations against designated force(s), or persons or property with designated special status is authorised.

427 Attack on designated force(s) in designated circumstances is authorised.

Appendix C

Standard IDCRITS [36]

This section is classified, but can be obtained separately.

Appendix D

Dempster-Shafer's basic terminology

Dempster-Shafer has its own extensive terminology, partly because Dempster-Shafer theory contains many new notions, partly because well-known notions are given a new name. An overview of the most common terms is given:

Frame of discernment:

a sample space Θ is called a frame of dicernment or shortly a frame, we assume the frames to be finite.

Basic Probability Assignment:

A mass function or basic probability assignment (BPA), over a frame Ω is a function $m : 2^{\Omega} \rightarrow [0, 1]$ satisfying the following conditions:

$$m(\oslash) = 0 \tag{D.1}$$

$$\sum_{A \subseteq \Omega} m(A) = 1 \tag{D.2}$$

$$m(A) \geq 0 \tag{D.3}$$

The quantity $m(A)$ is a measure for the belief that is assigned to exactly the set A. $m : 2^{\Omega} \rightarrow [0, 1]$ means that each subset of the frame is associated with a number between 0 and 1.

Focal element:

The subset A which is associated with a BPA.

Belief function:

Let m be a mass function over a frame Ω. The belief function Bel induced by m is defined as follows.

For every $A \subseteq \Omega$,

$$Bel\,(A) = \sum_{B \subseteq A} m\,(B) \tag{D.4}$$

Plausibility function:

Let m be a mass function over frame Ω. The plausibility function Pl induced by m is defined as follows:

For every $A \subseteq \Omega$,

$$Pl\,(A) = \sum_{A \cap B \neq \oslash} m\,(B) \tag{D.5}$$

$Pl(A)$ is a measure of the belief which is not (yet) assigned to propositions which indicste the falsity of A.