Thesis NLDA
Simulating Anti-Submarine Warfare using MANA

Date March 2014
Authors Sub-Lieutenant W.A. Knippenberg
Classification Ongerubriceerd

Principal Netherlands Defence Academy
Faculty of Military Sciences
Department of Military Technical Sciences
Operations Research

Supervision Ir. H.W. Noordkamp
Dr. H. Monsuur
Dr. Ir. R.H.P Janssen
Ir. R.R. Hordijk

Copy no.
No. of copies -
No. of pages 56 (excluding appendices)
No. of appendices 5

The classification designation Ongerubriceerd is equivalent to Unclassified, Stg. Confidentieel is equivalent to Confidential and Stg. Geheim is equivalent to Secret.

All rights reserved. No part of this report may be reproduced in any form by print, photoprint, microfilm or any other means without the previous written permission from TNO.

All information which is classified according to Dutch regulations shall be treated by the recipient in the same way as classified information of corresponding value in his own country. No part of this information will be disclosed to any third party.

In case this report was drafted on instructions from the Ministry of Defence the rights and obligations of the principal and TNO are subject to the standard conditions for research and development instructions, established by the Ministry of Defence and TNO, if these conditions are declared applicable, or the relevant agreement concluded between the contracting parties.

© 2014 TNO
Abstract

The agent-based simulation program MANA is a military oriented tool for exploring scenarios. This research project focuses on the usability of MANA for testing warship concepts in an early phase of the design process. The design should be tested on its performance in the different aspects of naval warfare. For this research project, an Anti-Submarine Warfare scenario is modelled in MANA, which will be used in Monte-Carlo simulations. The implementation of the simulation models is (partly) validated with an analytical model. To examine its proper functioning and its output, the model in MANA is compared with a model in the more complex tool UWT. Additionally, the insights MANA provides in concepts is analysed. This analysis shows that MANA is useful for modelling simple to medium complex scenarios, and for testing concepts in an early stage. It is inadequate, however, for modelling more complex scenarios or behaviour.
Preface

"Ninety percent of operations research is beer"

Scientist at Coastal Command section OR, during WW-II [1].

The report before you is the finalisation of 5 years of education at the Royal Netherlands Naval Academy. Part of the course to become a naval officer was a bachelor's degree of 3 years at the Netherlands Defence Academy (NLDA). This report is the thesis of my bachelor, Military Systems and Technology. For this bachelor I specialised in Operations Research. Different techniques from this field of expertise are used for this thesis.

The research has been done at the Netherlands Organisation for Applied Scientific Research (TNO) in The Hague. It was executed in cooperation with the department of Military Operations at TNO. I would like to express my gratitude towards this group and towards my supervisor in particular, Ir. Wouter Noordkamp. His knowledge and guidance helped to complete this research project. Additionally the knowledge of Dr. Ir. B.J. van Oers, from the Netherlands Defence Material Organisation (DMO), about MANA and his insights in its use helped improving the models for this thesis. Furthermore I would like to thank my supervisors at the academy, Dr. H. Monsuur, Dr. Ir. R.H.P. Janssen and Ir. R.R. Hordijk. Their sharp feedback, comments and questions helped shaping and improving this research. Finally, I would like to thank KLTZ R.M. Platel and LTZ1 M.J.M. Hezemans for their knowledge and insight in (anti-)submarine operations.

This thesis is written for those who want to use simulation tools for operational or military conceptual tests. It provides them with knowledge about MANA, how it can be used for modelling naval scenarios and what strength and drawbacks are in the process of using MANA. Besides it helps to gain insights to where opportunities lies when a new military simulation tool would be designed.

A great part of Operations Research consists of discussing the research, the models, the input parameters and the output analysis with colleagues, experts and friends. These discussions provide insights in real operations and help focus the research. This is often done at informal settings with a cup of coffee, or at night with a glass of beer. Therefore the comment from a scientist at the OR section of Coastal Command in England during the Second World War still holds ground. Ninety percent of Operations Research is beer (and coffee, and whisky).

Luitenant-ter-zee der derde klasse Willem Knippenberg,
## Contents

1. **Introduction** ....................................................................................... 11  
   1.1 Background ......................................................................................... 11  
   1.2 Study objective .................................................................................... 11  
   1.3 Structure of the thesis ........................................................................... 12  

2. **Operational and Theoretical background** ............................................ 13  
   2.1 Anti-Submarine Warfare ........................................................................ 13  
   2.2 Simulation studies ................................................................................ 14  
   2.3 Agent-based models ............................................................................ 15  
   2.4 MANA ................................................................................................. 16  
   2.5 Stochastic properties of simulations ....................................................... 18  
   2.6 Validation and verification of simulations ................................................. 18  
   2.7 Reliability of simulations ........................................................................ 19  

3. **Scenario set-up** .................................................................................. 21  
   3.1 Generic scenario .................................................................................. 21  
   3.2 Base scenario and tactics ..................................................................... 21  
   3.3 Measurement of Effectiveness ............................................................... 25  

4. **Modelling the scenarios** ..................................................................... 27  
   4.1 Analytical approximation ....................................................................... 27  
   4.2 The simulation framework ..................................................................... 29  
   4.3 The model layer ................................................................................... 29  
   4.4 The simulation layer ............................................................................. 31  

5. **Analysis of the models** ....................................................................... 35  
   5.1 Verification with analytical model ............................................................ 35  
   5.2 Quantitative comparison between MANA and UWT .................................. 35  
   5.3 Analysis of the scenarios through MANA ................................................ 38  
   5.4 Sensitivity analysis through MANA ........................................................ 44  
   5.5 Qualitative comparison between MANA and UWT .................................... 46  

6. **Limitations in MANA** .......................................................................... 49  
   6.1 Helicopter flight time limitations .............................................................. 49  
   6.2 False contact simulation ........................................................................ 50  
   6.3 Coordinate deployment of resources ....................................................... 50  

7. **Conclusion and recommendations** ..................................................... 53  

8. **References** ......................................................................................... 55  

### Appendices

A. Modelling the sensors  
B. Derivation of the equations  
C. Extended output of the base scenarios in MANA  
D. The MANA base models  
E. The Matlab programmes
### Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAW</td>
<td>Anti-Air Warfare</td>
</tr>
<tr>
<td>ABM</td>
<td>Agent Based Modelling</td>
</tr>
<tr>
<td>ASuW</td>
<td>Anti-Surface Warfare</td>
</tr>
<tr>
<td>ASW</td>
<td>Anti-Submarine Warfare</td>
</tr>
<tr>
<td>ASWC</td>
<td>Anti-Submarine Warfare Commander</td>
</tr>
<tr>
<td>C2</td>
<td>Command &amp; Control</td>
</tr>
<tr>
<td>CertSub</td>
<td>Certain Submarine contact</td>
</tr>
<tr>
<td>CI</td>
<td>Confidence Interval</td>
</tr>
<tr>
<td>CPA</td>
<td>Closest Point of Approach</td>
</tr>
<tr>
<td>DMO</td>
<td>Netherlands Defence Materiel Organisation</td>
</tr>
<tr>
<td>CW</td>
<td>Continuous Wave (sonar pulse)</td>
</tr>
<tr>
<td>DTA</td>
<td>Defence Technology Agency (New Zealand)</td>
</tr>
<tr>
<td>FF</td>
<td>Frigate</td>
</tr>
<tr>
<td>FM</td>
<td>Frequency Modulated (sonar pulse)</td>
</tr>
<tr>
<td>HELRAS</td>
<td>Helicopter Long-Range Active Sonar</td>
</tr>
<tr>
<td>HVU</td>
<td>High Value Unit</td>
</tr>
<tr>
<td>JSS</td>
<td>Joint Support Ship</td>
</tr>
<tr>
<td>LFAS</td>
<td>Low Frequency Active Sonar</td>
</tr>
<tr>
<td>LPD</td>
<td>Landing Platform Dock</td>
</tr>
<tr>
<td>MANA</td>
<td>Map Aware Non-Uniform Automata</td>
</tr>
<tr>
<td>METFR</td>
<td>Maximum Effective Torpedo Firing Range</td>
</tr>
<tr>
<td>MFF</td>
<td>Multi-purpose frigate</td>
</tr>
<tr>
<td>MoE</td>
<td>Measurement of Effectiveness</td>
</tr>
<tr>
<td>NLDA</td>
<td>Netherlands Defence Academy</td>
</tr>
<tr>
<td>NonSub</td>
<td>NonSubmarine contact</td>
</tr>
<tr>
<td>PD</td>
<td>Periscope Depth</td>
</tr>
<tr>
<td>PosSub</td>
<td>Possible Submarine contact</td>
</tr>
<tr>
<td>ProbSub</td>
<td>Probable Submarine contact</td>
</tr>
<tr>
<td>RNLN</td>
<td>Royal Netherlands Navy</td>
</tr>
<tr>
<td>RoE</td>
<td>Rules of Engagement</td>
</tr>
<tr>
<td>SA</td>
<td>Situational Awareness</td>
</tr>
<tr>
<td>SS</td>
<td>Submarine</td>
</tr>
<tr>
<td>SSC</td>
<td>Coastal Submarine</td>
</tr>
<tr>
<td>SSK</td>
<td>Conventional diesel-electric Submarine</td>
</tr>
<tr>
<td>SSN</td>
<td>Nuclear Submarine</td>
</tr>
<tr>
<td>TAS</td>
<td>Towed Array Sonar</td>
</tr>
<tr>
<td>TDZ</td>
<td>Torpedo Danger Zone</td>
</tr>
<tr>
<td>TNO</td>
<td>Netherlands Organisation for Applied Scientific Research</td>
</tr>
<tr>
<td>UWT</td>
<td>Underwater Warfare Testbed</td>
</tr>
</tbody>
</table>
1 Introduction

This chapter explains the relevance and the objectives of this study and outlines the structure of the thesis. The background is discussed in Section 1.1. Section 1.2 explains the objectives of this thesis and Section 1.3 provides the structure of the document.

1.1 Background

Already during the Second World War, Operations Research was used to optimise the defence strategies against German U-boats. The UK and the USA founded special groups with scientists from all sorts of expertise. The goal of these groups was to optimise the operations against the U-boats using scientific means [1]. Since then, the techniques and technology used in Operations Research have improved greatly, using the possibilities of computers for numerous complex calculations and analyses of enormous amounts of data. Nowadays, warfare against submarines still remains an active area of research and techniques for optimising this kind of warfare are still applied.

This Anti-Submarine Warfare (ASW) is one of the aspects of naval warfare. Examples of other aspects are Anti-Surface Warfare or Anti-Air Warfare. When a new naval warship is designed, it should perform to a desired level in all of these warfare aspects. Often it is not possible to assess the exact performances in real warfare scenarios in the design state of the project. Therefore, a warship is designed with the best possible equipment within the budget, hoping the performance will be sufficient. To change this, the Netherlands Defence Materiel Organisation (DMO) requires a system to quickly assess the effectiveness of new ship concepts in the different naval warfare aspects. This way, resources can be spent more effectively. TNO is developing a tool for this purpose in the Holon-project [2]. Part of this project is examining suitable simulation models. These models can simulate the effectiveness of new concepts. Multiple scenarios can be simulated for the different naval warfare aspects. One of the scenarios in the ASW domain is a sea-basing scenario. In this scenario an amphibious taskgroup (sea-base) needs to be protected against an underwater threat.

1.2 Study objective

In this thesis a model will be developed to simulate the ASW sea-basing scenario. This model will be developed using the agent-based modelling program MANA (Map Aware Non-Uniform Automata). The goal of this thesis is to compare this model in MANA with the UWT model (Underwater Warfare Testbed), developed by TNO.

This TNO model is an extensive model, designed to simulate underwater warfare with a high level of detail. This complexity makes it difficult to use for quick simulations on a conceptual level. The simpler MANA model could be better suited for this use. Despite a lower level of detail, the output and analysis should not differ much from reality or the UWT model, for concept ship models. Therefore a comparison will be made between the models. The models will first be compared looking at the results of the simulation. They will subsequently be compared on the level of insight in tactics and concepts, the possibilities to implement new tactics and ship concepts and their relative adaptability for changing situations.
This leads to the following research question:

What are the benefits and downsides of using MANA for simulating complex military operations compared with the UWT model of TNO, in terms of (1) the results of the simulations, (2) the insights in tactics, (3) the possibilities to implement new tactics and ship concepts and (4) the adaptability of the model for changing situations?

1.3 Structure of the thesis

To answer this question, the thesis starts with the necessary theoretical background in Chapter 2. This chapter describes the operational basics of ASW and the theory of modelling and simulations. This theory is necessary to comprehend the modelling of an ASW scenario in MANA and the comparison with other systems. In Chapter 3 the sea-basing scenario will be outlined. This scenario will be modelled in MANA, such as is explained in Chapter 4. The same scenario is modelled in the UWT program for a fair comparison. The results of simulations with the MANA model and the performance of the model are analysed in Chapter 5, in order to compare the modelling in MANA with the UWT. The models will be compared both qualitatively and quantitatively. Hereafter, more scenarios are modelled in MANA in order to test the limits of MANA. This way the adaptability and boundaries of MANA are analysed in Chapter 6. Chapter 7 ends with the conclusions of the thesis. This chapter summarises the findings of the preceding chapters and concludes the research.
2 Operational and Theoretical background

This chapter describes the theory needed to comprehend the contents of this thesis. To this end, some basic concepts of Anti-Submarine Warfare (ASW) are explained first. After this, the theory of simulations and models will be discussed, with a detailed look at the agent-based model MANA.

2.1 Anti-Submarine Warfare

ASW is one of the aspects of naval warfare. Other aspects can be Anti-Air Warfare (AAW) or Anti-Surface Warfare (ASuW). This section will give a global insight in ASW [3]. For a more detailed analysis, see ATP-28(B) ([3]) or AXP-5.

2.1.1 Aim of ASW

The aim of ASW is to deny the enemy effective use of their submarines. This can be achieved by detecting and possibly destroying an enemy submarine, or by deterring it from the area of operation.

2.1.2 Threat in ASW

The threat in ASW is a submarine (SS). This threat can vary from a massive nuclear powered submarine (SSN) to small coastal submarines (SSC). The weaponry can vary from torpedoes to missiles.

When a submarine wants to use a torpedo during an attack, it needs to come close enough to the target to make sure the torpedo is effective. The maximum distance within which the submarine can pose a threat with a torpedo is called the Maximum Effective Torpedo Firing Range (METFR). The area around a surface unit with this distance as radius, from which the submarine can pose a threat, is called the Torpedo Danger Zone (TDZ).

2.1.3 Principles of ASW

Due to the nature of submarines, the wide variety of types of ASW forces, and the effect of the environment, the following factors have a fundamental influence on ASW:

- The submarine operates beneath the surface of the ocean. Furthermore, it is manoeuvrable and can rapidly change depth, course and speed.
- It is possible that the position of an enemy submarine becomes known only after it has launched its torpedo.
- The ASW forces can be spread over a large area of operation. Thus, the ASW commander (ASWC) must delegate significant responsibility to unit level.
- Detection, classification, identification and localisation are all primarily carried out by acoustic sensors, both active and passive. Acoustic detection and counterdetection ranges vary significantly with equipment and environmental conditions.

2.1.4 Keys to success in ASW

The success of an ASW operation should not only be measured by the destruction of all enemy submarines. Depending on the specific mission and available forces, success can be achieved through deterrence, coercion, prevention, disruption, or destruction. For this success, the following factors can be of influence:

- The acoustic discretion of naval units, to force the submarine to Periscope Depth (PD) for target acquisition.
• The capability of the acoustic sensors and their operators to detect, classify, localize, identify, and track a submerged submarine.
• The capability of surface and subsurface units to effectively evade enemy torpedo attacks.
• The capability of the ASWC to effectively coordinate the deployment of all ASW forces.

2.1.5 ASW-capable units

For the execution of ASW, various units may be available. Which units are deployed, depends on the specific mission and the composition of the task force. Table 2.1 provides an overview of the units available to the Royal Netherlands Navy (RNLN) [3]. These units are helicopters, frigates (FF) with or without a Towed Array Sonar¹ (TAS) and their own submarines (SSK).

Table 2.1: Comparison of RNLN ASW platforms

<table>
<thead>
<tr>
<th>Platform</th>
<th>Capabilities</th>
<th>Limitations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASW HELO</td>
<td>Speed of response, unattainable for submarines</td>
<td>Limited duration</td>
</tr>
<tr>
<td>FF (w/o TAS)</td>
<td>Endurance, Command &amp; Control (C2) facilities, embarked helicopter</td>
<td>Low speed, short range sonars, vulnerable</td>
</tr>
<tr>
<td>FF (with TAS)</td>
<td>Endurance, C2 facilities, embarked helicopter, good sonar</td>
<td>Low speed, vulnerable</td>
</tr>
<tr>
<td>SSK</td>
<td>Endurance, operates in same environment as enemy submarine</td>
<td>Intermittent communication, low speed, needs to surface</td>
</tr>
</tbody>
</table>

In many naval operations, only a limited number of units are available for ASW. Therefore, deployment tactics are critical, as well as outstanding ship designs.

2.2 Simulation studies

Few real-world systems are simple enough to use analytical solutions to obtain exact information. Military operations, for example, are often too complex to allow realistic models to be evaluated analytically. These complex models can be evaluated numerically using simulations. The output data are gathered and analysed to estimate the true characteristics of the system [4].

Figure 2.1 [4] gives a schematic overview of the process in a general simulation study. These steps can be adjusted for different cases. The main observation, however, should be that a valid model of the real world needs to be created and that this model has to be programmed in a valid simulation model. These steps are the modelling phase of the study [5]. The steps hereafter are the user phase.

When a working model has been created, the user-phase starts. In this phase, the user runs a number of simulations to generate output. This output needs to be translated back to the real-world. The user will interpret the meaning of the output [5].

¹. For example the Low Frequency Active Sonar (LFAS)
The simulation models can be classified in many different ways, such as continuous vs. discrete event models or deterministic vs. stochastic models. A continuous model simulates a system continuous in respect to time. This means that the state of the system will be generated at fixed time intervals, the system applies time-slicing of length $\Delta t$. A discrete event simulation on the other hand calculates the time until the next event, and does not simulate in between two events. The time-steps in these models are not constant. A stochastic model uses random variables for some parts in the simulation, in contrast to a deterministic simulation, where every variable is known.

The model in the simulation program used in this thesis, MANA, is continuous and stochastic. The model is also agent-based.

### 2.3 Agent-based models

Agent-based modelling (ABM) is a technique used to break down complex phenomena, such as ASW-operations, into micro-behaviours with reduced complexity. The model is built with simple blocks, called agents. These agents interact with each other based on simple behavioural rules. Due to their interaction, the result of the simulation becomes a complex system [6].

The core of ABM are the agents. In typical ABM simulations, agents represent single human beings, but there are exceptions. For example, agents can be countries, ships or even groups of people. In general, something is an agent if, and only if [6]:

1. It acts autonomously.
2. It has individual goals and behaviour.
3. It interacts with other agents.
4. It is reactive.

Furthermore, agents can have spacial characteristics. This means the agent has a certain position within the environment. For the modelling of the ASW scenario this is critical, because the relative location of agents determines whether one agent can detect the other.

ABM is a suitable means for solving problems which in general have the following properties [6]:

- **Temporal aspects**: Agents act on behavioural triggers. This means they are embedded in time.
- **Information about behaviour at individual level**: ABM simulations are modelled on individual level. Therefore, data on this level must be available.
- **Medium numbers**: Although behaviour on individual level is important, the results on macro-level are the output. Having too few agents could emphasize the individual behaviour rather than the interactions. Having too many agents can be infeasible due to computational expenses.
- **Local interactions**: The complexity on the macro-level is caused by the interactions on the micro-level. So local interaction must be part of the problem.
- **Heterogeneity**: Most ABM simulation requires a heterogeneous population of agents, in order to create the effects on macro-level.

ASW scenarios satisfy at least four of these requirements. Only the requirement of medium numbers is used less stringently. As was explained in Section 2.1, the number of units in an ASW scenario is limited. This means that individual behaviour, and not the interactions, may affect the output greatly. This is not necessarily a disadvantage, because it reflects actual ASW operations. Therefore the behaviour of the units must be modelled with great care, to create actual reactions.

The program used in this thesis, is MANA. This is an agent based modelling program developed by the Defence Technology Agency (DTA), in New Zealand.

### 2.4 MANA

MANA is designed to be used as a scenario-exploring model. According to the designers of MANA, the problem with highly detailed physics-based models is often their lack of military mission-critical implementations, such as Situational Awareness (SA), Command and Control (C2) and sensor systems. The behaviour of forces on the battlefield is mostly influenced by these features and not by completely pre-determined factors. Furthermore, the level of detail of physics-based models may be higher than necessary for scenario-exploring simulations. These models squander time and resources using calculations that are too complex [7].

Therefore the goal of this combat model is to design models bottom-up: building the units itself to generate a whole battlefield. This way the essence of the situation is captured and non-essential details are avoided. Furthermore, MANA uses the following key concepts [7]:

- **SA**: Agents build their own SA and can share this with other agents.
- **Communication**: Communication between agents is possible. The links are adjustable using a range of parameters.
- **Waypoints**: A route of waypoints can be defined, not just an ultimate goal.
- **Event-driven Personality Changes**: Events such as detecting an enemy can trigger a different personality set. Agents have reactive behaviour.
### 2.4.1 Movement in MANA

Agents decide which way to move through attraction or repulsion from other agents, waypoints or terrain features. This attraction or repulsion can be set by the user. In MANA-V (the version used in this thesis) movement is vector-based. Each agent monitors all other agents and terrain features within sensor range. A vector is calculated toward each agent or feature. Vectors can also be calculated toward distant agents appearing on the agent’s shared SA map, who are not necessarily in sensor range. This is visualised in Figure 2.2 [8]. Weights are added to each vector. These weights are the personality weightings defined by the user. The resulting vector is calculated with Equation 2.1.

\[ F = \sum_{i=1}^{N} w_i \cdot F_i \]  

(2.1)

Where \( F_i \) is the individual vector towards phenomenon \( i \) (an agent, a waypoint or terrain). \( w_i \) is a combination of the personal weighting and a distance-dependant factor, so that agents further away have less influence [8]. With this vector and the time-step \( \Delta t \) the new position is calculated using Newton’s second law \( a = \frac{F}{m} \) and the standard kinematic equations for constant acceleration. The value of \( m \) is arbitrary, with \( m \) small \((m < 1)\) for quick accelerations and \( m \) large \((m >> 1)\) for slow accelerations. \( S_0 \) and \( v_0 \) are the position and speed in the previous time step.

\[ S = S_0 + v_0 \Delta t + \frac{1}{2} a \Delta t^2 \]  

(2.2)

\[ v = v_0 + a \Delta t \]  

(2.3)

It should be noted that the value of \( v \) is determined by the current state of the agent and that vector \( F \) only affects the direction of \( v \).

### 2.4.2 Sensors in MANA

MANA has two ways to model the agent’s sensors, a simple and an advanced sensor model. The simple model provides a full circle response cookie-cutter model of both detection and classification. The advanced model has more parameters to set. In this model a range-integration time profile for both detection
and classification can be set. Furthermore, the advanced model offers the ability to choose particular classes the sensor can see. In both models the sensor aperture can be set, to model a blind sector [7][8]. Appendix A discusses the modelling of a sensor in more detail.

2.4.3 Communication in MANA
Communication between agents is also possible. In MANA, links between agents can be set to share their SA. This way, agents can see agents who are not within their own sensor range. The parameters of this link, such as range, capacity or accuracy, can be set by the user to model real military links [7].

2.4.4 Weapons in MANA
MANA also has the ability to model military weapons. These can be either kinetic energy weapons or high explosive weapons [7]. The effectiveness and range can be set to simulate all kind of military weapons. The weapon model, however, is limited to dumb weapons. Guided ammunition, such as missiles or torpedoes can not be implemented in this model by default.

2.5 Stochastic properties of simulations
When simulating the model, some parameters are variable. In ASW scenarios, for example, the precise direction of approach of an enemy submarine is unknown. This parameter is a random variable. When a simulation with stochastic parameters like these is built, Monte-Carlo methods can be used. Monte-Carlo simulations use randomly generated numbers for the stochastic variables. A key property of this method is to repeat the simulation a large number of times, each with other random numbers for the stochastic parameters. This way, the influence of an unknown parameter can be simulated [9].

Scientific research must be repeatable, so stochastic simulations as well. Additionally, different policies must be compared in a fair way. This can be achieved with set seed values for a random generator. When a random generator is used to produce stochastic parameters, a specific seed value will generate a specific number. The random numbers in MANA are obtained using a built-in Delphi function, as this is the language MANA is written in. This function uses a pseudo random number generator with a cycle of $2^{32}$ and a 32 bit seed. This means it can approximately take on $4.3 \cdot 10^9$ values before the cycle will repeat. When a MANA simulation is started, the Delphi generator is randomized with a seed value. This seed can be set by the user.

In a MANA model, random numbers are used for determining fire and moving orders and for weapon and sensor specification when advanced models are in use.

2.6 Validation and verification of simulations
During the development of a simulation model, the model has to be validated. Validating a model is the process of establishing in what way the model gives an accurate description of reality, when looking at the intended output [10]. Both the model itself and the implementation of the model have to be validated. The validation of the implementation is called verification. The questions you should ask yourself is: Did I model the right thing and did I model it correctly?
The first part of the question, is the model itself a good representation of reality, could be answered by expert assessments. They have the knowledge to assess whether the model is right.

For the second part, there are various ways for the verification of a model. One way is to compare the output of a simulation with an analytical calculation. This is possible for simple models. When the simple variant of a model is validated this way, an assumption can be made that more complex variants of the model are also valid. Another way of verification is comparing the input-output relation with other (valid) simulating models. As the model in this thesis will be compared with the UWT from TNO, this way of verification is used anyway as part of the comparison between MANA and the UWT.

2.7 Reliability of simulations

After a set of simulations, the output can be analysed. The meaning and interpretation of the output is one important aspect of this analysis, but the reliability of the output is certainly another important aspect.

One way to examine the reliability is to calculate the confidence interval (CI) of the output. In a simulation batch, the parameter of interest will be the probability of detecting a submarine by the ASW task force. Since the output of a single run in the scenario is either submarine detected (success) or submarine not detected (failure), and the output of multiple runs is mutually independent, the number of successes in \( n \) runs has a binomial distribution, with parameters \( n \) and \( p \), where \( n \) is the number of simulation runs and \( p \) the probability of success. \( p \) can be estimated with Equation 2.4, where \( X \) is the number of successes.

\[
\hat{p} = \frac{X}{n} \tag{2.4}
\]

A binomial proportion confidence interval can be calculated around this value. When the assumption is made that the distribution of the error can be approximated with a normal distribution, the formula for the CI is given in Equation 2.5. This assumption can be justified by the central limit theorem, when \( n \) is large enough. In Equation 2.5, \( \hat{p} \) is an estimation of \( p \) and \( z_{\alpha/2} \) is the \( (1 - \frac{\alpha}{2}) \) percentile of a standard normal distribution [11].

\[
CI = \hat{p} \pm z_{\alpha/2} \cdot \sqrt{\frac{\hat{p}(1 - \hat{p})}{n}} \tag{2.5}
\]

As the use of this approximation is justified by the central limit theorem, the sample size of a series of simulations should be greater than 30. Also the equation applies poorly when \( p \) is close to 0 or 1. A rule of thumb for this case is that \( n \cdot \min(p, 1 - p) \geq 5 \). When these conditions cannot be satisfied, other more complex methods must be used.
3 Scenario set-up

To compare the model in MANA with the simulations of TNO, the same scenario will be used. This scenario is composed in the Holon-project. This chapter will give an outline of this scenario \cite{12}[13]. For a more extensive description of the scenario, including choices for tactics and calculations for sensor ranges, see the TNO report *Planning of underwater defence in expeditionary scenarios* \cite{14}.

3.1 Generic scenario

A small RNLN task force is in position for amphibious operations in an area of operation. The goal of the task force is to land units and material on shore. This stationary sea-base has to be protected against enemy threats, including submarines. To this end, an ASW task group is formed. This task group consists of ASW-frigates and helicopters. Their goal is to protect a Joint Support Ship (JSS) or a Landing Platform Dock (LPD). These ships are the High Value Units (HVU) of the task force.

The threat is a simple diesel-electric submarine (SSK), trying to approach the task force and attack the HVUs with a torpedo. Assumed is that the SSK will not attack any of the frigates, as the strategic value of a frigate is too low compared with the strategic value of the SSK itself.

The defending units follow a search pattern trying to detect the SSK before it enters a region around the HVU where it can launch a torpedo. When an SSK is positively detected by one of the units, the task force will try to prevent the threat of executing its mission. This can be achieved by attacking it, prosecuting it, or relocating the task force. However, this scenario focuses on detecting the enemy submarine. Torpedo attacks or hampering the SSK will therefore not be simulated.

3.2 Base scenario and tactics

In the base scenario, the HVUs are protected by Multi-purpose frigates (MFF) and NH-90 helicopters from the RNLN. There will be three variants:

1. 1 MFF and 1 NH-90 (continuously on station)
2. 2 MFFs and 1 NH-90 (continuously on station)
3. 1 MFF and 2 NH-90s (continuously on station)

Assumed is that the task force has enough helicopters to continuously keep one (or two) on station. In all variants, the frigates circle around the HVU, following an octagon. The helicopters follow the octagon as well, using a dipping sonar at the vertices of the octagon. This configuration is visualised for all three variants in Figure 3.1. All ASW units are spread evenly over the octagon. To maintain this, the dip time of the NH-90 is such as to keep the speed of advance the same as the MFF.

The choice for the octagon is made in the HOLON-project because it is an unclassified tactic. It is a simple symmetric strategy, where the number of vertices is limited. This is because sonars such as the LFAS are less effective when this ship is turning.

The defensive forces have the goal to keep the enemy submarine from firing a torpedo at a HVU. Therefore, the SSK should not come closer to the HVU then the Maximum Effective Torpedo Firing Range (METFR) of the SSK.
3.2.1 Sensors

The defending forces have various sensors available for detecting and classifying the SSK. To simplify the simulation, only the best sensor will be modelled. The MFF will use the towed Low Frequency Active Sonar (LFAS) and the NH-90 a Helicopter Long-Range Active Sonar (HELRAS), which is a dipping sonar. These sonars are visualised in Figure 3.2.

In real operations, the sonar operator needs to positively classify the contact as a submarine. The contact can be classified as NonSub (Nonsubmarine contact), PosSub Low (low possibility the contact is a submarine), PosSub High (high possibility the contact is a submarine), ProbSub (probable submarine contact), or CertSub (certain submarine contact). Which level of classification is assigned to a contact depends on which Contact Classification Terms apply for the contact [15]. This specific classification is not simulated, but to simulate the number of checks which are performed to achieve a sufficient level classification (chart-checks, localisation, correlation with other sensors, etc.), the range of the classification is lower than the detection-range. The actual values of these ranges depend on the following factors.

- The environment: The sound speed profile in the water depends on the state of the sea (salinity, depth and temperature). This influences the performance of a
• The sonar: The choice for the sonar pulse (CW (Continuous Wave) or FM (Frequency Modulated)) and the depth of the sonar influence the range against different targets.
• The threat: The performance of a sonar depends on the depth and speed of the submarine.
• Own speed: The speed of the unit using the sonar influences the performance of the sonar. A higher speed results in more noise.

For the detection and classification a cookie-cutter principle will be maintained. The ranges used in this scenario are presented in Table 3.1 [13].

Table 3.1: Sonar performances of defending units

<table>
<thead>
<tr>
<th>Sonar</th>
<th>Detection range [km]</th>
<th>Classification range [km]</th>
</tr>
</thead>
<tbody>
<tr>
<td>LFAS (MFF)</td>
<td>12</td>
<td>9.144</td>
</tr>
<tr>
<td>HELRAS (NH-90)</td>
<td>12</td>
<td>9.144</td>
</tr>
</tbody>
</table>

In the base scenarios, the investigation of unknown contacts is not simulated. The ASW units will not react on contacts who are detected, but not yet classified. This could be examined in more advanced tactics, in which false contacts are simulated as well. So in the base scenarios, the detection range has no effect on the course of the simulation, only the classification range has an effect on the outcome.

3.2.2 Weapon deployment

The goal of this scenario is to timely detect an enemy submarine. The actions after the detection are not of interest for this study. Therefore, the weapon deployment or evasive actions of the defending units are not modelled in this scenario.

3.2.3 Threat description

The underwater threat for the base scenario is a conventional diesel-electric submarine. The strategic goal of the submarine is to disrupt the amphibious operations of the task force in the region. Therefore its tactical goal is to locate and destroy the HVUs of the task force, from which it knows the approximate location. Since its goal is to disrupt the task forces capabilities, it is assumed that it will not attack the frigates in the task group defending the HVUs. This is typical in the Rules of Engagement (ROE) for commanders of submarines nowadays.
For operational picture-compilation the SSK uses only passive sonars. The sensors are also cookie-cutter modelled, with different ranges as counter-detection ranges for the different units in the task force. With these sonars it will try to reach the HVU undetected. For this goal it can use various tactics, of which three possibilities are simulated: The kamikaze approach, the semi-kamikaze approach and the cautious approach [16].

In the kamikaze approach the SSK will follow a straight line to the approximate location of the HVU with a speed of S1 knots, until it detects the exact location. In this tactic it will not try to evade searching units. This tactic is visualised in Figure 3.3. This tactic sounds rather harsh, but it is not used as a real-world tactic. It is used as a measurement of a tactic or concept without enemy behaviour. A low detection probability in this tactic means big gaps in the defence. It is a baseline for the other simulations with an actual form of enemy behaviour.

![Figure 3.3: The kamikaze approach of the SSK](image)

When the SSK wants to avoid detection, it can use the semi-kamikaze or the cautious approach. In the semi-kamikaze approach it sets out the same way as the kamikaze approach, but it will abort his approach for a short time when it detects a searching unit, whether it is a frigate or a helicopter. It will lay dead in the water until it can continue its approach undetected. This is visualised in Figure 3.4. This approach has benefits over the kamikaze approach, because the SSK can see the ASW-units before they can see him; the counter-detection range is always greater than the detection range when active sensors are involved.

![Figure 3.4: The semi-kamikaze approach of the SSK](image)

The final tactic is the cautious approach. In this tactic the SSK tries a new path when it detects an ASW-unit. It actively avoids the searching units. This is visualised in Figure 3.5 for detecting an ASW-surface unit (MFF). The SSK will try to
keep this surface unit at a distance of R1 km, by adjusting its course and sailing aft of the surface unit. When the SSK detects the HELRAS of an NH-90, it will face the HELRAS and stop its further movement. It will lay dead with a minimal reflecting surface, to minimize the detection probability of the HELRAS.

Figure 3.5: The cautious approach of the SSK

3.3 Measurement of Effectiveness

The main measurement of effectiveness (MoE) for a certain tactic in these scenarios is defined as:

The probability that the task group can detect an approaching SSK before the HVU comes within the METFR of the SSK.

The defending forces can achieve this goal in two ways. Either by detecting the SSK in time, or by deterring the submarine from coming close enough to the HVU. The opposing force wins when the SSK can come within METFR of the HVU.

This MoE shows the effectiveness of a strategy as a whole, but not weak or strong points within it. The probability of detecting the enemy submarine for individual units shows their own effectiveness. Therefore a secondary MoE is defined as the probability of detecting the SSK by individual ASW-units in the defence strategy. This MoE may clarify certain strong and weak points within the strategy.

The effect on the MoE in a certain scenario can be measured by a sensitivity analysis. In this analysis, the influence of a certain parameter is examined. When a small change in this parameter has a large effect on the MoE, this parameter is of great importance in the concept or tactic.
4 Modelling the scenarios

This chapter will provide an outline of the models built for this thesis. First an analytical model is presented for the simple variants. Hereafter the simulation models will be discussed.

4.1 Analytical approximation

For the kamikaze approach, an analytical model can be created. The octagon defence patrol can be modelled with a $1^{st}$ order approximation with a linear patrol, where the threat crosses perpendicular to the course of the defenders with a uniform distribution over the length of the patrol. This situation is a linear barrier patrol [17]. Figure 4.1 shows this linear patrol. In this figure $L$ is the length of a full circle in the octagon, and therefore the length of the barrier patrol. $V_i$ is the speed of unit $i$ and $R_i$ the detection/classification range of unit $i$. The SSK approaches perpendicular to the barrier.

![Figure 4.1: Linear barrier patrol with 1 MFF and 1 NH-90.](image)

For an analytical calculation of the probability of detection in Figure 4.1, the visualisation in Figure 4.2 can be used. In this figure, the speed of the ASW-units relative to the speed of the SSK is drawn, instead of the absolute speed of all units. This means the submarine can be placed motionless anywhere in the rectangle with length $L$ and width $B = L \cdot \tan \alpha$. In this model is $\alpha = \arctan \left( \frac{U}{V_f} \right)$, with $V_f$ as the speed of the frigate and $U$ as the speed of the SSK. The probability of detecting the submarine is then the proportion of the rectangle covered by the sensors of the ASW-units. In Figure 4.2 this is the green area divided by the sum of the green and gray areas.

After simplification, the equation for the probability of detection by the frigate is given in Equation 4.1 and for the NH-90 in Equation 4.2. Appendix B discusses the derivation of the formulas.

\[
P_{\text{frigate detection}} = \frac{2R_f}{L} \sqrt{1 + \left( \frac{V_f}{U} \right)^2} - \frac{R_f^2}{L^2} \left( 1 + \left( \frac{V_f}{U} \right)^2 \right), \text{ if } L \gg R_f
\]  

\[
P_{\text{helo detection}} = \frac{2 \cdot R_h \cdot U \cdot t_{\text{disp}} + \pi \cdot R_h^2}{L \cdot U \cdot \left( \frac{1}{R_h} + t_{\text{disp}} + t_{\text{set}} \right)}, \text{ if } R_h \geq 0.5 \cdot l \text{ and } L \gg l
\]
Figure 4.2: The analytical model with 1 MFF and 1 NH-90 for a kamikaze approach

Where:
- \( R_f \) = Detection range of the MFF.
- \( R_h \) = Detection range of the NH-90.
- \( V_f \) = Speed of the MFF.
- \( V_h \) = Speed of the NH-90.
- \( U \) = Speed of the SSK.
- \( L \) = Length of linear patrol.
- \( l \) = Length of a single leg.
- \( t_{dip} \) = Time the helicopter is in active dipping position.
- \( t_{set} \) = Time the helicopter needs for setting and recovering the dipping sonar.

The combined probability of detection can be given by Equation 4.3. In this equation, \( N_{frigate} \) is the number of searching frigates and \( N_{helo} \) the number of searching helicopters. \( \omega \) is a correction for overlap in the sweep areas between two units. When there is no overlap, \( \omega = 0 \).

\[
P_{\text{detection}} = N_{frigate} \cdot P_{\text{frigate}}^{\text{detection}} + N_{helo} \cdot P_{\text{helo}}^{\text{detection}} - \omega \tag{4.3}
\]

It should be noted that this model is a \( 1^{\text{st}} \) order approximation of the real scenario. For the movement of the helicopter this has little impact. This is because the relative motion of the helicopter is always directed at the submarine at the moments the helicopter could detect the SSK, since the absolute speed of the helicopter is 0 at those moments. And moreover, the detection circles of the different dip-positions at the vertices do not overlap each other.

For the frigate, however, it can make a difference. In the cases where the submarine approaches from a direction where one of the vertices of the octagon is located, the SSK approaches the leg of the frigate at an angle of \( 22.5^\circ \). Furthermore, the absolute, and thus relative, movement of the frigate is different per leg. On the leg prior to the leg where a detection could be made, the relative movement is more directed towards the SSK. Taking only this into account, it would
mean that in the octagon the probability of detection is slightly higher. However, at the vertices there is an overlap of the swept area, which would give a slightly lower probability of detection.

Combining these differences, the analytical model will probably give a somewhat conservative approximation of the probability of detecting an SSK by a MFF.

4.2 The simulation framework

The scenarios are modelled in MANA. As has been explained in Chapter 2, MANA is suitable to build the military models itself, due to the blocks for modelling C2, weapon and sensor systems. However, it is not well suited to simulate multiple runs with a range of input parameters. Also MANA offers no useful tools for preparation or analysis of a simulation batch. For these purposes, a simulation framework is designed and built.

Figure 4.3 shows the framework designed in this thesis. The structure has three phases, divided over three layers: the model layer (green), the engine layer (red) and the simulation layer (blue). The first phase is the preparation phase, which starts in the model layer, where the base model is built. This is done using MANA. The model in MANA is then saved as an XML file in the engine layer. This file can be adjusted by the simulation layer for one or multiple runs. The simulation layer can adjust parameters such as the starting position of the SSK or the speed of the ASW-units. These adjustments are controlled by Matlab. After this implementation and the adjustment, the simulation phase starts, where the adjusted file will be ran in MANA-C, the command version of MANA. This program runs in the background and does not squander time and memory on the visual simulations. The output of these simulations is written in CSV files. In the analysis phase, the output is read and processed by Matlab for the analysis of the simulation output.

![Figure 4.3: The structure of the simulation framework](image)

4.3 The model layer

This section will discuss a few important settings in MANA. A complete outline of the models in MANA is given in Appendix D.
4.3.1 ASW-units behaviour

In all base scenarios, the defending ASW-units follow the set octagon, defined as 8 looped waypoints. Their movement vector is only defined by their next waypoint. Both the frigates and the helicopters have a cookie-cutter sonar with a fixed range and no blind sectors.

Also, both ASW-units kill the SSK the moment they classify it. Although this is not realistic, this is done in order to create more readable output, as the casualty log is better registered in MANA than the detection log. A successful classification is measured more quickly this way. Furthermore, a kill can be used as a stop condition, so the simulation run can be terminated after a classification.

In the base scenario, the frigates operate only in the default state. Their behaviour is constant throughout the mission. This means they sail with constant speed from waypoint to waypoint.

The helicopters have 4 states, to simulate the dipping process. They fly to a waypoint in the default state with constant speed and zero detection range. When they arrive at a waypoint, they wait 6 minutes with zero detection to simulate the lowering of the HELRAS. After 6 minutes their state changes and their classification range is set on \( R_{\text{classification}} \) for \( t_{\text{dip}} \) seconds. Finally they change to a state where they wait 6 minutes with zero detection to simulate the recovering of the HELRAS. Then they fly to the next waypoint. This process is summarised in Table 4.1.

<table>
<thead>
<tr>
<th>State</th>
<th>Classification range [km]</th>
<th>Speed [kts]</th>
<th>Duration [s]</th>
<th>Next state</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default</td>
<td>0</td>
<td>100</td>
<td>Flight time</td>
<td>-</td>
</tr>
<tr>
<td>Reach Waypoint</td>
<td>0</td>
<td>0</td>
<td>6 \cdot 60</td>
<td>Spare 1</td>
</tr>
<tr>
<td>Spare 1</td>
<td>9.144</td>
<td>0</td>
<td>Diptime</td>
<td>Spare 2</td>
</tr>
<tr>
<td>Spare 2</td>
<td>0</td>
<td>0</td>
<td>6 \cdot 60</td>
<td>Default</td>
</tr>
</tbody>
</table>

4.3.2 SSK behaviour

The SSK can have three types of behaviour. A behaviour for the kamikaze approach, a behaviour for the semi-kamikaze approach and a behaviour for the cautious approach. Each approach, and thus behaviour, is modelled slightly differently.

In all three types of approach, the SSK has at least 2 sensors. One is used to detect the active LFAS and HELRAS sounds. This sonar has a range greater than the detection range of these sonars themselves \( (R_{\text{counter}} = 2 \cdot R_{\text{classification}} = 18.29\text{km}) \). The other sensor is used to detect the engine sounds of the HVU. The range of this sensor is considerably smaller, and set to 9 km. The NH-90 is invisible for the SSK when the helicopter is not actively searching with its HELRAS.

For the kamikaze approach the behaviour is straightforward. The SSK approaches the center of the defence screen directly, until it detects the HVU and attacks it. Detecting the MFF or NH-90 does not affect its actions.

When the SSK uses the semi-kamikaze approach, it should only lay still when the ASW-units can pass in front of the SSK without detecting the submarine. When this is no longer possible, the SSK should sail through and hope it can pass in front of the ASW unit in time. To simulate this, the sonar of the SSK has a confined aperture. This is visualised in Figure 4.4. The width of the aperture
is calculated so, that the SSK will detect an ASW-unit with which it has an CPA (Closest Point of Approach) less than the classification range of that ASW-unit. When the absolute speeds are perpendicular, angle $\beta$ is calculated with Equation 4.4, using the relative speed of both units. $V$ is the speed of the frigate and $U$ is the speed of the SSK. The derivation is discussed in Appendix B. With this sensor, the SSK will wait when it detects an ASW-unit, and continue when nothing is detected.

\[
\beta = \arctan\left(\frac{V}{U}\right) + 0.5 \cdot \arctan\left(\frac{U}{V}\right)
\]

(4.4)

In the model for the cautious approach the SSK reacts actively when it detects an ASW unit. Its behaviour states are as follows:

- When no enemies are detected: Follow shortest path to the center of the sea base, this is the approximate location of the HVU.
- Detecting an LFAS: Adjust course to pass the MFF $R_1$ [m] aft. Still close the distance to the center of the sea base.
- Detecting an active HELRAS: Lie still and face the direction of the HELRAS, when within $R_1$ [m] of the HELRAS. This is to minimize the reflecting surface. This action results in a smaller classification range for the HELRAS in the model.
- Detecting HVU: Close distance to HVU, keeping above actions in account.
- Clear zone: Range $R_1$ is the distance the SSK wants to stay clear of the ASW-units. This is set on $1.3 \cdot R_{\text{classification}} = 11.9$ [km].

This behaviour is summarised in Table 4.2, where the trigger states in MANA are given, with their impact. In this table, and in the model, an enemy threat 1 is a HVU, an enemy threat 2 a HELRAS and an enemy threat 3 an LFAS.

### 4.4 The simulation layer

A short outline of the Matlab files is given in this section. The complete view of the Matlab codes is given in Appendix E.

When a scenario is modelled in MANA and written as an XML-file, it can be used in the simulation layer. All the actions in this layer are executed by Matlab.

In one run, the probability an ASW-unit can detect an SSK greatly depends on its position at the moment the SSK appears. To take this into account, the simulation layer can set the starting position of the SSK at various positions.
Table 4.2: SSK trigger states for cautious approach

<table>
<thead>
<tr>
<th>State</th>
<th>Meaning</th>
<th>Speed</th>
<th>Behaviour</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default</td>
<td>Nothing detected</td>
<td>6</td>
<td>Nothing is detected. Approach to center of sea-base.</td>
</tr>
<tr>
<td>Enemy contact 2</td>
<td>Active HELRAS detected</td>
<td>0</td>
<td>NH-90 is detected. Stop all movement and face contact.</td>
</tr>
<tr>
<td>Enemy contact 3</td>
<td>LFAS detected</td>
<td>6</td>
<td>M-frigate is detected. When within $R_1$ km, steer away. When further away, pass $R_1$ km aft of frigate.</td>
</tr>
<tr>
<td>Enemy contact 1</td>
<td>HVU detected</td>
<td>6</td>
<td>HVU detected. Forget all else and approach to HVU.</td>
</tr>
</tbody>
</table>

For a fair comparison with the UWT, these starting positions are set on predefined locations. The threat direction is varied in steps of 15° and the range to the center of the sea-base is varied from 45 km to 120 km in steps of 5 km. So there are $24 \cdot 16 = 384$ fixed initial positions. The distance range from 45 to 120 is chosen such that the ASW-units make one full circle along the octagon in the time the SSK takes to travel this 75 km. This way all combinations of the defence and offence positioning are guaranteed. The distribution is visualised in Figure 4.5. Figure 4.6 shows the structure of the Matlab-file for the simulation using the fixed initial positions.

However, the 384 predefined locations are not sufficient for the analysis of the tactics or concepts. With those positions, there is a dependence between the waypoints of the octagon and the direction of the threat. This can greatly affect the output of the simulation, especially when the number of directions from which the SSK can approach is limited. In order to assure an independence, the start positions for the analysis are randomly chosen. The distance to the center of the
sea-base is a uniform distribution from 45 km to 120 km and the direction is a uniform distribution from $0^\circ$ to $359^\circ$. The randomness is placed in MANA, and not in Matlab. This way, a large batch of simulations can be run in MANA and the speed per simulation increases greatly. This distribution is visualised in Figure 4.7 with 7500 random positions. For all simulations in this thesis, the number of 7500 runs is used, as this number of runs gives a 95%-CI of around $\pm 1.0\%$. Figure 4.8 shows the structure of the Matlab-file for the simulation with random positions.
Figure 4.7: 7500 randomly distributed positions for the SSK

Figure 4.8: Structure diagram of simulation file with random initial positions SSK
5 Analysis of the models

This chapter contains the analysis of the base scenarios and tactics. First the verification of the MANA model with the analytical model is examined in Section 5.1. Then the quantitative results of UWT and MANA are compared in Section 5.2. Third, tactics and concepts are analysed using the MANA simulations in Section 5.3, to see how well MANA is suited for evaluating concepts and tactics. The evaluation in MANA can be visualised with graphics to demonstrate the strengths in such a scenario. After this, a sensitivity analysis of the scenarios in MANA is given in Section 5.4. Finally, using the insights obtained from the analyses, a qualitative comparison between MANA and UWT is made in Section 5.5.

5.1 Verification with analytical model

For the verification of the MANA models with the analytical model, the kamikaze approach is simulated with the 7500 random initial positions for the SSK against defence strategy 1. The output of these simulations and the analytical results are given in Table 5.1. The CI given with each probability is calculated using Equation 2.5, with $\alpha = 0.05$.

Table 5.1: Comparison between analytical model and MANA, for the kamikaze approach with random initial positions for the SSK against 1 MFF and 1 NH-90

<table>
<thead>
<tr>
<th>Probability</th>
<th>MANA [%]</th>
<th>Analytical [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_{\text{detection}}$</td>
<td>54.5 ± 1.1</td>
<td>51.3</td>
</tr>
<tr>
<td>$P_{\text{frigate}}$</td>
<td>28.2 ± 1.0</td>
<td>25.2</td>
</tr>
<tr>
<td>$P_{\text{helo}}$</td>
<td>26.3 ± 1.0</td>
<td>26.1</td>
</tr>
</tbody>
</table>

As was reasoned in Section 4.1, the analytical probability for the frigate would be conservative but the analytical probability for the helicopter should match. The output of Table 5.1 corresponds with this statement. The analytical probability of detection for the helicopter lies in the 95%-CI of the simulation, and the probability for the frigate is slightly lower in the analytical model. This leads to the conclusion that the simple variants of the model are verified. It can be assumed that the implementations of the semi-kamikaze and cautious approach are also valid.

5.2 Quantitative comparison between MANA and UWT

For a good quantitative comparison between MANA and UWT, the initial positions of the SSK in the scenarios are kept the same. The output from both MANA and UWT of all scenarios with the three approaches and the three defence options are given in the next three subsections. The comparison of the output will also be used as verification of the simulations. Not only the output, but the time it took to simulate is logged as well. The quantitative analysis of the strategies themselves through MANA is given in Section 5.3.

When simulating the kamikaze approach against defence strategy 1, both MANA and the UWT took about 23 minutes. For this test, both programs were run on the same computer. Although both programs can still be optimised for the simulation time, they perform similarly in their current state.
In MANA however, the speed per run will increase greatly when random positions are used for the SSK and a full batch of \( n \) runs can be simulated instead of \( n \) single runs that have to be defined separately.

### 5.2.1 Comparison of the kamikaze approach

First the results for the kamikaze approach are compared for all three defence strategies. The distance from the center to the legs of the octagon in these strategies is set to 25000 yards (=22860 m). Table 5.2 gives the overall results for both MANA and UWT, from 384 runs with a fixed distribution for the initial positions of the SSK.

Table 5.2: Comparison for kamikaze approach, against all three defences, with fixed initial positions for SSK

<table>
<thead>
<tr>
<th>Defence</th>
<th>MoE</th>
<th>MANA [%]</th>
<th>UWT [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( P_{\text{detection}} )</td>
<td>49.0</td>
<td>47.7</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{frigate}} )</td>
<td>27.9</td>
<td>27.9</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{helo}} )</td>
<td>21.1</td>
<td>19.8</td>
</tr>
<tr>
<td>2</td>
<td>( P_{\text{detection}} )</td>
<td>76.6</td>
<td>75.0</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{frigate1}} )</td>
<td>27.3</td>
<td>27.3</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{frigate2}} )</td>
<td>27.9</td>
<td>27.9</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{helo}} )</td>
<td>21.4</td>
<td>19.8</td>
</tr>
<tr>
<td>3</td>
<td>( P_{\text{detection}} )</td>
<td>67.2</td>
<td>64.3</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{frigate}} )</td>
<td>25.0</td>
<td>24.7</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{helo1}} )</td>
<td>21.1</td>
<td>19.8</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{helo2}} )</td>
<td>21.1</td>
<td>19.8</td>
</tr>
</tbody>
</table>

Table 5.2 shows that the output from MANA matches the output of the UWT for the kamikaze approach. The difference in probability for the NH-90 between both programs is caused by the fact that in MANA an agent does not need to be exactly at a waypoint. A radius around the waypoint is defined wherein the agents must come. This is because an agent must travel a full distance in one time step according to its vector. For the ASW scenarios in MANA this means the helicopter sometimes detects an SSK which travels right through the middle of an octagon leg. When the NH-90 is exactly at the waypoint, as in the UWT, the SSK is invisible. This is visualised in Figure 5.1.

![Figure 5.1: Diping positions of NH-90 and the blind spot](image-url)
This difference is acceptable, because when the initial positions of the SSK would be random instead of predefined, the misalignment of the helicopter at its way-point would not affect the output.

5.2.2 Comparison of the semi-kamikaze approach

The results of the semi-kamikaze approach are summarised in Table 5.3. The defence octagon and the initial positions are identical to the previous approach.

Table 5.3: Comparison for the semi-kamikaze approach, against all three defences, with fixed initial positions for SSK

<table>
<thead>
<tr>
<th>Defence</th>
<th>MoE</th>
<th>MANA [%]</th>
<th>UWT [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$P_{detection}$</td>
<td>38.3</td>
<td>37.5</td>
</tr>
<tr>
<td></td>
<td>$P_{frigate}$</td>
<td>22.7</td>
<td>22.4</td>
</tr>
<tr>
<td></td>
<td>$P_{helicopter}$</td>
<td>15.6</td>
<td>15.1</td>
</tr>
<tr>
<td>2</td>
<td>$P_{detection}$</td>
<td>70.3</td>
<td>61.7</td>
</tr>
<tr>
<td></td>
<td>$P_{frigate1}$</td>
<td>29.4</td>
<td>24.2</td>
</tr>
<tr>
<td></td>
<td>$P_{frigate2}$</td>
<td>22.4</td>
<td>22.4</td>
</tr>
<tr>
<td></td>
<td>$P_{helicopter}$</td>
<td>18.5</td>
<td>15.1</td>
</tr>
<tr>
<td>3</td>
<td>$P_{detection}$</td>
<td>59.4</td>
<td>54.9</td>
</tr>
<tr>
<td></td>
<td>$P_{frigate}$</td>
<td>28.4</td>
<td>24.7</td>
</tr>
<tr>
<td></td>
<td>$P_{helicopter1}$</td>
<td>15.4</td>
<td>15.1</td>
</tr>
<tr>
<td></td>
<td>$P_{helicopter2}$</td>
<td>15.6</td>
<td>15.1</td>
</tr>
</tbody>
</table>

Table 5.3 shows an difference between the outputs of MANA and the UWT. Against defence strategy 1 this difference is not significant, as both programs give similar results. Yet against an extra defence unit the programs differ. This difference in results is caused by the difference in modelling. The moment and distance at which the approaching SSK will stop its movements is defined less efficiently in MANA, therefore the SSK waits too long and is caught by next defence unit. This appears to be the unit after a helicopter (MFF 1 in both defence 2 and 3).

However, compared with the results of the kamikaze approach in Table 5.2, a similar trend is visible in both programs: evading the defence units results in a clearly better chance of success for the SSK against all three defence strategies.

5.2.3 Comparison of the cautious approach

Finally, in Table 5.4 the results are shown for the cautious approach. The same defence octagon and initial positions are set as in the last two approaches.

Table 5.4 shows a difference between the output as well. The cautious behaviour of the SSK seems to be more effective against a frigate in MANA and more effective against a helicopter in the UWT. These differences in the output arise through the differences in modelling. For the reaction against the NH-90, for example, the decrease in reflecting surface of the SSK is modelled by a smaller classification range for for the NH-90 (see Subsection 4.3.2). In the UWT, the NH-90 needs at least 2 sonar pings for a positive classification of the SSK, one for detection and one for classification. If the SSK would lie still before the second ping at the right distance, in between the original classification range and the decreased classification range, the NH-90 will not classify the SSK and the SSK can continue its approach. In MANA, however, the sensor of the NH-90 will detect and classify the submarine instantaneously if it is in range. The reaction against a frigate is also
Table 5.4: Comparison for the cautious approach, against all three defences, with fixed initial positions for SSK

<table>
<thead>
<tr>
<th>Defence</th>
<th>MoE</th>
<th>MANA [%]</th>
<th>UWT [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$P_{\text{detection}}$ 31.3</td>
<td>25.8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$P_{\text{frigate}}$ 13.1</td>
<td>15.4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$P_{\text{helo}}$ 18.2</td>
<td>10.4</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>$P_{\text{detection}}$ 60.9</td>
<td>64.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$P_{\text{frigate1}}$ 20.3</td>
<td>27.9</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$P_{\text{frigate2}}$ 12.8</td>
<td>18.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$P_{\text{helo}}$ 27.9</td>
<td>18.5</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>$P_{\text{detection}}$ 53.1</td>
<td>43.8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$P_{\text{frigate}}$ 20.8</td>
<td>22.9</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$P_{\text{helo1}}$ 15.1</td>
<td>10.4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$P_{\text{helo2}}$ 17.2</td>
<td>10.4</td>
<td></td>
</tr>
</tbody>
</table>

related to differences in the modelling. In MANA this reaction appears to be more efficient, at least for the scenarios in this thesis.

Despite the difference, MANA and the UWT still show a concurring trend in the global output. Evading results in a significantly better probability of success for the SSK. Besides, defence 3 is significantly better than 2, which is in turn better than defence strategy 1.

5.3 Analysis of the scenarios through MANA

For a good analysis of a scenario, without a dependence between approach and defence, the scenarios are simulated with 7500 random initial positions for the SSK. Figure 4.7 gave a survey of the distribution of these positions. The MoEs are summarised in Table 5.5 for all approaches and defences. All MoEs are given with their 95%-CI. The analysis of the kamikaze approach against all defence strategies is also visualised with graphics of the initial positions of the SSK with their outcome. The graphical analyses of the other approaches are given in Appendix C.

The goal of a simulation is to make an assessment of a certain tactic or concept. For this reason, the MoE from a simulation needs to be interpreted correctly. In this thesis the interpretation is as follows:

The MoE is the proportion of submarine approaches that would be a threat without a defence, but are no threat with this concept or strategy.

With this translation, the MoE is the reduction of the risk for a HVU.

When the kamikaze approach is used, the MoE is the baseline for a tactic. It is the absolute proportion of covered area with this tactic. The MoE of the other two approaches state how well the covered proportion is against evasive submarines. Together with the baseline, these MoEs provides the effectiveness of a tactic or concept.
Table 5.5: MoE with 95%-CI for all scenarios, with 7500 random initial positions for SSK, simulated with MANA

<table>
<thead>
<tr>
<th>Approach</th>
<th>MoE</th>
<th>Defence 1 [%]</th>
<th>Defence 2 [%]</th>
<th>Defence 3 [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kamikaze</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P_{detection}$</td>
<td>54.5 ± 1.1</td>
<td>80.8 ± 0.9</td>
<td>75.9 ± 1.0</td>
<td></td>
</tr>
<tr>
<td>$P_{frigate1}$</td>
<td>28.2 ± 1.0</td>
<td>26.9 ± 1.0</td>
<td>23.8 ± 1.0</td>
<td></td>
</tr>
<tr>
<td>$P_{frigate2}$</td>
<td>–</td>
<td>27.2 ± 1.0</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>$P_{helo1}$</td>
<td>26.3 ± 1.0</td>
<td>26.7 ± 1.0</td>
<td>26.3 ± 1.0</td>
<td></td>
</tr>
<tr>
<td>$P_{helo2}$</td>
<td>–</td>
<td>–</td>
<td>25.8 ± 1.0</td>
<td></td>
</tr>
<tr>
<td>Semi-kamikaze</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P_{detection}$</td>
<td>42.0 ± 1.1</td>
<td>74.2 ± 1.0</td>
<td>64.9 ± 1.1</td>
<td></td>
</tr>
<tr>
<td>$P_{frigate1}$</td>
<td>22.8 ± 1.0</td>
<td>28.0 ± 1.0</td>
<td>28.3 ± 1.0</td>
<td></td>
</tr>
<tr>
<td>$P_{frigate2}$</td>
<td>–</td>
<td>22.9 ± 1.0</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>$P_{helo1}$</td>
<td>19.2 ± 0.9</td>
<td>23.3 ± 1.0</td>
<td>18.7 ± 0.9</td>
<td></td>
</tr>
<tr>
<td>$P_{helo2}$</td>
<td>–</td>
<td>–</td>
<td>17.9 ± 0.9</td>
<td></td>
</tr>
<tr>
<td>Cautious</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P_{detection}$</td>
<td>32.2 ± 1.1</td>
<td>61.1 ± 1.1</td>
<td>58.0 ± 1.1</td>
<td></td>
</tr>
<tr>
<td>$P_{frigate1}$</td>
<td>13.3 ± 0.8</td>
<td>17.9 ± 0.9</td>
<td>17.7 ± 0.9</td>
<td></td>
</tr>
<tr>
<td>$P_{frigate2}$</td>
<td>–</td>
<td>12.6 ± 0.8</td>
<td>–</td>
<td></td>
</tr>
<tr>
<td>$P_{helo1}$</td>
<td>18.9 ± 0.9</td>
<td>30.6 ± 1.0</td>
<td>18.8 ± 0.9</td>
<td></td>
</tr>
<tr>
<td>$P_{helo2}$</td>
<td>–</td>
<td>–</td>
<td>21.5 ± 0.9</td>
<td></td>
</tr>
</tbody>
</table>

When analysing Table 5.5, there is a difference between a statistical significance and the interpretation. The difference in probability of detecting an SSK can be statistically significant, but for operational purposes they can be the same. For example, when a commander wants a probability of at least 55%, it does not matter whether this is 58.0% or 61.1%. Therefore it is up to a client or operator to interpret the output. Keeping this in mind, the following can be said about the output from MANA:

- Following an reactive approach is always better for a submarine. Both the semi-kamikaze and the cautious approach give the SSK a significantly better change for a successful torpedo attack than a kamikaze approach.
- Adding a third unit to the defence significantly increases the probability of detection. It does not matter whether this is a MFF or an NH-90. The addition of a third unit becomes especially effective against evasive submarines. The extra frigate in defence 2, for example, increases the probability of detection against a kamikaze approach with a factor 1.48, against a semi-kamikaze approach with a factor 1.65 and against a cautious approach with a factor 1.90.
- It seems odd that frigate 1 performs better in defence 2 and 3 against a semi-kamikaze approach than against a kamikaze approach. This phenomena is also visible with helicopter 1 in defence 2 against a cautious SSK. This peculiar increase is due to the fact that other ASW units force the SSK towards this unit. When the SSK tries to react on a defence units, it is caught by the following unit. This occurs only with three defence units as the gaps between two units is smaller in this case.
- It seems that an NH-90 performs worse than the MFF as a defence unit against a kamikaze or semi-kamikaze approach. However, against a cautious approach, the NH-90 performs significantly better. This is because the SSK cannot detect a helicopter in flight, while it can detect and track a frigate all the times. Therefore the evading actions are less effective against helicopters.

Figures 5.2, 5.3 and 5.4 show a more detailed analysis of the kamikaze approach.
against all defence strategies. Figures 5.2a, 5.3a and 5.4a show the initial positions of the submarines that were detected, and by which ASW-unit. Figures 5.2b, 5.3b and 5.4b show the initial location of those submarines that came within the METFR of the HVU without being detected. Together these figures show all 7500 randomly simulated initial positions in a simulation batch.

When analysing the first set of figures in Figure 5.2, first and foremost it is apparent that the defence is symmetrical; there are no weak or strong zones in the strategy. However, it is notable that the NH-90 never detects an SSK that approaches right through the middle of a leg. This is because the NH-90 only uses its HELRAS at a vertex, and the range of the HELRAS appears to be slightly less than half the distance of one leg. So there are some weak points in the defence. Furthermore, the initial positions that are detected spiral outwards, due to the symmetrical defence. This is because when the SSK starts from a direction a couple of degrees further clockwise, and further away, it encounters the same locations of the defence units.

Additionally, the size of a group of positions from which the SSK was undetected states something about the ease with which the SSK could penetrate. Since the band of black positions in Figure 5.2b is wide, the SSK has room and time both to manoeuvre and to make small errors. If this band were thin, the SSK would have little room for error and must penetrate perfectly.
Figure 5.2: Initial positions of the SSK, against the first defence strategy, following a kamikaze approach, simulated in MANA.
(a) Positions which are positively detected by which unit.

(b) Positions for successful, unclassified submarines and their torpedo launch positions.

Figure 5.3: Initial positions of the SSK, against the second defence strategy, following a kamikaze approach, simulated in MANA.
Figure 5.4: Initial positions of the SSK, against the third defence strategy, following a kamikaze approach, simulated in MANA.
When analysing all the visual output for the kamikaze approach it is notable that the number of detected SSKs rises significantly when a third unit is deployed, which concurs with Table 5.5. In both Figure 5.3b and 5.4b, the number of black positions is less than in Figure 5.2b. Furthermore, the band of these positions is clearly less wide, which means that the SSK has less room for error in its approach.

For the choice of the third unit, it would seem that an extra frigate has more impact than an extra helicopter for the kamikaze approach, which was also stated in the analysis of Table 5.5. This is remarkable, as the helicopter is seen as one of the most effective ASW-units, which includes detecting a submarine. This outcome is due to the fact that the behaviour of the NH-90 is adjusted to the MFF, in order to remain a symmetrical defence. Key qualities of the helicopter, such as speed and manoeuvrability, are not used to their full potential this way. But the disadvantage that the HELRAS is not always active, is incorporated in the model. Only against cautious submarines, the strength of the NH-90 over the MFF became visible in Table 5.5: A helicopter in flight is undetectable to a submarine.

5.4 Sensitivity analysis through MANA

It is important that the analysis of a ship concept or tactic is not to limited to a small set of situations. Therefore different threat behaviours were taken into account for each defence scenario. But the impact of a single parameter could also affect the output. Therefore a sensitivity analysis will be executed to analyse the effect of two parameters: the radius of the inner circle of the octagon and the speed of the frigate. The effect of these parameters is tested in two scenarios: a kamikaze and a cautious approach against defence strategy 1.

5.4.1 Influence of the radius of the defence octagon

The radius of the defence octagon has an influence on the output. When this parameter is decreased, the defending units are closer to each other, and the gaps between them are smaller. This will increase the probability of classifying an SSK. However, when this parameter is set too small, there is not enough time for prosecution of the SSK, as the SSK is closer to the HVU and might launch its torpedo before it gets prosecuted effectively. Figure 5.5 shows the sensitivity of the radius. In this figure, the radius of the inner circle of the octagon is set against the probability of classifying the SSK. This figure does not take the time needed for prosecution into account, therefore the probability generally increases with a decreasing radius.

The data shown in Figure 5.5 shows the effect when a wrong diameter is set in the scenario. For example, when someone sets distances in yards instead of meters or vice versa. The small increase in the probability at 11 km is due to the fact that at smaller distances, the radius comes close to the METFR. The result is that the SSK does not need to penetrate the whole screen to enter the TDZ.

5.4.2 Influence of the speed of the frigates

Another parameter which affects a scenario, is the speed of the frigate. This parameter determines how quickly a full round is sailed. Furthermore, this parameter affects the dipping time of the helicopter’s HELRAS in this scenario: an increased speed of the frigate means a shorter dipping time. In real operations, the speed of the frigate also affects the performance of the towed sonar. This effect is not taken into account, and the range of the sonar is kept constant for all speeds. Figure 5.6 shows the effect of the speed on the probability of classifying the SSK.
Figure 5.5: sensitivity of the radius of the octagon

Figure 5.6: sensitivity of the speed of the frigate
The analysis in Figure 5.6 shows an increase in the probability of detecting the SSK when the speed increases. However, as stated above, the speed shown in this figure has no effect on the sensor parameters. When the dependence between the speed and the sensor performance is known, this analysis could optimise the speed in a tactic or concept.

5.5 Qualitative comparison between MANA and UWT

Besides the quantitative comparison between the results of a simulation in MANA and UWT, as is given in Section 5.2, the qualitative properties of the two different programs are also compared. The comparison will focus on three aspects: the level of insights the results can give in new tactics or concepts, the possibility of implementing new units and concepts, and finally the adaptability to changing situations. The results of this comparison originate from the users of MANA and UWT.

5.5.1 Insights in scenarios

The numeric and graphical results, discussed in Sections 5.3 and 5.4, show the different insights MANA can give in a scenario. Especially the figures with the initial positions of the threats give a clear visualisation of the strengths and weaknesses of a scenario or tactic. Combined with the sensitivity analysis for critical parameters, MANA (with Matlab) can give a good insight in a concept or tactic.

However, the output MANA can generate, does not differ from the output of the UWT. The output in the UWT can be analysed and visualised the same way. Only the time the UWT would take for a sensitivity analysis as presented in this research, is enduring compared to the MANA/Matlab simulation framework, because MANA is significantly faster with random initial positions for the SSK. Therefore the UWT is less suitable for a thorough sensitivity analysis.

5.5.2 Implementation of new concepts

Both models have the ability to implement new units, otherwise the scenarios could not have been built for the simulations. However, the way in which units are modelled is completely different in MANA and the UWT.

MANA offers a user-friendly interface to model the agents. After a quick introduction the user can easily understand the interface, and set behavioural actions and reactions. Furthermore, a user can check settings in a transferred model rather easily to see how an agent is modelled. However, this accessibility comes with a price: the behavioural aspects which can be modelled are limited. For more complex behaviour than the cautious approach, MANA is inadequate and has insufficient means for modelling behaviour.

In contrast, the UWT is a complex tool. Settings and behavioural are numerous and are coded within the script of a model. This has the benefit that there is usually a way to model a new behavioural reaction. However, thorough knowledge of the programme is necessary to accomplish this. Furthermore, just checking the settings by someone else is not as easy, as some parameters are set in the script.

5.5.3 Adaptability

The final qualitative aspect is the possibility to adapt the model when operational conditions or concepts change.

In the MANA/Matlab framework, parameters such as sensors, speed and weapons can be changed easily, both in MANA itself as through Matlab. For example, the
sensor ranges can be linked to a table with operational and environmental properties. This way, the detection and classification ranges can be linked to changing conditions. In the UWT, these kind of parameters can be changed easily, if the user has the knowledge where in the programme the parameters are defined.

For behavioural changes, the same comments apply as were stated in Subsection 5.5.2. New behaviour in MANA is quickly limited and new behaviour in the UWT must be coded in the script.
6 Limitations in MANA

This chapter will discuss three models in which the limitations of simulating with MANA become visible. All models are variations of the sea-base scenario from Chapter 3. First, the endurance of a single helicopter is limited. Secondly, false sonar contacts are added. Finally, a coordination between ASW-units is added.

6.1 Helicopter flight time limitations

In the Holon-base scenarios, it was assumed that the task-force had enough helicopters to keep one in the air at all times. However, this is not realistic in most naval operations. The time a helicopter can be used in an operation is limited by its fuel and its crew’s endurance, and often the number of helicopters is not sufficient to keep one airborne at all times. To make a simulation more realistic, the endurance of the helicopter should be limited and the helicopter must return to the frigate from which it has set off.

In MANA this can be accomplished by embussing the NH-90 on the MFF. This means that the helicopter is stationed at its parent until a trigger releases it. Another trigger can make the NH-90 embuss again at the frigate. Table 6.1 shows the modelling of the frigate to simulate the limited endurance and maintenance for the NH-90. This is on top of the settings from the base scenario. The endurance of the NH-90 in seconds can be set by the initial fuel of the frigate. This seems contradictory, but MANA offers no trigger for return to parent, but only for embuss children. Therefore the endurance of the helicopter must be monitored at the frigate. The maintenance duration is set by the duration of the Refuelled by anyone state at the frigate.

<table>
<thead>
<tr>
<th>State</th>
<th>Meaning</th>
<th>Fuel usage [-/s]</th>
<th>Embussing Behaviour</th>
<th>Next state</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default</td>
<td>Sailing without helicopter</td>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Run Start</td>
<td>Release NH-90</td>
<td>1</td>
<td>Release child squads</td>
<td>Default</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fuel out</td>
<td>Call back NH-90</td>
<td>0</td>
<td>Embuss children</td>
<td>Default</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Refuelled by anyone</td>
<td>NH-90 back is</td>
<td>0</td>
<td>-</td>
<td>Spare 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spare 1</td>
<td>Release NH-90</td>
<td>1</td>
<td>Release child squads</td>
<td>Default</td>
</tr>
</tbody>
</table>

In the model of the NH-90 the state must embuss is added. This state gives the command to return to the frigate, when the helicopter gets called back. On arrival
at the frigate, the helicopter will refuel the frigate in order to reset the timer for the endurance.

The drawback of modelling the flight time limitations this way is that it is not possible to keep the helicopter opposite of the frigate in the octagon. The helicopter flies to its next waypoint, which is the waypoint after the point from which it flew to the frigate. However, the frigate keeps sailing during the time the helicopter receives its maintenance. Therefore the formation is not maintained. This way of modelling would therefore be more suited for defence strategies where each unit has its own zone, and it does not matter what the relative location of the units is.

Also it is not possible to embuss multiple helicopters on one ship and let them alternate their flight hours. In MANA, a parent can release all children or none. Releasing one child and holding the other is not possible. It could be possible to embuss multiple helicopters on as many ships. However, coordinating their flight time would require some out-of-the-box solutions as will be explained in Section 6.3.

6.2 False contact simulation

The ASW-units in the base scenarios did not react on unknown contacts. This was modelled because no false contacts were added in the simulation. Adding those to the simulation could make the model more useful. This way, certain reactions to unknown contacts within different tactics could be tested, or the improvement of a ship concept can be tested when the number of false contacts is diminished. In MANA, an unknown contact is an agent which is detected, but not classified. With cookie-cutter sensors, an unknown lies within the detection range, but outside the classification range.

A way to simulate false contacts is by adding a squad of $N$ enemy agents. These units can be stationary or moving, but they do not react on the SSK or ASW-units. By increasing the number of agents in the squad, the density of false contacts can be simulated. With this modelling, reactions to unknown contacts (whether false or SSK) can be examined for different densities. This information can be used for testing ship or sonar concepts and the number of false contacts they generate.

To make sure the false contacts do not make the ASW-unit react after it sails away from them, more tricks needs to be inserted. It can be resolved by shooting the false contacts after classification, without killing them. When the false contacts are shot, their class and threat level change, and they cannot trigger anymore ASW-units for $x$ seconds.

6.3 Coordinate deployment of resources

In the models so far, all behaviour and actions of units are solely based on their own SA and decisions. In real operations, units often act coordinated. For example, in prosecuting a classified submarine, one helicopter can use its HELRAS to track the submarine and another ASW-unit can attack it. MANA, however, offers no ready model blocks for coordination between agents. Only SA can be shared, but no commands can be given.

The only way to solve this and simulate a sort of coordination between two units is by letting one agent shoot the other agent, without killing him. The agent who is shot can react to this trigger with a certain set behaviour. This way of coordination is a 1/0 command; _Do the set command now_!. However, an agent will not
shoot a classified friend, but they can shoot an unknown who might be a friendly. So the agents will need a sensor with a detection range to see its friend, but a classification range that cannot classify him as such. This way, an NH-90 can give a frigate a command to prosecute a classified enemy submarine when the helicopter detects one.

Though this way commands can be given, combining this coordination with the false contacts in the previous subsection requires even more tricks. For example, a blue force shoots an enemy, which shoots an other blue force agent to give commands. Although this creates some possibilities, it is not a clean way of modelling a scenario.
7 Conclusion and recommendations

The goal of this thesis is twofold. First, different models in MANA were built for ASW sea-basing scenarios. Second, the modelling in MANA is compared with the modelling in the TNO tool UWT, in order to examine the usefulness of MANA for the Holon-project; is MANA useful for testing ship concepts in an early stage of the design process.

To this end nine scenarios are defined, with three defence strategies and three different levels of behaviour for the SSK. These levels consist of an approach with no behaviour, an approach with simple behaviour, and an approach with a more advanced behaviour.

The analysis of the simulation models of these scenarios has resulted in the following conclusions:

- The simulation framework designed in this thesis, containing MANA and MATLAB, is capable of preparing the ASW scenarios, simulating the runs and analysing the output. Furthermore, graphical output is able to give a more detailed insight into the effectiveness of a concept or tactic.
- For the scenario with the most simple behaviour of the SSK, an analytical model is created. The input-output relations of this analytical model correspond to the MANA model. The implementation of the simple models is therefore validated.
- For the quantitative comparison, the output generated with MANA is in line with the output of the more complex UWT programme of TNO. For scenarios with no behaviour for the SSK the results are similar. When simple forms of behaviour are modelled for the SSK, the results show a comparable trend. When more complex behaviour is modelled, the difference in modelling is clearly evident in a difference in the output. This output, however, still shows a comparable trend.
- On the qualitative side, MANA is a suitable tool for modelling tactics and concepts with a low to medium level of complexity. The user interface is accessible and easy to comprehend. However, when the settings or behaviour become more complex, certain tricks are necessary in order to model some features. At a certain point, MANA becomes inadequate and offers insufficient possibilities. For those models, a tool with more features is required.

With these conclusions, it can be stated that MANA is useful as a simulating tool for the Holon-project. Especially for testing concepts in an early state for parameters such as endurance, speed, sensor characteristics or detection counter-detection trade-off. It can be used as a tool for shifting through multiple concepts and selecting those concepts that seem promising.

When MANA is used in the Holon-project, or other simulation environments, some additional research should be done. Therefore, the following recommendations are made:

- The analyses of the simulations is currently the most time-consuming phase. This is because for a proper analysis of the casualties, at least one single file is produced per run. So a large number of runs produce as many files, which then needs to be opened and read. Therefore, this phase could be optimised greatly by encoding a more efficient way of opening and reading all the output files.
• Contact should be established with DTA in New-Zealand, the originators of MANA, about the tool. This way improvements could be made to break some limitations of MANA. These improvements can be, but are not limited to:
  – Adding the ability for logic operators between trigger states.
  – Add a way to rank the trigger states in a certain priority.
  – Add a possibility for setting different stochastic distributions for parameters such as the duration of a state.
  – Adding the possibility to react on CPAs, and not only on current distances.
• For a more complete use in both testing ship concepts and operational tactics, the prosecution of a classified submarine should be modelled as well.
• When the framework in this thesis is used for operational purposes, a connection could be made to environmental and operational parameters, such as sensor characteristics. Furthermore, real defence strategies should be implemented beforehand so an operator can simply choose one. When the programme is used this way, operators and analysts should take into account that the output of the model is not impeccable. Especially when behaviour is modelled, an adversary could always follow a different type of behaviour. However, the model is well suited to give solid insights into the used tactics.
8 References

A Modelling the sensors

As was discussed in this thesis, one of the strengths of MANA are the blocks where sensor, weapon and communication systems can be modelled. This appendix focusses on the sensor block in more detail, since this block is most used in the scenarios in this thesis.

A.1 Sensor to model

(Military) sensors, such as radars, sonars, and even the old eyeball Mark-I, are complicated real-world systems. It is hard to predict when a sensor will detect, classify or possibly identify a target, especially since this depends on particular target and environmental conditions. To limit the number of calculations and the time needed, there are simplified sensor models.

The most simple model is the cookie-cutter model [17]. This is a 1/0 model: all contacts within the maximum range are detected and all contacts outside this range are invisible. An example of a cookie-cutter sensor is shown in Figure A.1. In this figure everything within 10 nm is detected with probability 1 and everything outside with probability 0. The maximum range is calculated for a particular environment and target.

![Figure A.1: An example of a cookie-cutter sensor](image)

In a more advanced model the probability of detecting a contact is dependent on the distance. An example of this type of sensor is given in Figure A.2. When the target comes closer to the sensor, the probability increases. The values for a probability at a distance are also based on a particular environment and target.

A.2 MANA sensor model settings

In the MANA sensor block, both of the models described above can be implemented. In the simple sensor block, a cookie-cutter sensor can be implemented. For both the detection and the classification, a maximum range can be defined.

In the advanced sensor block, a distance-dependent table can be implemented. For the detection, the average time between detections can be given at different distances. For the classification, a probability per time-step $\Delta t$ can be given at different distances. For the value of a distance in between two defined distances, MANA interpolates the values at the defined distances.
Figure A.2: An example of a sensor with a distance-dependent probability

To simulate changing environmental conditions, the values of a sensor can be changed per state of the agent. For example, the detection-range can be decreased when an agent reaches a waypoint, simulating deteriorating environmental conditions. Furthermore, to simulate changing target conditions, the advanced model offers a list of classes that a particular sensor can see. This way, an enemy agent can change his state, and thus class, after a certain trigger. This, a diving submarine could be simulated, for example, to which the sonars have a different range. Figure A.3 displays the control panel in MANA for the advanced sensor model. Visible are the sensor ranges, the detectable target classes and the number of active sensors. The values in the figure are from the NH-90 at the dip in this thesis.

Figure A.3: The MANA advanced sensor control panel
B Derivation of the equations

This appendix will derive the equations given in Chapter 4.

B.1 Probability of detection in the analytical model

As was explained in Section 4.1, the probability of detection can be derived by calculating the part of the swept area in the relative plot. Figure B.1 shows this figure again. In this figure the movements of a frigate and a helicopter relative to the SSK are visualised. The SSK is then motionless anywhere in the grey area with a uniform distribution.

![Diagram showing the analytical model with 1 MFF and 1 NH-90 for a kamikaze approach](image)

The probability of detection is therefore the area of the green surfaces divided by the grey area. The areas of the green surfaces will be derived per unit in the next subsections.

B.1.1 Probability for frigate

The grey area can be calculated with the following equation, where $V_f$ the speed of the frigate is and $U$ the speed of the submarine.

$$A_{\text{grey}} = L \cdot B = L \cdot L \cdot \tan(\alpha) = L^2 \cdot \frac{U}{V_f}$$

(B.1)

The area of the green surface swept by the frigate is the area of the rectangle, minus four triangles. The triangles are magnified in Figure B.2. The areas of triangles I and II are calculated with:

$$A_I = 0.5 \cdot \frac{R_f^2}{\tan(\alpha)}$$

$$A_{II} = 0.5 \cdot R_f^2 \cdot \tan(\alpha)$$
Here is $R_f$ the classification range of the frigate.

Using these equations, the swept area of the frigate is calculated with the following equation:

\[
A_{\text{frigate}} = 2R_f \cdot \sqrt{L^2 + B^2 - R_f^2 \tan(\alpha)} - \frac{R_f^2 \tan(\alpha)}{
\tan(\alpha)} \\
= 2R_f \cdot \sqrt{L^2 + (L \cdot \frac{U}{V_f})^2 - R_f^2 (\tan(\alpha) + \frac{1}{\tan(\alpha)})} \\
= 2R_f L \cdot \sqrt{1 + \left(\frac{U}{V_f}\right)^2 - R_f^2 \left(\frac{U}{V_f} + \frac{V_f}{U}\right)}
\]

(B.2)

The last restriction is added because the correction for the triangles is only correct when the classification range is small comparing to the length of $L$.

**B.1.2 Probability for helicopter**

Figure B.3 shows the relative movement of a helicopter at one dipping position. Here is $l$ the distance between two dipping positions.

The grey area around eight dipping positions is equal to the grey area derived for the frigate, because the movement of the helicopter is kept dependent of the movement of the frigate. For an independent movement, however, the grey area around one dipping position is equal to $A_{\text{dip}}^{grey} = L \cdot b$. Here $b$ is calculated using
the following equation, where $t_{dip}$ the active dip time of the sonar is and $t_{set}$ the setting time and recovering time for the HELRAS dipping sonar and $V_h$ the speed of the helicopter.

$$b = (t_{set} + t_{dip}) \cdot U + l \cdot \tan(\beta)$$

$$\tan(\beta) = \frac{U}{V_h}$$

Therefore:

$$A_{grey}^{dip} = L \cdot U \cdot \left( \frac{l}{V_h} + t_{dip} + t_{set} \right)$$

The swept area for one dipping position is the sum of the green rectangle and two semicircles.

$$A_{helicopter} = 2 \cdot R_h \cdot U \cdot t_{dip} + \pi \cdot R_h^2$$

Combining Equation B.5 with Equation B.4, the probability of detection for a helicopter is equal to:

$$P_h = \frac{2 \cdot R_h \cdot U \cdot t_{dip} + \pi \cdot R_h^2}{L \cdot U \cdot \left( \frac{l}{V_h} + t_{dip} + t_{set} \right)}, \text{ if } R_h \geq 0.5 \cdot l \text{ and } L >> l$$

The condition $R_h \geq 0.5 \cdot l$ is added because the swept area of the different dipping positions may not have any overlap. Furthermore, the number of dipping positions on barrier front $L$ must be great enough, in order to neglect small parts of the swept area that lay outside the grey area, so $L >> l$.

### B.2 Aperture setting for SSK

The aperture of the sensor for the SSK in the semi-kamikaze approach is calculated so, that it will detect an ASW-unit with which it has an CPA (Closest Point of Approach) less than the classification range of that ASW-unit (with a safety factor $\alpha \geq 1$). For this calculation, the relative speed between the ASW-unit and the SSK is used. Assumed is that the ASW-unit moves perpendicular to the SSK, as is shown in Figure B.4.
When the SSK wants to keep a distance \( \alpha \cdot R_f \) clear of the ASW-units, with \( R_f \) as the classification range of an ASW-unit, the sensor aperture is visualised in Figure B.5. This model is similar to the way Limiting Lines of Approach are calculated for ASW screens [3]. In this model a safety zone is put around the SSK, were no ASW-units should be. The relative speed an ASW-unit could have are placed tangent to this safety zone. Thereupon, the SSK should only react on ASW-units within the blue sensor limits in Figure B.5.

However, it is not possible in MANA to model this sensor aperture. Therefore a simpler approximation is used. Figure B.6 shows the model to calculate the simpler aperture. In this model, the relative motion between the SSK and the ASW-unit is placed so that is it tangent to the detection circle around the MFF. The aperture for the SSK \( 2 \cdot \beta \) is then equal to \( \beta = \gamma + \zeta \). Here is \( \gamma \) the angle between the relative and absolute motion of the SSK and \( 2 \cdot \zeta \) the angle between the relative and absolute motion of the MFF. With \( \gamma + 2 \cdot \zeta = 90^\circ \). So:

\[
\beta = \arctan\left(\frac{V_f}{U}\right) + 0.5 \cdot \arctan\left(\frac{U}{V_f}\right) \tag{B.7}
\]

With \( V_f \) is the speed of the frigate and \( U \) the speed of the SSK.
Figure B.6: Approximation model for aperture of SSK in the semi-kamikaze approach
C Extended output of the base scenarios in MANA

This appendix reproduces the settings and graphical output of all base scenarios, as they were simulated in MANA.

C.1 Settings

The settings in all simulations were as followed:
\[ METFR = 9 \ [\text{km}] \]
\[ V_{\text{frigate}} = 12 \ [\text{kts}] \]
\[ V_{\text{helo}} = 100 \ [\text{kts}] \]
\[ V_{\text{submarine}} = 6 \ [\text{kts}] \]
\[ R_{\text{LFAS}} = 9.144 \ [\text{km}] \]
\[ R_{\text{HELAS}} = 9.144 \ [\text{km}] \]
\[ R_{\text{subcounter-detection}} = 15 \ [\text{km}] \]
\[ R_{\text{octagon-inner-circle}} = 22.86 \ [\text{km}] \]
Seed = 1488

C.2 MoE output

Table C.1: MoE for all scenarios, with 7500 random initial positions for SSK, simulated with MANA

<table>
<thead>
<tr>
<th>Approach</th>
<th>MoE</th>
<th>Defence 1 [%]</th>
<th>Defence 2 [%]</th>
<th>Defence 3 [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kamikaze</td>
<td>( P_{\text{detection}} )</td>
<td>54.5 ± 1.1</td>
<td>80.8 ± 0.9</td>
<td>75.9 ± 1.0</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{frigate1}} )</td>
<td>28.2 ± 1.0</td>
<td>26.9 ± 1.0</td>
<td>23.8 ± 1.0</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{frigate2}} )</td>
<td>–</td>
<td>27.2 ± 1.0</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{helo1}} )</td>
<td>26.3 ± 1.0</td>
<td>26.7 ± 1.0</td>
<td>26.3 ± 1.0</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{helo2}} )</td>
<td>–</td>
<td>–</td>
<td>25.8 ± 1.0</td>
</tr>
<tr>
<td>Semi-kamikaze</td>
<td>( P_{\text{detection}} )</td>
<td>42.0 ± 1.1</td>
<td>74.2 ± 1.0</td>
<td>64.9 ± 1.1</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{frigate1}} )</td>
<td>22.8 ± 1.0</td>
<td>28.0 ± 1.0</td>
<td>28.3 ± 1.0</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{frigate2}} )</td>
<td>–</td>
<td>22.9 ± 1.0</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{helo1}} )</td>
<td>19.2 ± 0.9</td>
<td>23.3 ± 1.0</td>
<td>18.7 ± 0.9</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{helo2}} )</td>
<td>–</td>
<td>–</td>
<td>17.9 ± 0.9</td>
</tr>
<tr>
<td>Cautious</td>
<td>( P_{\text{detection}} )</td>
<td>32.2 ± 1.1</td>
<td>61.1 ± 1.1</td>
<td>58.0 ± 1.1</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{frigate1}} )</td>
<td>13.3 ± 0.8</td>
<td>17.9 ± 0.9</td>
<td>17.7 ± 0.9</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{frigate2}} )</td>
<td>–</td>
<td>12.6 ± 0.8</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{helo1}} )</td>
<td>18.9 ± 0.9</td>
<td>30.6 ± 1.0</td>
<td>18.8 ± 0.9</td>
</tr>
<tr>
<td></td>
<td>( P_{\text{helo2}} )</td>
<td>–</td>
<td>–</td>
<td>21.5 ± 0.9</td>
</tr>
</tbody>
</table>
C.3 Graphics of first defence strategy

(a) Positions which are positively detected by which unit.

(b) Positions for successful, unclassified submarines and their torpedo launch positions.

Figure C.1: Initial positions of the SSK, against the first defence strategy, following a kamikaze approach, simulated in MANA.
(a) Positions which are positively detected by which unit.

(b) Positions for successful, unclassified submarines and their torpedo launch positions.

Figure C.2: Initial positions of the SSK, against the first defence strategy, following a semi-kamikaze approach, simulated in MANA.
(a) Positions which are positively detected by which unit.

(b) Positions for successful, unclassified submarines and their torpedo launch positions.

Figure C.3: Initial positions of the SSK, against the first defence strategy, following a cautious approach, simulated in MANA.
C.4 Graphics of second defence strategy

Figure C.4: Initial positions of the SSK, against the second defence strategy, following a kamikaze approach, simulated in MANA.
(a) Positions which are positively detected by which unit.

(b) Positions for successful, unclassified submarines and their torpedo launch positions.

Figure C.5: Initial positions of the SSK, against the second defence strategy, following a semi-kamikaze approach, simulated in MANA.
Figure C.6: Initial positions of the SSK, against the second defence strategy, following a cautious approach, simulated in MANA.
C.5 Graphics of third defence strategy

(a) Positions which are positively detected by which unit.

(b) Positions for successful, unclassified submarines and their torpedo launch positions.

Figure C.7: Initial positions of the SSK, against the third defence strategy, following a kamikaze approach, simulated in MANA.
(a) Positions which are positively detected by which unit.

(b) Positions for successful, unclassified submarines and their torpedo launch positions.

Figure C.8: Initial positions of the SSK, against the third defence strategy, following a semi-kamikaze approach, simulated in MANA.
Figure C.9: Initial positions of the SSK, against the third defence strategy, following a cautious approach, simulated in MANA.

(a) Positions which are positively detected by which unit.

(b) Positions for successful, unclassified submarines and their torpedo launch positions.
D The MANA base models

This appendix will give an outline of all settings in the MANA base models of this thesis.

D.1 Global settings

The battlefield in MANA is a map of 250 by 250 km. The origin of the field lies in the center. There are no hiding places or restrictions for movement. Table D.1 gives an overview of all agents in the base scenarios. The next three subsections give a more detailed view of the modelling of the agents.

Table D.1: Global settings for base scenarios in MANA

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Description</th>
<th>Agent class</th>
<th>agent</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base scenario</td>
<td>Basis scenarios from HOLON.</td>
<td>1</td>
<td>NH90</td>
<td>NH90 in flight</td>
</tr>
<tr>
<td></td>
<td>1 HVU, 1 SSK, 1/2 MFFs, 1/2 NH90s</td>
<td>2</td>
<td>HYU</td>
<td>HYU</td>
</tr>
<tr>
<td></td>
<td>3 NH90</td>
<td>3</td>
<td>NH90</td>
<td>NH90 in active dip</td>
</tr>
<tr>
<td></td>
<td>4 MFF</td>
<td>4</td>
<td>MFF</td>
<td>MFF met LFA5</td>
</tr>
<tr>
<td></td>
<td>8 SSK</td>
<td>8</td>
<td>SSK</td>
<td>SSK sailing</td>
</tr>
<tr>
<td></td>
<td>9 SSK</td>
<td>9</td>
<td>SSK</td>
<td>SSK silent</td>
</tr>
</tbody>
</table>

For the simulation of the models in MANA, a model time step of 5 seconds is used. Furthermore the data outputs of a simulation batch are Casualty location data and Positions data.

D.2 Settings for the frigate

The frigates only sail to the next waypoint. Their behaviour is therefore straightforward. Table D.2 gives a survey of the modelling of the frigates. The movement vector is only determined by the next waypoint.

Table D.2: Frigate modelling in the base scenarios in MANA

<table>
<thead>
<tr>
<th>Overview</th>
<th>State</th>
<th>Phase</th>
<th>next state</th>
<th>Class</th>
<th>Threat</th>
<th>Duration [s]</th>
<th>Speed [kts]</th>
<th>R_sensor [m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>default</td>
<td>Sail to next waypoint</td>
<td>-</td>
<td>4</td>
<td>3</td>
<td>-</td>
<td>12</td>
<td>9544</td>
<td></td>
</tr>
</tbody>
</table>

Sensors

<table>
<thead>
<tr>
<th>Sensor #</th>
<th>Target</th>
<th>target classes</th>
<th>Arc</th>
<th>Offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>sailing SSK</td>
<td>all</td>
<td>360</td>
<td>0</td>
</tr>
</tbody>
</table>

Weapons

<table>
<thead>
<tr>
<th>weapon #</th>
<th>Range</th>
<th>fire rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20000</td>
<td>1</td>
</tr>
</tbody>
</table>

D.3 Settings for the helicopter

The helicopters fly from waypoint to waypoint and use their sonar at the waypoints. To simulate this, a cycle of 4 states is used. Table D.3 gives a survey of the modelling of the helicopters. The movement vector is only determined by the next waypoint.
D.4 Settings for the submarine

The submarine has three ways to approach the sea-base. Each approach is modelled differently. For the kamikaze approach, the submarine follows a straight line to the center of the sea-base. The modelling of this is given in Table D.4. The movement is determined by the next waypoint (center of the sea-base) and the location of the HVU (when classified).

<table>
<thead>
<tr>
<th>State</th>
<th>Type</th>
<th>next state</th>
<th>Class</th>
<th>Threat</th>
<th>Duration [s]</th>
<th>Speed [kts]</th>
<th>R_sensor1 [m]</th>
<th>R_sensor2 [m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>default</td>
<td>Fly to next waypoint</td>
<td></td>
<td>1</td>
<td>2</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Reach waypoint</td>
<td>Search using HELAS</td>
<td>Spare1</td>
<td>1</td>
<td>2</td>
<td>0.6 * 60</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Spare1</td>
<td>Search using HELAS</td>
<td>Spares2</td>
<td>3</td>
<td>2</td>
<td>Tdp</td>
<td>0</td>
<td>9144</td>
<td>7600</td>
</tr>
<tr>
<td>Spare2</td>
<td>Recover HELAS</td>
<td>default</td>
<td>1</td>
<td>2</td>
<td>0.6 * 60</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Table D.4: Kamikaze submarine modelling in the base scenarios in MANA**

The modelling of the semi-kamikaze approach is given in Table D.5. In this approach the submarine can respond to ASW-units by stopping its motion. The movement is only determined by the next waypoint and the location of the HVU (when classified), but the behaviour triggers can be set by the frigates or the helicopters.

<table>
<thead>
<tr>
<th>State</th>
<th>Type</th>
<th>next state</th>
<th>Class</th>
<th>Threat</th>
<th>Duration [s]</th>
<th>Speed [kts]</th>
<th>R_sensor1 [m]</th>
<th>R_sensor2 [m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>default</td>
<td>Sail to center sea-base eq. HVU</td>
<td></td>
<td>8</td>
<td>3</td>
<td>6</td>
<td>9</td>
<td>9000</td>
<td></td>
</tr>
</tbody>
</table>

**Table D.5: Semi-kamikaze submarine modelling in the base scenarios in MANA**
Table D.5: Semi-kamikaze submarine modelling in the base scenarios in MANA

<table>
<thead>
<tr>
<th>State</th>
<th>Base</th>
<th>Next state</th>
<th>Class</th>
<th>Threat</th>
<th>Duration [s]</th>
<th>Speed [kts]</th>
<th>R sensor1 [m]</th>
<th>R sensor2 [m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>default</td>
<td>sail to center sea-base eq. HVU</td>
<td>-</td>
<td>8</td>
<td>3</td>
<td>-</td>
<td>6</td>
<td>11990</td>
<td>9000</td>
</tr>
<tr>
<td>Enemy contact 2</td>
<td>Detect NH90</td>
<td>default</td>
<td>8</td>
<td>3</td>
<td>30</td>
<td>6</td>
<td>11990</td>
<td>9000</td>
</tr>
<tr>
<td>Enemy contact 3</td>
<td>Detect MFF</td>
<td>default</td>
<td>8</td>
<td>3</td>
<td>30</td>
<td>6</td>
<td>11990</td>
<td>9000</td>
</tr>
</tbody>
</table>

Sensors

<table>
<thead>
<tr>
<th>Sensor #</th>
<th>Target</th>
<th>Target classes</th>
<th>Arc offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ASW units</td>
<td>3,4</td>
<td>153</td>
</tr>
<tr>
<td>2</td>
<td>HVU</td>
<td>2</td>
<td>300</td>
</tr>
</tbody>
</table>

Weapons

<table>
<thead>
<tr>
<th>weapon #</th>
<th>Range</th>
<th>Hit rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9000</td>
<td>1</td>
</tr>
</tbody>
</table>

COMS

<table>
<thead>
<tr>
<th>COM #</th>
<th>to Squad</th>
<th>info</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table D.6 shows the modelling of a cautious approach for the submarine. In this model the submarine reacts on all enemy agents with a different reaction. Its movement is defined by classified ASW-units, the next waypoint and an attempt to approach an frigate aft. The behavioural triggers are set by classified ASW-units.

Table D.6: Cautious submarine modelling in the base scenarios in MANA

<table>
<thead>
<tr>
<th>State</th>
<th>Base</th>
<th>Next state</th>
<th>Class</th>
<th>Threat</th>
<th>Duration [s]</th>
<th>Speed [kts]</th>
<th>R sensor1 [m]</th>
<th>R sensor2 [m]</th>
<th>R sensor3 [m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>default</td>
<td>sail to center sea-base</td>
<td>-</td>
<td>8</td>
<td>2</td>
<td>6</td>
<td>15000</td>
<td>12000</td>
<td>11990</td>
<td></td>
</tr>
<tr>
<td>Enemy contact 1</td>
<td>detect HVU -&gt; target</td>
<td>default</td>
<td>8</td>
<td>3</td>
<td>10</td>
<td>6</td>
<td>15000</td>
<td>12000</td>
<td>11990</td>
</tr>
<tr>
<td>Enemy contact 2</td>
<td>Detect NH90 -&gt; stay still</td>
<td>default</td>
<td>9</td>
<td>3</td>
<td>10</td>
<td>0</td>
<td>15000</td>
<td>12000</td>
<td>11990</td>
</tr>
<tr>
<td>Enemy contact 3</td>
<td>Detect MFF -&gt; evade</td>
<td>default</td>
<td>8</td>
<td>3</td>
<td>10</td>
<td>6</td>
<td>15000</td>
<td>12000</td>
<td>11990</td>
</tr>
</tbody>
</table>

Sensors

<table>
<thead>
<tr>
<th>Sensor #</th>
<th>Target</th>
<th>Target classes</th>
<th>Arc offset</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>MFF</td>
<td>4</td>
<td>300</td>
</tr>
<tr>
<td>2</td>
<td>HVU</td>
<td>2</td>
<td>300</td>
</tr>
<tr>
<td>3</td>
<td>NH90</td>
<td>3</td>
<td>300</td>
</tr>
</tbody>
</table>

Weapons

<table>
<thead>
<tr>
<th>weapon #</th>
<th>Range</th>
<th>Hit rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9000</td>
<td>1</td>
</tr>
</tbody>
</table>

COMS

<table>
<thead>
<tr>
<th>COM #</th>
<th>to Squad</th>
<th>info</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

|
E  The Matlab programmes

This appendix will give the Matlab files written for this thesis.

E.1 Simulator file for fixed initial positions

This code is the driverfile for running the simulations with the 384 fixed initial positions of the SSK.

```matlab
%% Simulatie programma voor MANA modellen
% Scriptie sea-basing simulation
% LTZ3 W.A. Knippenberg
% Januari 2014, versie 2.0
clc
clear all
close all
tic
rand('twister',1488);

%% Vaste parameters voor de simulatie:
R_f = 9.144 * 1000; %classificatie Bereik LFAS MFF [m]
R_h = 9.144 * 1000; %Bereik HELRAS NH90 [m]
Vf = 12; %Snelheid MFF [kts]
Vh = 100; %Snelheid NH90 [kts]
Vs = 6; %Snelheid Sub (wordt nog niet aangepast binnen model!)
Td = diptime(25*0.9144,Vf,Vh,8) ; %Diptijd (actief) van NH90 [s]

PosVerdeling_sub = 0; %Verdeling van beginpositie sub (0= 8 ... testposities, 1= 168 evenredig verdeeld, 2= N willekeurig ...
verdeeld)
N = 500; %Aantal willekeurige beginposities (als ...
PosVerdeling_sub == 2)

figure(1), hold on, axis([-130 130 -125 125]), grid on,
title('Initial positions for SSK'); %Figuur 1 van output ...

figure(2), hold on, axis([-40 40 -35 35]), grid on,
title('Positions where SSK is detected'); %Figuur 2 van ...
```
1)) scatter(150,150,[],'r'), plot(CC(:,1)./1000,CC(:,2)./1000, ... 
'color', [0.5 0.5 0.5]), xlabel('Distance [km]'), ylabel('Distance [km]'), legend('SSK detected by MFF-1', 'SSK detected by NH90-1', 'SSK detected by MFF-2', 'SSK detected by NH90-2', 'Defence route', 'location', 'southOutside')

37 figure(3), hold on, axis([-130 130 -125 125]), grid on, ... 
title('Starting positions for detected SSKs'); % Figur 3 van output voorbereiden
38 figure(3), scatter(150,150,[],[1 .6 .01]), ... 
scatter(150,150,[],[.8 .5 .2]), scatter(150,150,[],[.1 .6 1]), scatter(150,150,[],'r'), plot(CC(:,1)./1000,CC(:,2)./1000, ... 
'color', [0.5 0.5 0.5]), xlabel('Distance [km]'), ylabel('Distance [km]'), legend('SSK detected by MFF-1', 'SSK detected by NH90-1', 'SSK detected by MFF-2', 'SSK detected by NH90-2', 'Defence route', 'location', 'southOutside')

40 figure(4), hold on, axis([-130 130 -125 125]), grid on, ... 
title('Successful torpedo launches'); % Figur 4 van output voorbereiden
41 figure(4), scatter(150,150,'r'), ... 
scatter(150,150,'k'), plot(CC(:,1)./1000,CC(:,2)./1000,'color', ... 
[0.5 0.5 0.5]), xlabel('Distance [km]'), ylabel('Distance [km]'), legend('Launch position for Torpedo', 'Starting position for SSK', 'Defence route', 'location', 'southOutside')

44 % xml openen en ID's labelen aan juiste eenheden
switch approach
  case 1, Scen = ['Scenarios/kamikaze',num2str(Defence),'.xml'];
  case 2, Scen = ... ['Scenarios/semi_kamikaze',num2str(Defence),'.xml'];
  case 3, Scen = ['Scenarios/cautious',num2str(Defence),'.xml'];
end

[ s ] = xml2struct(Scen);

50 N_agents = length(s.specification.Squad);
51 for i=1:N_agents % ID's van verschillende squats verkrijgen
  Naam = num2str(s.specification.Squad{i}.SquadName.Text);
  switch Naam
    case ' SSK '
      ID_SSK = i;
    case ' HVU '
      ID_HVU = i;
    case ' NH90-1 '
      ID_NH90_1 = i;
    case ' MFF-1 '
      ID_MFF_1 = i;
    case ' NH90-2 '
      ID_NH90_2 = i;
    case ' MFF-2 '
      ID_MFF_2 = i;
  end
end

59 N_state = length(s.specification.Squad{ID_NH90_1}.state); % States van heli 1 uitlezen.Heli 2 is identiek aan heli 1!
60 for i =1:N_state
    staat = s.specification.Squad{ID_NH90_1}.state{i}.StateName.Text;
    switch staat
      case ' Sparel State '
        st_dip = i;
    end
end

69 % Vaste parameters instellen in xml-bestand
70 N_state = length(s.specification.Squad{ID_NH90_1}.state); % States van heli 1 uitlezen.Heli 2 is identiek aan heli 1!
71 for i=1:N_state
    staat = s.specification.Squad{ID_NH90_1}.state{i}.StateName.Text;
    switch staat
      case ' Sparel State '
        st_dip = i;
    end
end
if ...
    length(s.specification.Squad{ID_NH90_1}.state{st_dip}.SensorState) > 1; %kijken of heli 1 of meerdere sensoren heeft (bij ... cautious heeft hij er 2)
    s.specification.Squad{ID_NH90_1}.state{st_dip}.SensorState{1}.SensStClassify.Text = num2str(R_h); %classificatie range tegen varende sub van heli 1 instellen in juiste state
    s.specification.Squad{ID_NH90_1}.state{st_dip}.SensorState{2}.SensStClassify.Text = num2str(0.8*R_h); %classificatie range tegen stilligende sub van heli 1 instellen in ... juiste state
else
    s.specification.Squad{ID_NH90_1}.state{st_dip}.SensorState. ... van heli 1 instellen in juiste state
end
s.specification.Squad{ID_NH90_1}.state{1}.Speed.Text = num2str(Vh); %snelheid heli 1 instellen in juiste state (default)
s.specification.Squad{ID_NH90_1}.state{st_dip}.Trigger.duration.Text = num2str(Td); %actieve diptijd instellen voor heli 1
if exist('ID_MFF_2','var')
    s.specification.Squad{ID_MFF_2}.state.SensorState.SensStClassify.Text = num2str(R_f); %classificatie range van fregat 2 instellen
    s.specification.Squad{ID_MFF_2}.state.Speed.Text = num2str(Vf); ... %snelheid fregat 2 instellen
end
if exist('ID_NH90_2','var')
    s.specification.Squad{ID_NH90_2}.state{1}.Speed.Text = num2str(Vh); %snelheid heli 2 instellen in juiste state (default)
s.specification.Squad{ID_NH90_2}.state{st_dip}.Trigger.duration.Text = num2str(Td); %actieve diptijd instellen voor heli 2
if ...
    length(s.specification.Squad{ID_NH90_2}.state{st_dip}.SensorState) > 1; %kijken of heli 1 of meerdere sensoren heeft (bij ... cautious heeft hij er 2)
    s.specification.Squad{ID_NH90_2}.state{st_dip}.SensorState{1}.SensStClassify.Text = num2str(R_h); %classificatie range ... tegen varende sub van heli 2 instellen in juiste state
    s.specification.Squad{ID_NH90_2}.state{st_dip}.SensorState{2}.SensStClassify.Text = num2str(0.8*R_h); %classificatie ... range tegen stilligende sub van heli 2 instellen in ... juiste state
else
    s.specification.Squad{ID_NH90_2}.state{st_dip}.SensorState. ... van heli 1 instellen in juiste state
end
if approach == 2; %Als semi-kamikaze approach wordt gedraaid, ... aperture voor sensor1 instellen
    aperture = round(2* { atand(Vf/Vs) + .5*atand(Vs/Vf) } );
for k=1:length(s.specification.Squad{ID_SSK}.state)
    s.specification.Squad{ID_SSK}.state{k}.SensorState{1}. ... SensStMinAp.Text = num2str(aperture);
end
%% Variabele parameters (beginpos sub) maken en per run instellen
switch PosVerdeling_sub %check welke verdeling voor beginposities ... onderzeeboot is ingesteld
    case 1, S_sub = StartSub; % 168 vaste beginposities
    case 2, S_sub = StartSub_random(N); %N willekeurige beginposities
    case 0, S_sub = [32 32; -32 32; 32 -32; -32 -32; 45 0; 0 45; ... 0 -45; -45 0]*1000; %8 beginposities op kompaskoersen ... voor snelle systeemchecks
end

%% simulaties doen
seed = round(1000000*rand); %Seed binnen MANA instellen.

for i=1:size(S_sub,1) %voor elke beginpositie sub een simulatie ... draaien
    s.specification.Squad{ID_SSK}.Homes.HomePos.Home_x_coord.Text ... = num2str(S_sub(i,1)); %x-pos sub start instellen
    s.specification.Squad{ID_SSK}.Homes.HomePos.Home_y_coord.Text ... = num2str(S_sub(i,2)); %y-pos sub start instellen
    struct2xml(s, 'Scenarios/temp.xml'); %wegschrijven naar ... xml-bestand
    out = ['sim_out', num2str(i)]; %output bestand naam geven ... voor analyse achteraf
end

%% Standaard output
if cas_red(i,1) == 1 %als slachter in een sub was, was ... de verdediging succesvol
    figure(1), scatter(S_sub(i,1)./1000,-S_sub(i,2)./1000,'b');
elseif cas_blue(i,1) == 1 %als slachtoffer bij taskforce ... was, was verdediging niet succesvol
    figure(1), scatter(S_sub(i,1)./1000,-S_sub(i,2)./1000,'r');
else %anders waren er geen slachtoffers en was de simulatie eerder klaar (batterij) ... sub op: verdediging deels succesvol
    figure(1), scatter(S_sub(i,1)./1000,-S_sub(i,2)./1000,'g');
end
161 switch finder_ID
162 case ID_MFF_1
163     figure(2), ...
164     scatter(dead_ssk(1)./1000,-dead_ssk(2)./1000,[],...
165          [.1 .6 .01]);
166     figure(3), ...
167     scatter(S_sub(i,1)./1000,-S_sub(i,2)./1000,[],...
168          [.1 .6 .01]);
169     Xf_1 = Xf_1+1;
170 case ID_NH90_1
171     figure(2), ...
172     scatter(dead_ssk(1)./1000,-dead_ssk(2)./1000,[],...
173          [.8 .5 .2])
174     figure(3), ...
175     scatter(S_sub(i,1)./1000,-S_sub(i,2)./1000,[],...
176          [.8 .5 .2]);
177     Xh_1 = Xh_1+1;
178 case ID_MFF_2
179     figure(2), ...
180     scatter(dead_ssk(1)./1000,-dead_ssk(2)./1000,[],...
181          [.1 .6 1]);
182     figure(3), ...
183     scatter(S_sub(i,1)./1000,-S_sub(i,2)./1000,[],...
184          [.1 .6 1]);
185     Xf_2 = Xf_2+1;
186 case ID_NH90_2
187     figure(2), ...
188     scatter(dead_ssk(1)./1000,-dead_ssk(2)./1000,[],...
189          'r')
190     figure(3), ...
191     scatter(S_sub(i,1)./1000,-S_sub(i,2)./1000,[],...
192          'r');
193     Xh_2 = Xh_2+1;
194 end
195 elseif caslog(6) == ID_HVU %SSK fired torpedo
196     torpedo = caslog(15:16);
197     figure(4), scatter(torpedo(1)./1000,-torpedo(2)./1000,'r');
198     figure(4), scatter(S_sub(i,1)./1000,-S_sub(i,2)./1000,'k');
199     Xs = Xs+1;
200 end
201 disp(i)
202 end %Einde simulatie i
203 toc
204 
205 switch approach
206 case 1, disp('SSK followed kamikaze approach')
207 case 2, disp('SSK followed semi-kamikaze approach')
208 case 3, disp('SSK followed cautious approach')
209 end
210 switch Defence
211 case 1, disp('Defence strategy 1 was used')
212 case 2, disp('Defence strategy 2 was used')
213 case 3, disp('Defence strategy 3 was used')
214 end
215 simtime_seconds = toc
216 simtime_minutes = simtime_seconds/60
217 P_det = sum(cas_red)/length(cas_red) %Detectiekans uit standaard ...
if PosVerdeling_sub == 2
    z = norminv(0.975,0,1);
    CI_det = [P_det-z*sqrt((1/N)*P_det*(1-P_det)) ...
              P_det+z*sqrt((1/N)*P_det*(1-P_det))]
end

P_f1 = Xf_1/length(cas_red) %Detectiekans fregat 1 uit detail ...
      analyse
P_h1 = Xh_1/length(cas_red) %Detectiekans heli 1 uit detail ...
      analyse
P_f2 = Xf_2/length(cas_red) %Detectiekans fregat 2 uit detail ...
      analyse
P_h2 = Xh_2/length(cas_red) %Detectiekans heli 2 uit detail ...
      analyse
if PosVerdeling_sub == 2
    CI_f1 = [P_f1-z*sqrt((1/N)*P_f1*(1-P_f1)) ...
              P_f1+z*sqrt((1/N)*P_f1*(1-P_f1))]
    CI_f2 = [P_f2-z*sqrt((1/N)*P_f2*(1-P_f2)) ...
              P_f2+z*sqrt((1/N)*P_f2*(1-P_f2))]
    CI_h1 = [P_h1-z*sqrt((1/N)*P_h1*(1-P_h1)) ...
              P_h1+z*sqrt((1/N)*P_h1*(1-P_h1))]
    CI_h2 = [P_h2-z*sqrt((1/N)*P_h2*(1-P_h2)) ...
              P_h2+z*sqrt((1/N)*P_h2*(1-P_h2))]
end
P_s = Xs/length(cas_red) %Winkans voor sub
if approach == 1
    switch Defence
        case 1, [P_analytisch,P_ana_f,P_ana_h] = ...
              P_analytic(25*0.9144,Vf,6,Vh,R_f/1000,R_h/1000,8,1,1) ...
              %Detectiekans voor analytisch geval.
        case 2, [P_analytisch,P_ana_f,P_ana_h] = ...
              P_analytic(25*0.9144,Vf,6,Vh,R_f/1000,R_h/1000,8,2,1) ...
              %Detectiekans voor analytisch geval.
        case 3, [P_analytisch,P_ana_f,P_ana_h] = ...
              P_analytic(25*0.9144,Vf,6,Vh,R_f/1000,R_h/1000,8,1,2) ...
              %Detectiekans voor analytisch geval.
    end
end
E.2 Simulator file for random initial positions

This code is the drivefile for running the simulations with N random initial positions of the SSK.

```matlab
%% Simulatie programma voor MANA modellen
%% Scriptie sea-basing simulation
%% LTZ3 W.A. Knippenberg
%% January 2014, versie 1.0
clc
clear all
close all
tic
rand('twister',1488);

%% Vaste parameters voor de simulatie:
R_f = 9.144 * 1000 ; %classificatie Bereik LFAS MFF [m]
R_h = R_f ; %Bereik HELAS NH90 [m]
Vf = 12 ; %Snelheid MFF [kts]
Vh = 100 ; %Snelheid NH90 [kts]
Vs = 6 ; %Snelheid SSK [kts]
R_oct = 2.5*9144; %Straal [m] ingeschreven cirkel octagon

Td = diptime(R_oct/1000,Vf,Vh,8) ; %Diptijd (actief) van NH90 [s]

figuren = 1; %Figuren opmaken?(0 = nee, 1 = ja)
N = 7500; %Aantal willekeurige beginposities (N ≥ 2)
approach = 2; %Keuze van apprpach (1=kamikaze approach, ... 
2=semi-kamikaze approach, 3=cautious approach)
Defence = 3; %Keuze van Defencestrategie (1=1MFF&1NH90, ...
2=2MFF&1NH90 , 2=1MFF&3NH90)

%% Figure voorbereiden
if figuren == 1
[CC] = waypoints(8,[9469 22860]); %Waypoints van ... 
verdedegingsstrategie
CC = [CC ; CC(1,:)];
figure(1), hold on, axis([-130 130 -125 125]), grid on, ... 
title('Initial positions for SSK'); %Figuur 1 van output ...
voorbereiden
figure(1), scatter(150,150,'b'), ...
scatter(150,150,'r'),plot(CC(:,1)./1000,CC(:,2)./1000,'color' ...
,[0.5 0.5 0.5]),xlabel('Distance [km]'),ylabel('Distance ...
[km]'), legend('SSK classified','SSK launched ... 
torpedo','Defence route','location','southOutside' )
figure(2), hold on, axis([-40 40 -35 35]), grid on, ... 
title('Positions where SSK is classified'); %Figuur 2 ...
van output voorbereiden
figure(2), scatter(150,150,[],[.1 .6 .01]), ...
scatter(150,150,[],[.8 .5 .2]),scatter(150,150,[],[.1 .6 ...
1]),scatter(150,150,[],'r'),plot(CC(:,1)./1000,CC(:,2)./1000 ...
,'color', [0.5 0.5 0.5]),xlabel('Distance ...
[km]'),ylabel('Distance [km]'), legend('SSK classified by ... 
MFF-1','SSK classified by NH90-1','SSK classified by ...
MFF-2','SSK classified by NH90-2','Defence ... 
route','location','southOutside')
figure(3), hold on, axis([-130 130 -125 125]), grid on, ...
```
% Starting positions for classified SSKs
figure(3), scatter(150,150,[],[.1 .6 .01]), ...
scatter(150,150,[],[.8 .5 .2]),scatter(150,150,[],[.1 .6 ...
1]),scatter(150,150,[],'r'),plot(CC(:,1)./1000,CC(:,2)./1000 ...
,'color',[0.5 0.5 0.5]),xlabel('Distance [km]'),ylabel('Distance [km]'), legend('SSK classified by ...
MFF-1','SSK classified by NH90-1','SSK classified by ...
MFF-2','SSK classified by NH90-2','Defence ...
routelocation','southOutside')

% Successful torpedo launches
figure(4), hold on, axis([-130 130 -125 125]), grid on, ...
title('Successful torpedo launches'); % Figure 4 van ...
output voorbereiden
figure(4), scatter(150,150,'r'), ...
scatter(150,150,'k'),plot(CC(:,1)./1000,CC(:,2)./1000,'color', ...
[0.5 0.5 0.5]),xlabel('Distance [km]'),ylabel('Distance [km]'), legend('Launch position for Torpedo','Starting ...
position for SSK','Defence route','location','southOutside')

%% xml openen en ID’s labelen aan juiste eenheden
switch approach
    case 1, Scen = ['Scenarios/kamikaze',num2str(Defence),'_v2.xml'];
    case 2, Scen = ...
        ['Scenarios/semi_kamikaze',num2str(Defence),'_v2.xml'];
    case 3, Scen = ['Scenarios/cautious',num2str(Defence),'_v3.xml'];
end
[s] = xml2struct(Scen);

N_agents = length(s.specification.Squad);
for i=1:N_agents % ID’s van verschillende squats verkrijgen
    Naam = num2str(s.specification.Squad{i}.SquadName.Text);
    switch Naam
        case ' SSK '
            ID_SSK = i;
        case ' HVU '
            ID_HVU = i;
        case ' NH90-1 '
            ID_NH90_1 = i;
        case ' MFF-1 '
            ID_MFF_1 = i;
        case ' NH90-2 '
            ID_NH90_2 = i;
        case ' MFF-2 '
            ID_MFF_2 = i;
    end
end

%% Vaste parameters instellen in xml-bestand
%% Logfiles instellen
if figuren == 1
    s.specification.Battlefield.settings.RecPositions.Text = ' ...
    yes ';
else
    s.specification.Battlefield.settings.RecPositions.Text = ' no ';
end

%% Waypoints instellen:
[s] = setwaypoints(ID_MFF_1,R_oct,s);
[s] = setwaypoints(ID_NH90_1,R_oct,s);
if exist('ID_MFF_2','var')
    [s] = setwaypoints(ID_MFF_2,R_oct,s);
end
```matlab
end
if exist('ID_NH90_2','var')
    [s] = setwaypoints(ID_NH90_2,R_oct,s);
end

%% ASW-units instellen:
N_state = length(s.specification.Squad{ID_NH90_1}.state); %States ... van heli 1 uitlezen. Hel 2 is identiek aan heli 1!
for i =1:N_state
    staat = s.specification.Squad{ID_NH90_1}.state(i).StateName.Text;
    switch staat
        case ' Spare1 State '
            st_dip = i;
        end
end

s.specification.Squad{ID_MFF_1}.state.SensorState.SensStClassify.Text = num2str(R_f); %classificatie range van fregat 1 instellen
s.specification.Squad{ID_MFF_1}.state.Speed.Text = num2str(Vf); %snelheid fregat 1 instellen

if ... %kijken of heli 1 of meerdere sensoren heeft (bij ... cautious heeft hij er 2)
s.specification.Squad{ID_NH90_1}.state{st_dip}.SensorState{1}.SensStClassify.Text = num2str(R_h); %classificatie range ... van heli 1 instellen in juiste state
s.specification.Squad{ID_NH90_1}.state{st_dip}.SensorState{1}.classTable.classTablePoint.classTableRange.Text = num2str(R_h);
else
    s.specification.Squad{ID_NH90_1}.state{st_dip}.SensorState. ... SensStClassify.Text = num2str(R_h); %classificatie range ... van heli 1 instellen in juiste state
    s.specification.Squad{ID_NH90_1}.state{st_dip}.SensorState. ... classTable.classTablePoint.classTableRange.Text = ... num2str(R_h);
end

s.specification.Squad{ID_NH90_1}.state(1).Speed.Text = ... num2str(Vh); %snelheid heli 1 instellen in juiste state
s.specification.Squad{ID_NH90_1}.state{st_dip}.Trigger.duration.Text = num2str(Td); %actieve diptijd instellen voor heli 1

if exist('ID_MFF_2','var')
s.specification.Squad{ID_MFF_2}.state.SensorState.SensStClassify.Text = num2str(R_f); %classificatie range van fregat 2 instellen
s.specification.Squad{ID_MFF_2}.state.Speed.Text = num2str(Vf); %snelheid fregat 2 instellen
end

if exist('ID_NH90_2','var')
s.specification.Squad{ID_NH90_2}.state(1).Speed.Text = ... num2str(Vh); %snelheid heli 2 instellen in juiste state
s.specification.Squad{ID_NH90_2}.state{st_dip}.Trigger.duration.Text = num2str(Td); %actieve diptijd instellen voor heli 2
if ... %kijken of heli 1 of meerdere sensoren heeft (bij ... cautious heeft hij er 2)
s.specification.Squad{ID_NH90_2}.state{st_dip}.SensorState{1}.SensStClassify.Text = num2str(R_h); %classificatie range ... van heli 1 instellen in juiste state
```
s.specification.Squad{ID_NH90_2}.state{st_dip}.SensorState{1}. ... 
    num2str(R_h);

else
    s.specification.Squad{ID_NH90_2}.state{st_dip}.SensorState{1}. ... 
        SensStClassify.Text = num2str(R_h); %classificatie range ... 
        van heli 1 instellen in juiste state 
    s.specification.Squad{ID_NH90_2}.state{st_dip}.SensorState{1}. ... 
        classTable.classTablePoint.classTableRange.Text = ... 
        num2str(R_h);
end
end

%%% SSK instellen
switch approach
    case 1
        s.specification.Squad{ID_SSK}.state.Speed.Text = ... 
            num2str(Vs); %snelheid SSK instellen
    case 2
        s.specification.Squad{ID_SSK}.state{1}.Speed.Text = ... 
            num2str(Vs); %snelheid SSK instellen
        aperture = round(2*(atan2(Vf/Vs) + .5*atan2(Vs/Vf))) ; ... 
            %Als semi-kamikaze approach wordt gedraaid, aperture ... 
            voor sensor1 instellen
        for k=1:length(s.specification.Squad{ID_SSK}.state)
            s.specification.Squad{ID_SSK}.state(k).SensorState{1}. ... 
                SensStMinAp.Text = num2str(aperture); %arperture ... 
                    arc instellen
            s.specification.Squad{ID_SSK}.state(k).SensorState{1}. ... 
                classTable.classTablePoint.classTableRange.Text = ... 
                    num2str(round(1.3 * R_f)); %Sensor classificatie ... 
                        (reactie afstand) instellen
        end
    case 3
        for k=1:length(s.specification.Squad{ID_SSK}.state)
            s.specification.Squad{ID_SSK}.state(k).SensorState{3}. ... 
                classTable.classTablePoint.classTableRange.Text = ... 
                    num2str(round(1.3 * R_f)); %Sensor classificatie ... 
                        (reactie afstand) instellen
            staatniet = ... 
                s.specification.Squad{ID_SSK}.state(k).StateName.Text;
            if strcmp(staatniet, ' Contact State 2 ')
            else
                s.specification.Squad{ID_SSK}.state(k).Speed.Text = ... 
                    num2str(Vs); %snelheid SSK instellen
            end
        end
end

%%% veranderingen wegschrijven:
struct2xml(s, 'Scenarios/temp.xml'); %wegschrijven naar xml-bestand

out = ['test_out']; %output bestand naam geven voor analyse achteraf

%% simulaties doen
seed = round(1000000*rand); %Seed binnen MANA instellen.

cas = RunSim('temp',out,N,seed); %simulatie draaien, per ... 
    simulatie slachtoffers binnenkrijgen

cas_blue = cas{1};
cas_red = cas{2};
toc
%% Grove output inlezen
matlabpool open
parfor j=1:N
    caslog(j,:) = ... 
    xlsread(fullfile('C:\Users\knippenbergwa\AppData\Local\My Local Documents\simulatie bestanden\output\',out,'_caslocs_', ... 
        num2str(j),'.csv'),'A8:Q8'); %casualtie logboek ... 
    binnenkrijgen van sim 
    rij = 6 + ID_SSK;
if figuren == 1 
poslog_sub(j,:) = ... 
    xlsread(fullfile('C:\Users\knippenbergwa\AppData\Local\My Local Documents\simulatie bestanden\output\',out,'_pos_', ... 
        num2str(j),'.csv'],['D',num2str(rij),':E',num2str(rij)]); ... 
%positie logboek binnenkrijgen van sim 
end 
disp(j)
end
matlabpool close
toc

%% output analyseren
Xs = 0; Xf_1 =0; Xh_1=0; Xf_2 =0; Xh_2=0;
for i=1:N
    dead_ssk = caslog(i,2:3);
    finder_ID = caslog(i,11);
    if Defence == 2 
        ID_MFF_2 = ID_MFF_2;
    elseif Defence == 3 
        ID_MFF_2 = -1;
    end
    if caslog(i,6) == ID_SSK %SSK was detected
        switch finder_ID
            case ID_MFF_1
                Xf_1 = Xf_1+1;
            case ID_NH90_1
                Xh_1 = Xh_1+1;
            case ID_MFF_2
                Xf_2 = Xf_2+1;
            case ID_NH90_2
                Xh_2 = Xh_2+1;
        end
    elseif caslog(i,6) == ID_HVU %SSK fired torpedo
        Xs = Xs+1;
    end
if figuren == 1 
    if cas_red(i,1) == 1 %als slachter én sub was, ... 
        was de verdediging succesvol 
        figure(1), ... 
        scatter(poslog_sub(i,1)/1000,-poslog_sub(i,2)/1000 ... 
            ,'b'); 
    elseif cas_blue(i,1) == 1 %als slachtoffer bij ... 
        taskforce was, was verdediging niet succesvol 
        figure(1), ... 
        scatter(poslog_sub(i,1)/1000,-poslog_sub(i,2)/1000 ... 
            ,'r'); 
    else 
        %anders waren er geen ... 
        slachtoffers en was de simulatie eerder klaar ... 
        (batterij sub op): verdediging deels succesvol
218  figure(1), ...
    scatter(poslog_sub(i,1)/1000,-poslog_sub(i,2)/1000 ...
    ,'g');
  end
220
221  if caslog(i,6) == ID_SSK %SSK was detected
222    switch finder_ID
223    case ID_MFF_1
224      figure(2), ...
225      scatter(dead_ssk(1)./1000,-dead_ssk(2)./1000,[] ...
226      ,[.1 .6 .01]);
227      figure(3), ...
228      scatter(poslog_sub(i,1)/1000,-poslog_sub(i,2)/1000 ...
229      ,[],[.1 .6 .01]);
230    case ID_NH90_1
231      figure(2), ...
232      scatter(dead_ssk(1)./1000,-dead_ssk(2)./1000 ...
233      ,[.8 .5 .2])
234      figure(3), ...
235      scatter(poslog_sub(i,1)/1000,-poslog_sub(i,2)/1000 ...
236      ,[.8 .5 .2]);
237    case ID_MFF_2
238      figure(2), ...
239      scatter(dead_ssk(1)./1000,-dead_ssk(2)./1000,[] ...
240      ,[.1 .6 1]);
241      figure(3), ...
242      scatter(poslog_sub(i,1)/1000,-poslog_sub(i,2)/1000 ...
243      ,[],[.1 .6 1]);
244    case ID_NH90_2
245      figure(2), ...
246      scatter(dead_ssk(1)./1000,-dead_ssk(2)./1000,[] ...
247      ,',r')
248      figure(3), ...
249      scatter(poslog_sub(i,1)/1000,-poslog_sub(i,2)/1000 ...
250      ,[],'r');
251  end
252  elseif caslog(i,6) == ID_HVU %SSK fired torpedo
253    torpedo = caslog(i,15:16);
254    figure(4), ...
255    scatter(torpedo(1)./1000,-torpedo(2)./1000,'r');
256    figure(4), ...
257    scatter(poslog_sub(i,1)/1000,-poslog_sub(i,2)/1000 ...
258    ,',k');
259  end
260  end
261
262  % MeO weergeven, en mogelijk vergelijken met analytische berekening
263  clc
264  switch approach
265    case 1, disp('SSK followed kamikaze approach')
266    case 2, disp('SSK followed semi-kamikaze approach')
267    case 3, disp('SSK followed cautious approach')
268  end
269  switch Defence
270    case 1, disp('Defence strategy 1 was used')
271    case 2, disp('Defence strategy 2 was used')
272    case 3, disp('Defence strategy 3 was used')
273  end
274  simtime_seconds = toc
275  simtime_minutes = simtime_seconds/60
276  P_det = sum(cas_red)/length(cas_red) %Detectiekans uit standaard ...
277  analyse
278  z = norminv(0.975,0,1);
\[ CI_{\text{det}} = [P_{\text{det}} - z \cdot \sqrt{(1/N) \cdot P_{\text{det}} \cdot (1-P_{\text{det}})} \ldots \\
\quad P_{\text{det}} + z \cdot \sqrt{(1/N) \cdot P_{\text{det}} \cdot (1-P_{\text{det}})}] \]

\[ P_{f1} = X_{f_1}/\text{length(cas_red)} \% \text{Detectiekans fregat 1 uit detail analyse} \]

\[ P_{h1} = X_{h_1}/\text{length(cas_red)} \% \text{Detectiekans heli 1 uit detail analyse} \]

\[ P_{f2} = X_{f_2}/\text{length(cas_red)} \% \text{Detectiekans fregat 2 uit detail analyse} \]

\[ P_{h2} = X_{h_2}/\text{length(cas_red)} \% \text{Detectiekans heli 2 uit detail analyse} \]

\[ CI_{\text{f1}} = [P_{f1} - z \cdot \sqrt{(1/N) \cdot P_{f1} \cdot (1-P_{f1})} \ldots \\
\quad P_{f1} + z \cdot \sqrt{(1/N) \cdot P_{f1} \cdot (1-P_{f1})}] \]

\[ CI_{\text{f2}} = [P_{f2} - z \cdot \sqrt{(1/N) \cdot P_{f2} \cdot (1-P_{f2})} \ldots \\
\quad P_{f2} + z \cdot \sqrt{(1/N) \cdot P_{f2} \cdot (1-P_{f2})}] \]

\[ CI_{\text{h1}} = [P_{h1} - z \cdot \sqrt{(1/N) \cdot P_{h1} \cdot (1-P_{h1})} \ldots \\
\quad P_{h1} + z \cdot \sqrt{(1/N) \cdot P_{h1} \cdot (1-P_{h1})}] \]

\[ CI_{\text{h2}} = [P_{h2} - z \cdot \sqrt{(1/N) \cdot P_{h2} \cdot (1-P_{h2})} \ldots \\
\quad P_{h2} + z \cdot \sqrt{(1/N) \cdot P_{h2} \cdot (1-P_{h2})}] \]

\[ P_{s} = X_{s}/\text{length(cas_red)} \% \text{Winkans voor sub} \]

\[
\text{if } \text{approach} = 1
\]

\[
\quad \text{switch Defence}
\]

\[
\quad \quad \text{case 1, } [P_{\text{analytisch}}, P_{\text{ana_f}}, P_{\text{ana_h}}] = \ldots \\
\quad \quad \quad \text{P_{analytic}(25 \cdot 0.9144, V_f, 6, V_h, R_f/1000, R_h/1000, 8, 1, 1)} \ldots \\
\quad \quad \quad \% \text{Detectiekans voor analytisch geval.}
\]

\[
\quad \quad \text{case 2, } [P_{\text{analytisch}}, P_{\text{ana_f}}, P_{\text{ana_h}}] = \ldots \\
\quad \quad \quad \text{P_{analytic}(25 \cdot 0.9144, V_f, 6, V_h, R_f/1000, R_h/1000, 8, 2, 1)} \ldots \\
\quad \quad \quad \% \text{Detectiekans voor analytisch geval.}
\]

\[
\quad \quad \text{case 3, } [P_{\text{analytisch}}, P_{\text{ana_f}}, P_{\text{ana_h}}] = \ldots \\
\quad \quad \quad \text{P_{analytic}(25 \cdot 0.9144, V_f, 6, V_h, R_f/1000, R_h/1000, 8, 1, 2)} \ldots \\
\quad \quad \quad \% \text{Detectiekans voor analytisch geval.}
\]

\[
\quad \text{end}
\]

\[
\quad \text{end}
\]
E.3 Sub-functions for the simulator

This Section contains all the function written for this thesis in Matlab.

E.3.1 RunSim.m

This function runs a certain scenario in Matlab and gives the casualties per side as output.

```matlab
function cas = RunSim(scenario,out,N_runs,seed)
% Running a number of simulations for a specified scenario.
% Input:
% Scenario = name of scenario (excluded .xml extension)
% Out = name of outbestand (excluded .csv extension)
% N_runs = number of runs for this scenario
% seed = seed value
% Output:
% Cas_blue = number of casualties on blue side per run
% Cas_red = number of casualties on red side per run

[status] = dos([' "C:\Program Files (x86)\MANA/MANAC" ... 
    -f"C:\Users\Willem Knippenberg\Documents\1Scriptie\simulatie ... 
    bestanden\Scenarios\',scenario,'.xml" -n',num2str(N_runs),' ... 
    -m"C:\Users\Willem Knippenberg\Documents\1Scriptie\simulatie ... 
    bestanden\output\',out ,'.csv" -e',num2str(seed)]);

R_end = 6+N_runs;
cas = cell(2,1);
cas_blue = xlsread(['C:\Users\Willem ... 
    Knippenberg\Documents\1Scriptie\simulatie ... 
    bestanden\output\',out ,'.csv'],['C7:C',num2str(R_end)]);
cas_red = xlsread(['C:\Users\Willem ... 
    Knippenberg\Documents\1Scriptie\simulatie ... 
    bestanden\output\',out ,'.csv'],['D7:D',num2str(R_end)]);
cas{1} = cas_blue;
cas{2} = cas_red;
```

E.3.2 StartSub.m

This function defines all the fixed initial positions for the SSK.

```matlab
function [Start] = StartSub;

Dist = 45:5:(45+75); %start distance for submarine [km]
Direc = (0:15:359)*(pi/180); %start direction between ... 
    starpositions for submarine [rad]

k=0;
for j=1:length(Direc)
    for i=1:length(Dist)
        k = k+1;
        Start(k,1) = round(Dist(i)*1000*sin(Direc(j)));
        %x-coordinate [m]
        Start(k,2) = -round(Dist(i)*1000*cos(Direc(j)));
        %y-coordinate [m]
    end
end
```
E.3.3 diptime.m

This function calculates the active dip time for the NH-90 is the base scenarios.

```matlab
function [Td] = diptime(R,Vf,Vh,Nw)

arc = 360 / Nw; % arc of angle at HVU between 2 waypoints [deg]
L = 2 * R * tand(0.5 * arc) * (1000); % Length of 1 leg [m]
Tf = L / (Vf * (1852/3600)); % Time of 1 leg for frigate [s]
Tt = L / (Vh * (1852/3600)); % Travel time for HELO between 2 waypoints [s]
Tset = 2 * 6 * 60; % time to set and retrieve sonar [s]
Td = Tf - Tt - Tset; % Dpite time for HELO at waypoint [s]
```

E.3.4 P_analytic.m

This function calculates the analytical solution for the kamikaze approach.

```matlab
function [P,Pf,Ph] = P_analytic(Dist,V,U,Vh,Rf,Rh,Nw,Nf,Nh)

Dist = 25 * .9144; V=12; U=6; Vh=100; Rf = 9.144; Rh = 9.144; Nw ... = 8; Nf=1; Nh=1;

arc = 360 / Dist; % arc of angle at HVU between 2 waypoints [deg]
```


\[
l = 2 \cdot \text{Dist} \cdot \tan(0.5 \cdot \text{arc}) / 1.852; \quad \% \text{Length of 1 leg \ [nm]}
\]

\[
L = Nw \cdot l; \quad \% \text{Length of 1 full circle of frigate \ [nm]}
\]

\[
Pf = \left( \left( 2 \cdot Rf / L \right) \cdot \sqrt{1 + \left( V / U \right)^2} \right) - \left( Rf / L \right)^2 \cdot \left( 1 + \left( V / U \right)^2 \right); \quad \% \text{probability of detection for 1 frigate}
\]

\[
T_dip = \text{diptime(Dist, V, Vh, Nw); \ Duptime for HELO}
\]

\[
S_dip = U \cdot T_dip / 3600; \quad \% \text{relative distance helo is at the dip}
\]

\[
Ph = Nw \cdot (\pi \cdot R_h^2 + 2 \cdot S_dip \cdot R_h) \cdot \left( \frac{V}{U \cdot L^2} \right); \quad \% \text{probability of detection for 1 helo}
\]

\[
P = Nf \cdot Pf + Nh \cdot Ph; \quad \% \text{probability of detection for whole task group}
\]

---

E.3.5 \textit{setwaypoints.m}

This function sets all the waypoints of the defence at a certain distance. The orientation of the defence remains the same.

\[
\text{function } [s] = \text{setwaypoints(ID,Rin,s)}
\]

\[
\text{str2double(s.specification.Squad{ID}.Homes.HomePos.Waypoints.\ldots}
\text{numberOf.Text); \% read number of waypoints which have to be set}
\]

\[
\text{str2double(s.specification.Squad{ID}.Homes.HomePos.Home_x_coord.\ldots}
\text{Text); \% read old home pos X}
\]

\[
\text{str2double(s.specification.Squad{ID}.Homes.HomePos.Home_y_coord.\ldots}
\text{Text); \% read old home pos Y}
\]

\[
\text{arc = atan2(Hx_old,-Hy_old); \% calculate direction of home ...}
\text{position, relative to North}
\]

\[
\text{num2str(Hx_new); \% set new home X}
\]

\[
\text{num2str(-Hy_new); \% set new home Y}
\]

\[
\text{str2double(s.specification.Squad{ID}.Homes.HomePos.Waypoints.\ldots}
\text{waypoint{Nw}.x_coord.Text); \% read old X of last waypoint}
\]

\[
\text{str2double(s.specification.Squad{ID}.Homes.HomePos.Waypoints.\ldots}
\text{waypoint{Nw}.y_coord.Text); \% read old Y of last waypoint}
\]
30 \text{arc} = \text{atan2}(W0x,-W0y); %calculate direction of last waypoint, ...
31 \text{relative to North}
32 \text{Start}(1) = \text{Rout}\cdot\sin(\text{arc}); %calculate new last waypoint X
33 \text{Start}(2) = -\text{Rout}\cdot\cos(\text{arc}); %calculate new last waypoint Y
34 \text{[CC]} = \text{round(waypoints(Nw,Start))}; %calculate other waypoints
35 \text{for } i=1:Nw
36 \text{rij} = \text{Nw} + 1 - i;
37 \text{s.specification.Squad[ID].Homes.HomePos.Waypoints.waypoint(i).} ...
38 \text{x_coord.Text = num2str(CC(rij,1));} %set waypoint i X
39 \text{y_coord.Text = num2str(CC(rij,2));} %set waypoint i Y
40 \text{end}

E.3.6 \textit{waypoints.m}

This function calculates all the waypoints when the first waypoint is defined.