WEAPON-TARGET ASSIGNMENT FOR AIR DEFENSE OF NAVAL FORCES: MODELS AND HEURISTICS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

CANER ARSLAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
INDUSTRIAL ENGINEERING

SEPTEMBER 2024

Approval of the thesis:

WEAPON-TARGET ASSIGNMENT FOR AIR DEFENSE OF NAVAL FORCES: MODELS AND HEURISTICS

submitted by **CANER ARSLAN** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Industrial Engineering Department, Middle East Technical University** by,

Prof. Dr. Naci Emre Altun
Dean, Graduate School of **Natural and Applied Sciences** _______________

Prof. Dr. Pelin Bayındır
Head of Department, **Industrial Engineering** _______________

Prof. Dr. Ömer Kırca
Supervisor, **Industrial Engineering, METU** _______________

Prof. Dr. Orhan Karasakal
Co-supervisor, **Industrial Engineering, Çankaya University** _______________

Examining Committee Members:

Prof. Dr. Pelin Bayındır
Industrial Engineering, METU _______________

Prof. Dr. Ömer Kırca
Industrial Engineering, METU _______________

Prof. Dr. Bahar Yetiş Kara
Industrial Engineering, Bilkent University _______________

Assoc. Prof. Dr. Seçil Savaşaneril
Industrial Engineering, METU _______________

Assist. Prof. Dr. Derya Dinler
Industrial Engineering, Hacettepe University _______________

Date:05.09.2024

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Caner Arslan

Signature :

ABSTRACT

WEAPON-TARGET ASSIGNMENT FOR AIR DEFENSE OF NAVAL FORCES: MODELS AND HEURISTICS

Arslan, Caner
Ph.D., Department of Industrial Engineering
Supervisor: Prof. Dr. Ömer Kırca
Co-Supervisor: Prof. Dr. Orhan Karasakal

September 2024, 140 pages

Air defense in maritime environment is the protection of friendly naval assets against aerial threats. The objective of minimizing the threat to the defended assets requires optimal allocation of scarce defense resources to the targets. Flexible command and control functionality is necessary to handle the dynamic nature of events in air defense. Coordination and automation should be ensured between sensors and weapons in single ship or task group air defense environment. To provide effective decision support on the automation of the decisions, fast and efficient algorithms are needed in the command-and-control systems of ships.

The naval air defense planning (NADP) problem consists of maneuvering decisions of the ships and assigning/scheduling weapons and sensors to threats so that the total expected survival probability of friendly units is maximized. The NADP problem can be defined as a specific version of the Weapon Target Assignment (WTA) problem, which has been extensively studied in the literature since 1950s. Compared to other studies, the NADP problem includes new features that makes the problem definition more realistic and applicable. It also deals with sensor assignment requirements, weapon/sensor blind sectors, sequence dependent setup times and ship's radar signature.

In this thesis work, the development of exact/heuristic solution approaches that provide fast and efficient decision support on the automation of the NADP decisions is aimed. A mixed-integer nonlinear programming (MINLP) model of the NADP problem is presented and heuristic solution approaches are developed for both Static and Dynamic problem. The computational results demonstrate that these heuristic approaches are both fast and efficient in solving the NADP problem.

Keywords: Naval Air Defense, Weapon Target Assignment, Engagement Scheduling, Combat Management, Decision Support Automation, Military Operations Research

ÖZ

DENİZ KUVVETLERİNİN HAVA SAVUNMASI İÇİN SİLAH-HEDEF ATAMA: MODELLER VE SEZGİSEL YÖNTEMLER

Arslan, Caner
Doktora, Endüstri Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Ömer Kırca
Ortak Tez Yöneticisi: Prof. Dr. Orhan Karasakal

Eylül 2024 , 140 sayfa

Deniz görev gruplarının hava savunması, dost deniz unsurlarının hava tehditlerine karşı korunmasıdır. Savunulan unsurlara yönelik tehdidi en aza indirme hedefi, kıt savunma kaynaklarının hedeflere optimum şekilde tahsis edilmesini gerektirir. Hava savunma harbinde olayların dinamik yapısını ele almak için esnek komuta ve kontrol işlevselliği esastır. Tek bir gemi veya görev grubu hava savunma ortamında sensörler ve silahlar arasında koordinasyon ve otomasyon sağlanmalıdır. Kararların otomasyonu konusunda etkin karar desteği sağlamak için gemilerin komuta kontrol sistemlerinde online olarak hızlı ve etkin algoritmalara ihtiyaç duyulmaktadır.

Deniz hava savunma planlaması (NADP) problemi, dost unsurlara yönelik tüm düşman tehditlerinin imha edilme olasılığının maksimize edilmesi için gemilerin manevra kararları ile silah/sensör atama/çizelgeleme faaliyetlerini içerir. NADP problemi, 1950'lerden beri literatürde yoğun olarak çalışılan Silah Hedef Atama (WTA) probleminin özel bir versiyonu olarak tanımlanabilir. Diğer çalışmalarla karşılaştırıldığında, NADP problemi, problem tanımını daha gerçekçi ve uygulanabilir hale getiren yeni özellikler içermektedir. Bu problemde sensör atama gereksinimleri, silah/sensör kör sektörleri, sıra bağımlı hazırlık süreleri ve geminin radar kesit alanı hususları da ele alınmaktadır.

Bu tez çalışmasında, NADP kararlarının otomasyonu konusunda hızlı ve etkin karar desteği sağlayan kesin/sezgisel çözüm yaklaşımlarının geliştirilmesi amaçlanmıştır. NADP probleminin matematiksel modellenmesi sunulmuş ve Statik ve Dinamik problem için heuristik çözüm yaklaşımları geliştirilmiştir. Sonuçlar geliştirilen heuristik yaklaşımların NADP problemini çözmede hem hızlı hem de etkili olduğunu göstermektedir.

Anahtar Kelimeler: Deniz Hava Savunma, Silah-Hedef Atama, Angajman Çizelgeleme, Savaş Yönetim, Karar Destek Otomasyonu, Askeri Yöneylem Araştırması

This thesis is dedicated to a future where "Peace at Home, Peace in the World", as envisioned by Mustafa Kemal Atatürk, becomes a guiding principle for global prosperity. As we tackle the complex challenges of military operations, our mission remains constant: to ensure deterrence and uphold peace.

ACKNOWLEDGMENTS

I would like to express my gratitude to my supervisors, Prof. Ömer KIRCA and Prof. Orhan KARASAKAL, for their guidance throughout my study. Their insights, constant support, and expertise in the field of military operations research were invaluable in shaping this work. They made it possible for me to have both an enjoyable and productive research period. The knowledge and experience I have gained during this period will guide me in my academic and professional career.

I also extend my thanks to the members of my thesis committee, Prof. Bahar Yetiş KARA, Prof. Pelin BAYINDIR and Assoc.Prof. İsmail Serdar BAKAL, for their valuable feedback and contributions, which greatly enriched the quality of this work.

But my biggest thanks go to my wife and two daughters for their understanding and patience. Their unwavering support and sacrifices enabled me to dedicate myself to this endeavor. Their contribution to this success is immense.

I must acknowledge the financial support provided by the Foudation of the Scientific and Technical Research Council of Türkiye (TUBITAK). Being a recipient of the BIDEB 2211-A National PhD Scholarship Program has greatly motivated my academic endeavors.

Lastly, I also worked hard to fulfill my full-time duties at the Turkish Naval Headquarters throughout my PhD studies. I am deeply grateful to my institution, superiors, and colleagues for their support during this period.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

ABBREVIATIONS	
ASM	Anti-Ship Missiles
CMS	Combat Management System
CIWS	Close-in Weapon Systems
DP	Dynamic Programming
DSS	Decision Support System
EN	Engagement Network
EW	Electronic Warfare
GA	Genetic Algorithm
G/M	Guided Missile
HAW	Home All-the-Way
IR	Infrared
MCG	Mid Course Guidance
NADP	Naval Air Defense Planning
PDMS	Point Defense Missile Systems
RCS	Radar Cross-Section
SAM	Surface-to-Air Guided Missiles
SLS	Shoot-Look-Shoot
SP	Shortest Path
TAC	Thesis Advisory Committee
TG	Task Group
UCAV	Unmanned Combat Aerial Vehicles
WTA	Weapon Target Assignment

CHAPTER 1

INTRODUCTION

1.1 Motivation and Problem Definition

Air defense in the maritime environment is protection of the friendly naval assets against aerial threats. With recent advancements in air threat technologies, including improvements in range and speed, the risk to naval platforms has increased significantly. In a maritime combat scenario, neutralizing the threat to defended assets requires optimal allocation of scarce defense resources to targets. In this thesis study, we define and address the Naval Air Defense Planning (NADP) problem, which consists of maneuvering decisions of the ships and scheduling weapons and sensors to the threats in order to maximize the total expected survival probability of friendly units.

Traditionally, NADP decisions were made by task group commanders and executed by combat system operators of ships. However, the effectiveness of these decisions relies on experience, training, and cognitive competency of the commanders. As the maritime combat environment becomes more complex and sophisticated, the task of making the right decisions on time becomes more challenging for decision-makers. Especially with the emergence of supersonic/hypersonic missiles in naval warfare, the necessity of making these decisions within seconds has become crucial. As a result, real-time decision support tools are essential for modern navies.

As a recent example, the conflicts between Israel and Iran can be highlighted. On April 13, 2024, Iran launched more than 300 unmanned aerial vehicles and missiles towards Israel [1]. Such attacks, involving multiple threats simultaneously, underscore the critical importance of having fast decision-support algorithms in the background to enable air defense systems to respond effectively and in coordination.

As an example for modern naval command and control systems, the ADVENT Combat Management System (CMS) that is being used in the MILGEM project (national warship program) of the Turkish Navy can be given. ADVENT CMS has an architecture that facilitates the user to make fast and correct decisions and aims a flexible structure in the use of weapons and sensors.

The air defense of naval ships or task groups follows the principle of layered defense by employing friendly air sorties, long/medium-range surface-to-air missiles, point defense missile systems, naval guns, Close-In Weapons Systems (CIWS), and Electronic Warfare (EW) systems. The dynamic nature of events in anti-air warfare requires flexible command and control functionality, with coordination and automation between sensors and weapons in a single ship or task group air defense environment.

To provide effective decision support on the decisions, fast and efficient algorithms are needed online and embedded in the command and control systems of the ships. The fundamental goal of NADP is to find an allocation plan that assigns available weapons and sensors in the task group against incoming threats. The problem includes maneuvering decisions of the ships and scheduling the defense resources to enemy threats so that the total expected survival probability of the friendly units is maximized.

The NADP problem can be defined as a specific version of the Weapon Target Assignment (WTA) problem, which has been extensively studied in the literature since 1950s. Compared to other studies, the NADP problem includes new features that make the problem definition more realistic and applicable. It also deals with sensor assignment requirements, weapon/sensor blind sectors, sequence-dependent setup times, and the ship's infrared/radar signature.

In summary, this research aims to address the increasingly complex and challenging nature of air defense in the maritime environment by developing fast and efficient algorithms for the NADP problem, which optimizes the allocation of defense resources to minimize threats to friendly naval assets.

1.2 Contributions and Novelties

This study introduces, for the first time in the literature, a realistic mathematical modeling of the naval air defense planning problem. Aimed at practical applicability in combat management systems, the model incorporates factors such as weapon/radar inventory constraints, setup times, blind sectors of systems, and the RCS/IR signatures of ships. In the model, routing (heading of the ships) and engagement planning (against which threat, with which weapon and when) decisions are made, considering all these factors. In addition, various objective functions are modeled and compared to further explore the problem features.

It is crucial not only to solve the problem optimally, but also to provide a solution in a short time, ensuring practical usability. Therefore, the focus is on achieving both near-optimal and computationally efficient solutions to address the critical need for quick decision-making in air defense.

Acknowledging the computational challenges of the nonlinear MINLP model, this research introduces a linearized model as an approximation. The linearized model proves successful in providing effective solutions within a reasonable time for a significant number of scenarios. To further decrease solution times, the research applies a decomposition approach to the mathematical model. While this process effectively reduces solution times, it is still insufficient to reach a level suitable for real-time applications. Therefore, heuristic algorithms are developed to efficiently solve the NADP problem.

Experiments conducted on both the static and dynamic problems have shown that the algorithms developed for the NADP problem provide fast and effective solutions as aimed. In addition, this study demonstrates that machine learning models can contribute to solving this problem effectively. Thus, through this thesis study, a novel formulation and effective solution approaches for the naval air defense planning problem have been developed.

1.3 The Outline of the Thesis

The outline of the thesis is as follows:

The findings obtained from the review of literature are explained in Chapter 2, providing context for NADP problem addressed in this study. This review highlights significant methodologies and the evolution of the WTA problem, emphasizing the need for a problem definition and solution specific to the naval air defense problem.

Chapter 3 describes the NADP problem, distinguishing between the static and dynamic versions. The mathematical formulation (MINLP) of the Static NADP problem is detailed in Chapter 4, including key assumptions underlying the model and four different objective functions.

Chapter 5 of the thesis focuses on solution approaches for the Static NADP problem. It begins with the linearization of the model and introduces a Greedy Algorithm for obtaining initial solutions swiftly. A thorough comparison between the exact and linearized models is presented, followed by an exploration of the decomposition approach for the solution of the mathematical model. Additionally, the results obtained using the three different objective functions are compared and reviewed. In this chapter, heuristic solutions to the problem are discussed in detail, and Two and Three-Stage Heuristic Algorithms are introduced. The performance of these heuristics is analyzed, providing valuable insights into their efficiency in addressing the Static NADP problem.

Chapter 6 of the thesis is dedicated to the Dynamic NADP problem. The chapter begins with an introduction that describes the dynamic nature of the problem. Mathematical formulation of the dynamic problem is provided in Section 6.2. A simulation structure designed for the Dynamic NADP Problem is presented in Section 6.3, outlining the steps involved in simulating the dynamic environment. The chapter then explores heuristic solution approaches tailored for the Dynamic NADP Problem in Section 6.4, including the $DHA^{GA+EN+DP}$ (dynamic version of the Three-Stage Heuristic Algorithm) Algorithm, and MHA^{EN+DP} (myopic approach) Algorithm. Section 6.5 presents the computational experiments conducted to evaluate the performance of these solution approaches. In Section 6.6, implementation of a modified-SLS firing policy and computational results are discussed. Additionally, a method for utilizing machine learning models in the threat prioritization step of the MHA^{EN+DP} Algorithm is proposed and tested in Section 6.7. This chapter contributes valuable insights into addressing the challenges posed by the dynamic nature of air defense scenarios in maritime environments.

Chapter 7 discusses the implementation of the NADP Decision Support Automation and proposes a methodology for the central use of the dynamic NADP Algorithm in combat management systems.

Finally, conclusions and planned future work are discussed in Chapter 8.

CHAPTER 2

REVIEW OF LITERATURE

In literature, it is observed that research in this area is gathered under the topic of the WTA problem. We can consider defense resources in the NADP problem as weapons and incoming threats as targets. The WTA problem aims to minimize the probability of destruction of the friendly assets by allocating the weapons in the inventory to the threats in the most appropriate way.

Relevant studies is reviewed to see the general features, assumptions, objectives, and solution methods of the WTA problem in literature. In terms of the available information regarding the problem environment (number of threats, number of weapons, routes and the final destination of the threats, probabilities of kill, and any other inputs needed to define the scenario), the articles on the WTA problem can be divided into two distinct groups; static and dynamic.

In the Static WTA (SWTA), all information regarding the environment is available and the problem is solved for a single decision instance. In this type of problem, possible subsequent developments in the environment are not taken into account. Whereas in the Dynamic WTA (DWTA) problem, the decisions taken in one stage may affect the decisions in the next time stage. In addition, some problem parameters may change over time and the solution of the problem may need to be updated by taking these changes into account.

There are various formulations of both SWTA and DWTA problems in the literature. A recent survey paper on the WTA problem presented by Kline et al. (2019) [2] summarizes the different formulations and exact/heuristic algorithms used to solve the problem.

In the earliest static formulation of the WTA problem defined by Manne (1958) [3], a scenario with m different weapon types and n targets is considered. There are d_j available inventory from weapon type j and each weapon type j kills the target i with probability p_{ij}. v_i represents the value of the target i. This problem is formulated as:

$$min \sum_{i=1}^{n} v_i \prod_{j=1}^{m} (1 - p_{ij})^{x_{ij}}$$

$$s.t. \sum_{i=1}^{n} x_{ij} \leq d_j, \forall j = 1, .., m$$

$$x_{ij} \geq 0, \forall i = 1, .., n, j = 1, .., m$$

where x_{ij} is the decision variable indicating the number of weapon type j assigned to target i. This is one of the simplest formulations of the WTA problem.

Figure 2.1 shows the most common features that help to classify articles into different groups in order to explain the assumptions of the studied WTA problem.

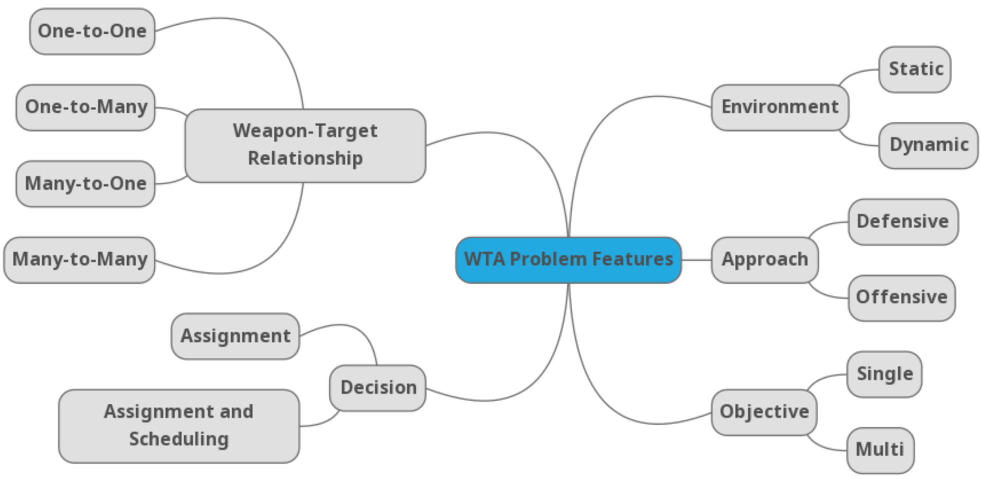


Figure 2.1: General Features to Group WTA Articles

It is seen that the formulation of the WTA problem changes according to the different ways of handling those features. As another feature that makes a difference in the problem formulation, the decisions considered in the studies are generally divided into two as assignment and assignment with scheduling. In some studies, besides the decision of which weapon will be allocated to which target, the decision of when the weapons will be fired is also made.

The presence of an offensive or defensive approach in the definition of the problem causes differentiation in the objective functions of the models. The maximum protection of friendly forces is aimed at the defensive version. In the offensive version, the maximum destruction of the targets is desired.

In terms of weapon-target relationship, studies can be analyzed in four different groups as one-to-one, one-to-many, many-to-one, and many-to-many. For instance, the formulation of Manne (1958) [3] presented above has a many-to-many weapon-target relationship. In this problem definition, weapon systems may be assigned to many targets and each target may be assigned by more than one weapon system.

The reviewed articles on the WTA problem are summarized in terms of these features and the solution approaches in Table 2.1.

The objective functions seen in the reviewed WTA studies are listed in Table 2.2. "Minimization of the value-weighted total survival probability of the targets" is the most commonly used objective in the literature.

Air defense problem for a naval task group was first studied by Karasakal (2004)[46]. "Maximization of the probability of no-leaker" objective is generally used in the articles (Karasakal et al. (2008)[11], Karasakal et al. (2011)[15], Taghavi and Ranjbar (2015)[17], Silav et al. (2019)[38], Silav et al. (2021)[41], Karasakal et al. (2021)[42]) on this problem. In Silav et al. (2019)[38] and Silav et al. (2021)[41], a bi-objective model is presented for this problem that maximizes the probability of no-leaker and minimizes the overall deviation from the initial schedule.

As a different approach from other WTA studies, Kwon et al. (1997) [4] and Kwon et al.(2007) [9] use "minimization of the overall firing cost" as the objective function and include in the model a constraint that ensures the probability of destroying targets be greater than or equal to the minimum desired level.

Lloyd and Witsenhausen (1986) [47] proved the NP-completeness of the WTA prob-

Table 2.1: Summary of Reviewed Articles on the WTA Problem

Article	Environment	Approach	Decision	Weapon-Target Relationship	Objective	Solution Approach	Solution Method
Kwon et al. (1997) [4]	Static	Offensive	Assignment and Scheduling	Many-to-Many	Single	Heuristic	A Greedy Type Heuristic.
Lee et al. (2002) [5]	Static	Offensive	Assignment	Many-to-One	Single	Heuristic	Ant Colony Optimization Algorithm
Lee et al. (2003) [6]	Static	Offensive	Assignment	Many-to-One	Single	Heuristic	Ant Colony Optimization Algorithm
Zeng et al. (2006) [7]	Static	Offensive	Assignment	Many-to-One	Single	Heuristic	Discrete Particle Swarm Optimization (DPSO) Algorithm
Ahuja et al. (2007) [8]	Static	Offensive	Assignment	Many-to-Many	Single	Heuristic	Very Large-Scale Neighborhood Search Algorithm Minimum Cost Flow Formulation-Based Construction Heuristic
Kwon et al. (2007) [9]	Static	Offensive	Assignment	Many-to-Many	Single	Heuristic	Branch-and-price Algorithm
Arslan et al. (2007) [10]	Static	Offensive	Assignment	One-to One	Single	Heuristic	Generalized Regret Monitoring With Fading Memory and Inertia Selective Spatial Adaptive Play
Karasakal (2008) [11]	Static	Defensive	Assignment	Many-to-Many	Single	Heuristic	Approximate Integer Linear Programming Models
Wu et al. (2008) [12]	Static	Defensive	Assignment	Many-to-One	Single	Heuristic	Genetic Algorithm
Madni and Andrecut (2009) [13]	Static	Offensive	Assignment	Many-to-One	Single	Heuristic	Simulated Annealing Algorithm
Cha and Kim (2010) [14]	Static	Offensive	Scheduling	-	Single	Heuristic	Branch and Bound Algorithm
Karasakal et al. (2011) [15]	Static	Defensive	Assignment and Scheduling	Many-to-Many	Single	Heuristic	Best Engagement Construction (BEC) Algorithm Quasi-Uniform Construction (QUC) Algorithm 2-Opt-Exchange (2OX) Algorithm Opt-Change (OC) Algorithm
Zhu et al. (2014) [16]	Static	Defensive	Assignment and Scheduling	Many-to-Many	Single	Heuristic	Opportunity Loss-Decentralized Markov Decision Process Model
Taghavi and Ranjbar (2015) [17]	Static	Defensive	Scheduling	-	Single	Heuristic	Branch and Bound Algorithm
Zhanwu et al. (2018) [18]	Static	Offensive	Assignment	Many-to-One	Single	Heuristic	Artificial Fsh Swarm Algorithm Improved Harmony Search Algorithm

Table 2.1 Continued:

Article	Environment	Approach	Decision	Weapon-Target Relationship	Objective	Solution Approach	Solution Method
Zhao et al. (2019) [19]	Static	Offensive	Assignment	Many-to-One	Single	Heuristic	Redesigned Auction-based Algorithm Task Swap Algorithm
Andersen et al. (2019) [20]	Static	Offensive	Assignment	Many-to-Many	Single	Heuristic	Linearization Using Probability Chains
Lu and Chen (2019) [21]	Static	Offensive	Assignment	Many-to-Many	Single	Exact	Column Enumeration with Branch and Bound
Sonuc (2020) [22]	Static	Defensive	Assignment	Many-to-One	Single	Heuristic	Modified Crow Search Algorithm
Luo et al. (2021) [23]	Static	Offensive	Assignment	Many-to-One	Single	Heuristic	Data-driven Policy Optimization with Deep Reinforcement Learning
Zhang et al. (2023) [24]	Static	Offensive	Assignment	Many-to-Many	Single	Heuristic	Whale Optimization Algorithm
Acar et al. (2023) [25]	Static	Offensive	Assignment	One-to-Many	Single	Heuristic	Quantum Algorithm
Park et al. (2023) [26]	Static	Defensive	Assignment and Scheduling	Many-to-Many	Single	Heuristic	A new MINLP Formulation and a Greedy Algorithm
Chang et al. (2023) [27]	Static	Offensive	Assignment	Many-to-Many	Single	Heuristic	Adaptive Large-scale Neighborhood Search (ALNS) Algorithm
Zou et al. (2024) [28]	Static	Defensive	Assignment	Many-to-Many	Multi	Heuristic	MOEA with a deep Q-network-based Adaptive Mutation Operator
Khosla (2001) [29]	Dynamic	Defensive	Assignment and Scheduling	One-to-Many	Single	Heuristic	Hybrid Genetic Algorithm
Xin et al. (2010a) [30]	Dynamic	Defensive	Assignment	One-to-One	Single	Heuristic	Tabu Search Algorithm
Xin et al. (2010b) [31]	Dynamic	Defensive	Assignment	One-to-One	Single	Heuristic	Rule-Based Constructive Heuristic
Leboucher et al. (2013a) [32]	Dynamic	Defensive	Assignment	One-to-One	Single	Heuristic	DPSO Combined with the Evolutionary Game Theory (EGT)
Leboucher et al. (2013b) [33]	Dynamic	Defensive	Assignment and Scheduling	One-to-One	Single	Heuristic	DPSO Combined with the Evolutionary Game Theory (EGT)
Ahner and Parson (2015) [34]	Dynamic	Defensive	Assignment	Many-to-Many	Single	Exact	Adaptive Dynamic Programming
Davis et al. (2017) [35]	Dynamic	Defensive	Assignment	Many-to-One	Single	Heuristic	Approximate Dynamic Programming

Table 2.1 Continued:

Article	Environment	Approach	Decision	Weapon-Target Relationship	Objective	Solution Approach	Solution Method
Gülpınar et al. (2018) [36]	Dynamic	Defensive	Assignment and Scheduling	Many-to-Many	Single	Heuristic	Traditional MDP Approach with a Backward DP Algorithm Construction Heuristic for Task-Resource Allocation with Retry Simulation-based DP Approach
Li et al. (2019) [37]	Dynamic	Defensive	Assignment	Many-to-One	Multi	Heuristic	Decomposition-based Multi-Objective Evolutionary Algorithms
Silav et al. (2019) [38]	Dynamic	Defensive	Assignment and Scheduling	Many-to-Many	Multi	Exact/ Heuristic	e-Constraint method Change and Exchange Heuristic New and Replace Heuristic
Summers et al. (2020) [39]	Dynamic	Defensive	Assignment	Many-to-One	Single	Heuristic	Approximate Dynamic Programming
Zhengrong et al. (2020) [40]	Dynamic	Offensive	Assignment	Many-to-One	Single	Heuristic	Defense Area Analysis Method
Silav et al. (2021) [41]	Dynamic	Defensive	Assignment and Scheduling	Many-to-Many	Multi	Heuristic	Weapon Target Swap (WTS) Algorithm
Karasakal et al. (2021) [42]	Dynamic	Defensive	Assignment and Scheduling	Many-to-Many	Multi	Heuristic	Artificial Neural Network using Adaptive Learning Algorithm
Kong et al. (2021) [43]	Dynamic	Offensive	Assignment	Many-to-Many	Multi	Heuristic	Multiobjective Particle Swarm Optimization Algorithm
Li et al. (2023) [44]	Dynamic	Offensive	Assignment	Many-to-Many	Single	Heuristic	Reinforcement Learning-based Multi-Agent Q-Learning
Liu et al. (2023) [45]	Dynamic	Offensive	Assignment	Many-to-One	Single	Heuristic	Reinforcement Learning Proximal Policy Optimization Algorithm

Table 2.2: List of the Objective Functions in Literature

Objective	**Articles**
Minimization of the value-weighted total survival probability of the targets	Khosla (2001) [29], Lee et al. (2002)[5], Lee et al. (2003)[6], Zeng et al. (2006)[7], Ahuja et al. (2007) [8], Wu et al. (2008) [12], Madni and Andrecut (2009) [13], Cha and Kim (2010) [14], Ahner and Parson (2015) [34], Davis et al. (2017) [35], Gülpınar et al. (2018) [36], Andersen et al. (2019) [20], Li et al. (2019) [37], Lu and Chen (2019) [21], Zhao et al. (2019) [19], Summers et al. (2020) [39], Sonuc (2020)[22], Acar et al. (2023) [25]
Maximization of the probability of no-leaker	Karasakal et al. (2008) [11], Karasakal et al. (2011)[15], Zhu et al. (2014)[16], Leboucher et al. (2013b)[33], Taghavi and Ranjbar (2015)[17], Silav et al. (2019)[38], Silav et al. (2021)[41], Karasakal et al. (2021)[42]
Maximization of the expected total damage value of the targets	Arslan et al. (2007)[10], Zhengrong et al. (2020)[40] Luo et al. (2021) [23], Kong et al. (2021) [43] Liu et al. (2023) [45], Zhang et al. (2023) [24] Zou et al. (2024) [28]
Maximization of the expected total value of the remaining assets	Xin et al. (2010a)[30], Xin et al. (2010b)[31]
Maximization of the safety margin	Leboucher et al. (2013b) [32]
Minimization of the overall firing cost	Kwon et al. (1997)[4], Kwon et al. (2007)[9] Li et al. (2019)[37], Zou et al. (2024) [28] Kong et al. (2021) [43], Chang et al. (2023) [27]
Minimization of the initial schedule disruption	Silav et al. (2019)[38], Karasakal et al. (2021)[42]
Maximization of allocation of same air targets into the schedule of weapons	Silav et al. (2021)[41]
Maximizing the damage effect per unit cost	Li et al. (2023)[44]

lem in its simplest form. Getting the optimal solution requires long running times even for small-size problems. Therefore, most of the WTA studies in literature focus on developing efficient heuristic solution methods.

In recent years, advancements in artificial intelligence have led to the incorporation of machine learning models into the solution approaches for this problem. Luo et al. (2021) [23] designs a data-driven policy optimization model with deep reinforcement learning. Liu et al. (2023) [45] applies a reinforcement learning proximal policy optimization algorithm to the dynamic WTA problem. A reinforcement learning-based multi-agent Q-learning model is proposed in Li et al. (2023) [44] for the multi-stage WTA problem. Zou et al. (2024) [28] presents a multi-objective evolutionary algorithm for the WTA problem that utilizes a deep Q-network-based adaptive mutation operator and a greedy-based crossover operator.

Some studies address sensor-target assignment decisions in addition to the WTA decisions. Bogdanowicz et al. (2007) [48] propose a model that aims to maximize the total value of assigning weapons and sensors to targets considering both independent and dependent weapon-sensor pairings. Chen et al. (2012) [49] construct the problem definition based on the benefit matrix for sensor-weapon-target pairings. Xin et al. (2018)[50] define the likelihood of successful engagement as the product of the weapon's probability of kill and the sensor's chance of detection.

Studies on the air defense problem for a naval task group have been scrutinized. The missile allocation problem (MAP) is introduced by Karasakal (2004) [46] to maximize air defense effectiveness of a naval task group. In Karasakal et al. (2008) [11], two integer linear programming models, which impose Shoot-Look-Shoot (SLS) engagement policy are developed for MAP. Karasakal et al. (2011) [15] present a discrete model to create optimal missile engagement schedule obeying SLS or a variant policy. Taghavi and Ranjbar (2015) [17] develop a branch and bound based heuristic algorithm to schedule missiles of a single weapon system. In Silav et al. (2019) [38], Silav et al. (2021) [41] and Karasakal et al. (2021) [42], air defense problem for a naval task group in dynamic environment is studied. A bi-objective missile rescheduling model is developed to update the engagement plan in case of disruptions in the initial schedule such as emergence of a new threat, failure of a SAM system, or destruction of an incoming threat. Karasakal et al. (2021) [42] proposes an artificial neural network solution approach that includes an adaptive learning algorithm to incorporate the decision maker's preferences into the model in order to get autonomous decision support.

Besides the academic articles, there are some technical reports on combat resource allocation planning in naval engagements in literature. In Benaskeur et al. (2007) [51], a metaheuristic algorithm based on Tabu search is presented for the coordination of anti-air warfare hardkill and softkill weapon systems for one warship. Blodgett et al.(2002) [52] use a naive Bayes method to decide the positioning of the ship to be able to use the hardkill and the softkill weapons most effectively.

Table 2.3 summarizes the assumptions of the studies on the naval WTA problem. The NADP problem studied in this article is also included in the table.

Table 2.3: List of Reviewed Articles on the Naval WTA Problem

Assumption	**Karasakal et al. (2008) [11]**	**Karasakal et al. (2011) [15]**	**Taghavi and Ranjbar (2015) [17]**	**Silav et al. (2019) [38]**	**Silav et al. (2021) [41]**	**Karasakal et al. (2021) [42]**	**Benaskeur et al. (2007) [51]**	**Blodgett et al.(2002) [52]**	**The NADP Problem**
C2 Capability	Full Coordination	Full Coordination	One Ship	Full Coordination	Full Coordination	Full Coordination	One Ship	One Ship	Full Coordination
Simultaneous Engagement Capacity	Unlimited	Unlimited & Limited	-	Unlimited	Unlimited	Unlimited	Two Concurrent	Two Concurrent	Limited
Missile Allocation Policy	SLS	SLS SSL	SLS	SLS	SLS	SLS	-	-	SLS
Setup Time	Fixed	Fixed	No	Fixed	Fixed	Fixed	-	-	Sequence Dependent
Sensor Illumination Requirements	No	No	No	No	No	No	-	No	Yes
Weapon-Sensor Blind Sectors	No	No	No	No	No	No	No	Yes	Yes
Maneuvering of the Ship	No	No	No	No	No	No	No	Yes	Yes

CHAPTER 3

DESCRIPTION OF THE NADP PROBLEM

A naval task group (TG) is a formation of various types of warships deployed together to accomplish a specific task or activity. Ships in the TG may possess different types of defense systems. Some ships, such as frigates and destroyers, may have area defense systems together with self-defense systems, whereas some ships may have no defense systems or only self-defense systems. Short-range surface-to-air missiles (SAM), close-in weapon systems (CIWS), naval guns, and softkill systems such as jammers, decoys, and flares can be considered as self-defense systems. Area defense systems such as medium- or longer-range SAMs can cover other ships within their effective range. Detailed explanation regarding the air defense systems of warships is presented in Appendix A.

When considering the planning of self-defense systems, it may be possible to reduce the problem to a single-ship-level problem. However, under the assumption of full coordination, self-defense systems need to be considered in order to determine the distribution of area defense systems and provide efficient air defense for all of the ships in the task group.

The emergence of new technologies, such as supersonic/hypersonic air-to-surface missiles (ASM) and unmanned combat aerial vehicles (UCAV), in threat environments has hardened the problem of air defense. The air threat faced by naval platforms worth billions of dollars has led to great advances in both quality and quantity in the last 20 years. The speed of ASMs has increased up to 5 to 6 Mach levels, while their range has reached nearly 1000 km.

In particular, types that can fly at low altitudes pose the greatest threat to naval platforms. Sea-skimming missiles can be detected at an average range of 20–25 km depending on the radar performance. If the speed of the threat missile is assumed to be 4 Mach, it will take a maximum of 15–18 seconds for the missile to reach the platform from this distance. The case that the enemy attack is carried out simultaneously in salvos or with sequential shots makes the situation much more complicated and pushes the limits for the coordination of the defense.

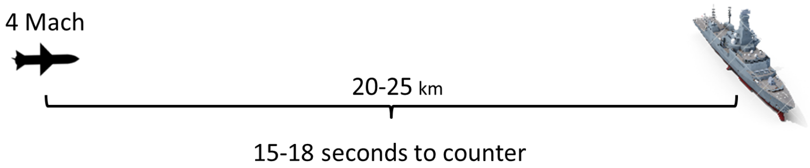


Figure 3.1: Seaskimming Missiles

In the context of naval warfare, fire channels are commonly used to engage airborne threats. A fire channel is a combination of a fire control radar and a weapon system that is used to engage and destroy targets. A fire control radar is used to track the target and provide accurate information about its position, velocity, and trajectory. This information is then used to aim the weapon system and fire at the target. Additionally, a fire control radar may provide illumination besides its primary function of tracking targets.

The time utilization plan of the fire control system radar is one of the main factors affecting the "maximum number of simultaneous engagements", which is perhaps the most important criterion for the combat performance of the naval platforms. The radar track and illumination requirements vary according to the guidance type of missiles.

Air defense missiles can have active homing (fire-and-forget), semi-active homing, or passive homing guidance. Active homing missiles are fired after the threat data are loaded, and they head towards the target by breaking all connections with the platform from which it is fired. This type of missiles carry a source of radiation onboard, as depicted in Figure 3.2.

In the case of semi-active guidance, the missile receives target information from the

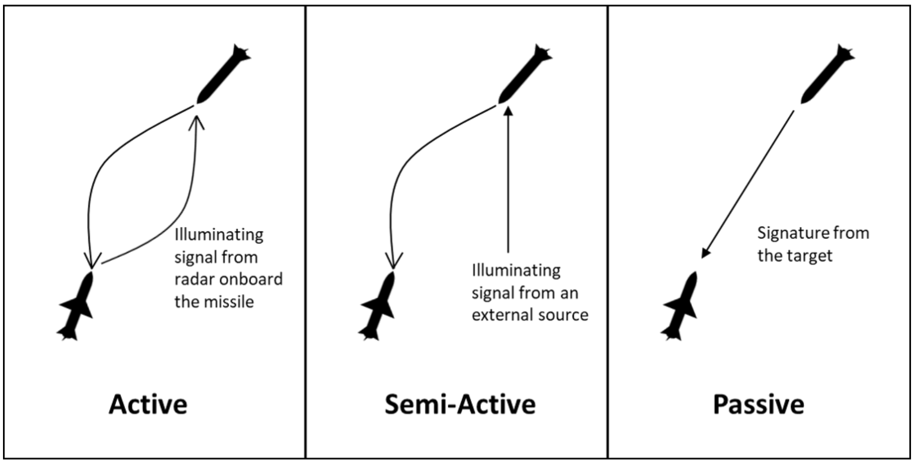


Figure 3.2: Guidance Types

illuminating sensor on the platform and is directed to the target using methods such as home-all-the-way (HAW) or mid-course-guidance (MCG). In the HAW method, the target is illuminated from the beginning of the engagement to the end, whereas in the MCG method, the missile is locked onto the target by illuminating only at the terminal stage. Unlike others, the passive guidance system is designed to detect the target through natural radiation, such as heat waves, light waves, or sound waves. In summary, SAM systems may have different radar illumination requirements.

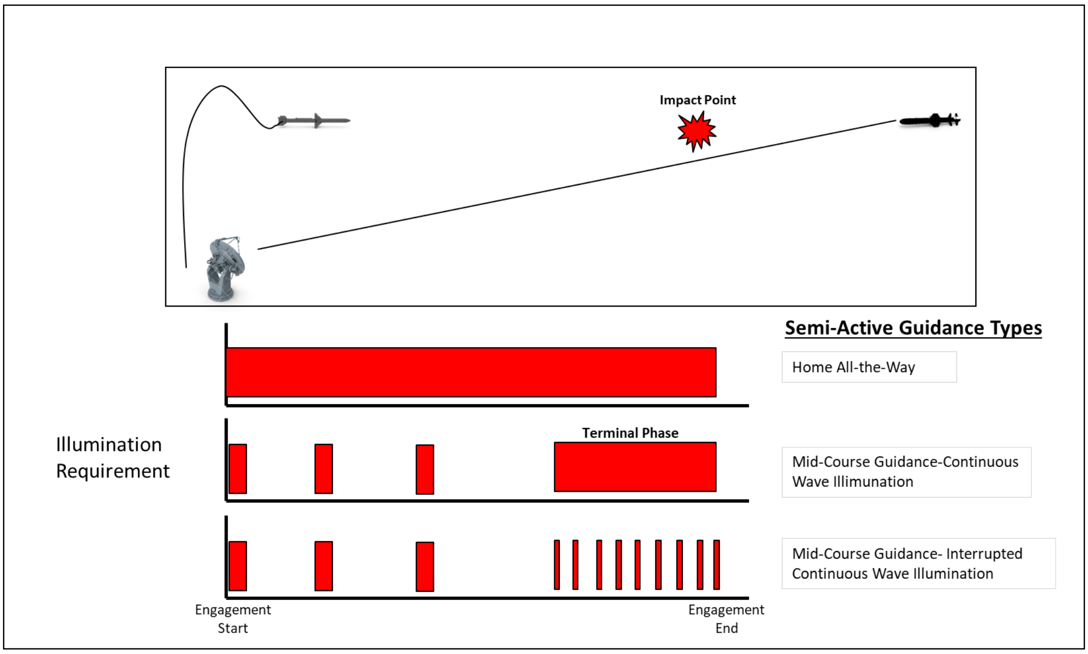


Figure 3.3: Semi-Active Guidance Types

Some SAM systems have retargeting capabilities. Retargeting capability provides the opportunity to direct the missile to a new target during flight. For instance, if the targeted threat ASM is destroyed before or a new higher-priority threat emerges, the SAM in flight can be retargeted.

Threat missiles can apply waypoints so that the eventual targets of the threats cannot be predicted at the beginning.

Another aspect that complicates the problem is that weapons and sensor systems may have blind sectors, as shown in Figure 3.4. Especially during the engagement of semi-active SAM systems, the threat ASM must be constantly in the coverage area of the sensors.

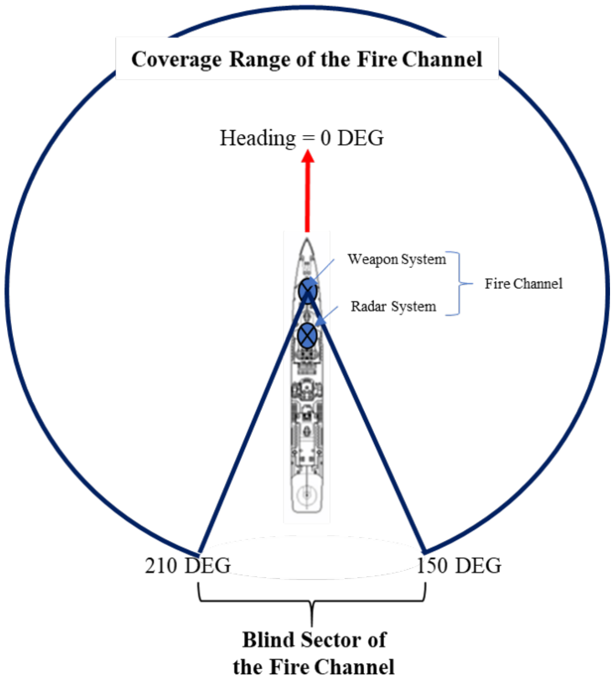


Figure 3.4: Blind Sector Example

Platforms must provide their defense by using their limited resources effectively. Taking into account possible future threats in engagement planning and the efficient use of defense resources is a critical issue that affects the final survival of a ship. Therefore, other defense systems, such as Naval Gun, point defense weapon systems, and soft-kill systems (jammer, chaff, flare, and decoys) need to be considered in the NADP problem.

In addition, there may be interactions among defense resources [53]. Therefore, it is necessary to coordinate the hardkill and softkill systems in order to have feasible plans.

The maneuvering of a ship can also be regarded as a defense resource that can be used to increase the ship's probability of survival. The ship can turn to expose its minimum radar cross-section (RCS) or IR signature to the incoming ASM. RCS is called the electromagnetic signature of an object, and it indicates radar detectability. Figure 3.5 shows an example radar cross section of a warship[54]. Objects with larger RCS can be detected more easily. Therefore, the NADP problem includes ship maneuvering decisions to increase the probability of survival.

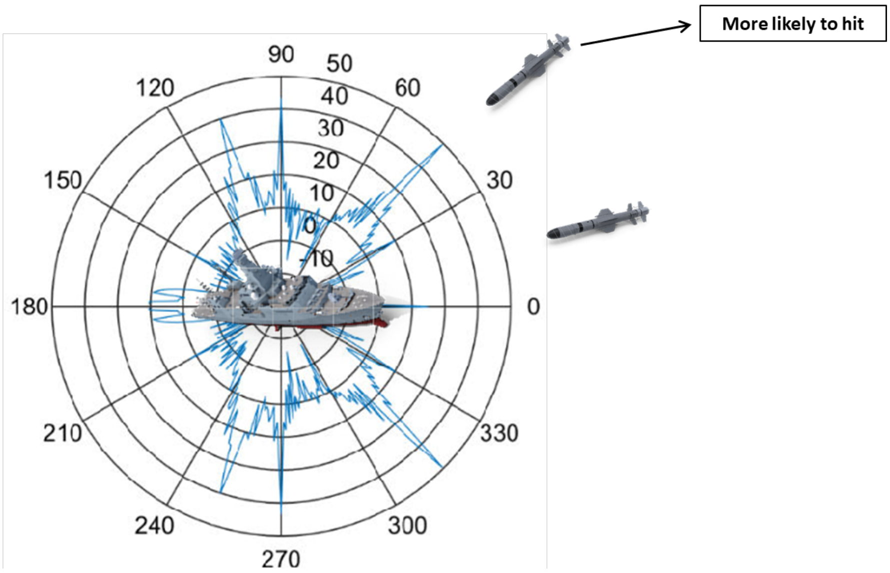


Figure 3.5: Example Radar Cross Section of a Warship

3.1 Static NADP Problem

In this thesis work, we studied both the static and the dynamic versions of the NADP problem. In the Static NADP problem, it is assumed that all the information about the threats that will appear in the future are known at time period 0 and the engagement planning is made accordingly. This includes information such as the detection and the hit times of the threats, possible engagement alternatives, and probability of kill values. Consequently, we plan for the entire scenario duration based on these known parameters. The decisions are made in a single decision instance without considering possible subsequent developments in the environment. It is assumed that the platforms and the weapon/sensor systems on these platforms are alive and usable throughout the scenario duration. The Static NADP problem involves making optimal decisions based on the determined initial conditions, but the ability to respond to changes in the war environment is limited.

Static NADP does not represent a realistic warfare environment; however, dealing with the static version first helped us to understand the problem and enabled us to construct valuable solution approaches. In addition, the fact that in the dynamic version, each new situation can be handled as a static problem with the data at hand, motivated us to start with the static problem first.

3.2 Dynamic NADP Problem

The Dynamic NADP problem extends the scope to address changing situation and evolving threats in real-time. Unlike the static version, the dynamic problem considers the fact that decisions made in one stage may impact decisions in subsequent stages. This dynamic nature of the problem requires continuous updates to the decision support system to adapt to evolving conditions such as the emergence or disappearance of threats, breakdowns in weapons or radar systems, changes in threat directions, and adjustments to other problem parameters in the maritime environment.

In a realistic naval warfare, threats appear one by one sequentially starting from a zero threat environment and the NADP decisions need to be updated in the face of every changing situation. Therefore, the renewed decisions have to be figured out in a feasible time to have an applicable decision support system. The Dynamic NADP problem involves making decisions on the allocation of defense resources while considering factors such as the emergence or disappearance of threats, breakdowns in weapons or radar systems, changes in threat directions, and adjustments to other problem parameters. The probability of platforms and weapon/sensor systems being destroyed by the threats at some point during the scenario also needs to be taken into account.

The goal remains the same, to maximize the total expected survival probability of friendly naval units. The decision support tools for the Dynamic NADP Problem need to operate in real-time, providing efficient algorithms embedded in the command and control systems of naval ships.

In summary, both the Static and Dynamic NADP Problems are integral parts of addressing the increasingly complex nature of air defense in the maritime environment. The Static version focuses on a single decision instance, while the Dynamic version incorporates real-time updates to adapt to the ever-changing conditions of naval warfare.

CHAPTER 4

MATHEMATICAL FORMULATION OF THE STATIC NADP PROBLEM

The NADP problem described in the previous chapter can be modeled as a mixed-integer nonlinear programming (MINLP) problem. In this chapter, assumptions and the mathematical formulation developed for the problem is presented.

4.1 Assumptions of the Model

The assumptions used to develop the NADP model are described below:

- The NADP problem considers a TG composed of multiple ships, which may be outfitted with weapon systems that can serve as either self-defense or area-defense system. It is possible that ships within the group have different weapon and sensor systems.
- It is assumed that all the information about the threats that will appear in the future are known at time period 0 and the engagement planning is made accordingly.
- For each weapon system, there is a limited number of available rounds.
- Full coordination is assumed between the ships in the TG.
- SAM systems may have different guidance (active, semi-active, passive homing, etc.) systems. Consequently, SAM systems may have different radar assignment requirements.
- Ships may have limited concurrent fire channels. Therefore a fire control system can handle a limited number of simultaneous engagements.

- Radar (track/illumination starting time) and weapon (time between two consecutive firings) setup times are considered in the model.
- Engagement duration to a threat is assumed to be the sum of the solution time of the fire control problem, launch delay, flight time, and look-up time.
- SLS firing policy is implemented.
- Ship positions are taken as stationary. The velocity of the ships is assumed to be negligible compared with the high speed of the threat ASMs.
- It is assumed that the threat missiles follow a straight trajectory and that the eventual targets of the threats are known.
- Radar and weapons systems may have blind sectors. Ships can change their headings in order to keep the threat in the coverage area.
- It is assumed that the combat management system's threat evaluation algorithms determine the single-shot kill probabilities for each engagement alternative.

4.2 Indices and Sets

$i \in A$	set of incoming threats (ASMs, UCAVs, etc.)
$j \in S$	set of available weapon systems (SAMs, naval guns, CIWS, chaff, etc.)
$r \in R$	set of radar systems
$n \in N$	set of friendly platforms
$b \in BS$	set of blind sectors of the sensor systems
$f \in F$	set of fire channels (weapon-radar system combination)
$f \in F_j$	set of fire channels that uses weapon system j
$f \in F_r$	set of fire channels that uses radar system r
$f \in F_n$	set of fire channels on ship n
$t \in T$	set of time slots $\{1, 2, 3, .., ts\}$
$\alpha \in \Phi$	set of angles {0,1,2,..,359}
$(i, f) \in G$	set of valid combinations of threats and fire channels
$t \in T_{if}$	set of time slots that fire channel f can start engagement to threat i, $\{t \in T : (i, f) \in G$ and $[t, t + \triangle_{ift}] \subseteq [q_{if}, r_{if}]\}$
$t \in TR_{if}$	set of time slots that the radar in fire channel f can be assigned to threat i (threat i is within the radar's minimum and maximum distances for tracking)
$(f, \rho) \in J_{it}$	set of combinations of (f, ρ) that time slot t for threat i will be blocked (if an engagement is started by fire channel f against threat i at time ρ, then no engagement can be started at time t against the same threat) , $\{(f, \rho) :$ $(i, f) \in G, \{t, \rho\} \in T_{if}$, and $t \subseteq [\rho, \rho + \triangle_{if\rho}]\}$
$\rho \in U_{ift}$	set of time slots that fire control radar will be blocked if an engagement is started by fire channel f against threat i at time t (illumination time requirement for the engagements, including radar setup times), $\{\rho \in TR_{if} : (i, f) \in G$,$t \in T_{if}$, and $\rho \subseteq [t - s_r, t + \triangle_{ift} - 1]\}$

4.3 Parameters

q_{if}	earliest beginning time of first engagement between threat i and fire channel f
l_{if}	latest ending time of last engagement between threat i and fire channel f
$[q_{if}, l_{if}]$	engageability interval for $(i, f) \in G$
$\triangle_{ift}$	engagement duration of (i, f) pair if engagement starts in time slot t
d_j	available rounds on weapon system j
s_j	setup time between two consecutive fires from weapon j
v_i	threat value of ASM i
v_n	value of ship n
rt_i	time that threat i reaches its target
tg_i	target ship of threat i
s_r	radar r's setup time needed for tracking and illumination
μ_{if}	upper bound on number of engagements from fire channel f against threat i when using the SLS firing policy
μ'_{if}	upper bound on number of engagements from fire channel f against threat i when using the modified SLS firing policy
p_{ift}	single shot kill probability of the engagement from fire channel f against threat i at time slot t
h_{n0}	initial heading (true) of the ship n
mr_n	maximum maneuvering turn rate (degree per time period) of the ship n
θ_{int}	bearing (true) of the threat i from the ship n at time slot t
$[bss_{fb}, bse_{fb}]$	fire channel f's starting-ending blind sector angles, (bss_{fb}: starting angle of the blind sector b of fire channel f, bse_{fb}: ending angle of the blind sector b of fire channel f)
$rcs_{i\alpha}$	miss probability of threat i when bearing from the target ship is α just before the hit
ts	scenario duration (the hit time of the latest threat)

4.4 Decision Variables

$$Y_{ift} = \begin{cases} 1, & \text{if the weapon system of fire channel } f \text{ is scheduled to start engagement against threat } i \text{ at time slot } t \\ 0, & \text{otherwise} \end{cases}$$

$$Z_{ift} = \begin{cases} 1, & \text{if the radar of fire channel } f \text{ is assigned to threat } i \text{ at time slot } t \\ 0, & \text{otherwise} \end{cases}$$

H_{nt} = heading of the ship n at time slot t, positive variable within $[000 - 359]$

HI_{nt} = auxiliary heading variable of the ship n at time slot t, free variable within $[(-mr_n), (359 + mr_n)]$

$$HL_{nt} = \begin{cases} 1, & \text{(auxiliary variable) if heading of the ship } n \text{ is between } [-(mr_n), -1] \text{ at time slot } t \\ 0, & \text{otherwise} \end{cases}$$

$$HG_{nt} = \begin{cases} 1, & \text{(auxiliary variable) if heading of the ship } n \text{ is between } [360, 359 + (mr_n)] \text{ at time slot } t \\ 0, & \text{otherwise} \end{cases}$$

$$FA_{nit} = \begin{cases} 1, & \text{(auxiliary variable) if bearing of the threat is greater than the heading of the ship } n \text{ at time slot } t \\ 0, & \text{otherwise} \end{cases}$$

B_{nit} = relative bearing of the threat i from the ship n at time slot t, positive variable within $[000 - 359]$

$$FG_{ifbt} = \begin{cases} 1, & \text{(auxiliary variable) if bearing of the threat is greater than starting angle of the fire channel } f\text{'s blind sector at time slot } t \\ 0, & \text{otherwise} \end{cases}$$

$$FS_{ifbt} = \begin{cases} 1, & \text{(auxiliary variable) if bearing of the threat is smaller than ending angle of the fire channel } f\text{'s blind sector at time slot } t \\ 0, & \text{otherwise} \end{cases}$$

$$BR_{i\alpha} = \begin{cases} 1, & \text{if relative bearing of the threat } i \text{ from its target ship is } \alpha \text{ just before the hit} \\ 0, & \text{otherwise} \end{cases}$$

PK_i = hit probability of threat i determined by the target ship's RCS or Infrared signature just before the hit

4.5 Objective Functions

Having various objective functions may give decision makers the chance to choose the most suitable alternative according to their preferences. Therefore, four different objective functions are modeled and compared in this study.

OBJ-1: Maximization of the probability of no-leaker for the entire TG.

$$max \prod_{i \in A} \left(1 - PK_i \prod_{t \in T, f \in F} (1 - p_{ift})^{Y_{ift}} \right)$$

or in other way;

$$max \prod_{i \in A} \left(1 - PK_i \prod_{t \in T, f \in F} (1 - p_{ift} Y_{ift}) \right)$$

OBJ-2: Minimization of the maximum hit probability of the threats.

$$min\, W$$

subject to

$$PK_i \prod_{t \in T, f \in F} (1 - p_{ift})^{Y_{ift}} \leq W \qquad \forall i \in A$$

OBJ-3: Minimization of the value-weighted hit probability of the threats.

$$min \sum_{i \in A} v_i \left(PK_i \prod_{t \in T, f \in F} (1 - p_{ift})^{Y_{ift}} \right)$$

OBJ-4: Maximization of the value-weighted survival probability of the friendly units.

$$max \sum_{n \in N} v_n \prod_{i \in A: tg_i = n} \left(1 - PK_i \prod_{t \in T, f \in F} (1 - p_{ift})^{Y_{ift}}\right)$$

Comparison results of the objective functions are given in Section 5.4.

4.6 Constraints

Weapon System Scheduling Constraints:

$$\sum_{t \in T, i \in A, f \in F_j} Y_{ift} \leq d_j \qquad \forall j \in S, \qquad (4.1)$$

$$\sum_{t \in T_{if}} Y_{ift} \leq \mu_{if} \qquad \forall (i,f) \in G, \qquad (4.2)$$

$$\sum_{(f,\rho) \in J_{it}} Y_{if\rho} \leq 1 \qquad \forall i \in A, t \in T, \qquad (4.3)$$

$$\sum_{i \in A, f \in F_j, t \in \{t,..,t+s_j-1\}} Y_{ift} \leq 1 \qquad \forall j \in S, t \in T, \qquad (4.4)$$

- Constraint set (4.1) provides the number of allocated munitions from a weapon system can not exceed the available inventory.
- Considering the engagement intervals, constraint set (4.2) limits the number of firings from each fire channel that can be scheduled to each threat.
- Constraint set (4.3) enforces the SLS policy. The allocation of weapon systems to a threat is prevented before the previous engagement finishes. Constraint sets (4.1-4.3) is first appeared in Karasakal et al. (2011) [15].
- Constraint set (4.4) enforces the setup time requirement between two consecutive firings from a weapon system.

Radar Illumination Requirement Constraints:

$$Y_{ift} \leq Z_{if\rho} \qquad \forall (i,f) \in G, t \in T_{if}, \rho \in U_{ift}, \tag{4.5}$$

$$Z_{ift} + Z_{i'f't} \leq 1 \qquad \forall i \in A, f \in F, t \in TR_{if}, i' \in A \setminus \{i\}, f' \in F_r : f \in F_r, \tag{4.6}$$

- Constraint set (4.5) provides that fire control radars are assigned to engaged threats during the required illumination time periods. The setup time needed for radars to start tracking and illumination is also considered in this constraint. If a fire channel is previously assigned to a threat, the engagement can be started immediately without considering the radar setup time.
- Constraint set (4.6) indicates that at most one threat can be simultaneously illuminated by each radar. No radar setup time is required between the two engagements from different fire channels with the same radar.

Ship Heading Constraints:

$$H_{n0} = h_{n0} \qquad \forall n \in N, \tag{4.7}$$

$$HI_{nt} \geq H_{n(t-1)} - mr_n \qquad \forall n \in N, t \in T, \tag{4.8}$$

$$HI_{nt} \leq H_{n(t-1)} + mr_n \qquad \forall n \in N, t \in T, \tag{4.9}$$

$$HI_{nt} \leq 359 + mr_n HG_{nt} \qquad \forall n \in N, t \in T, \tag{4.10}$$

$$HI_{nt} \geq 360 HG_{nt} - mr_n HL_{nt} \qquad \forall n \in N, t \in T, \tag{4.11}$$

$$HI_{nt} \geq -mr_n HL_{nt} \qquad \forall n \in N, t \in T, \tag{4.12}$$

$$HI_{nt} \leq -1 + (360 + mr_n)(1 - HL_{nt}) \qquad \forall n \in N, t \in T, \tag{4.13}$$

$$HL_{nt} + HG_{nt} \leq 1 \qquad \forall n \in N, t \in T, \tag{4.14}$$

$$H_{nt} = HI_{nt} + 360 HL_{nt} - 360 HG_{nt} \qquad \forall n \in N, t \in T, \tag{4.15}$$

- Constraint sets (4.7-4.15) provide the following inequality without disrupting the linearity and ensure the ship's maneuvering to happen obeying the maximum turning limits.

$$H_{n(t-1)} - mr_n \leq H_{nt} \leq H_{n(t-1)} + mr_n \qquad \forall n \in N, t \in T,$$

- Heading is the direction in which a ship points at any given moment. It is described as the angular distance relative to north. 000 (or 360) indicates true north, and it goes clockwise through 359.
- Constraint set (4.7) sets the initial heading of the ship.
- The maximum turning limit requirement and the ship's allowable heading interval for the next time period can be defined using (4.8-4.14).
- The heading of the ship n at time t is determined using the auxiliary decision variables HL_{nt}, HG_{nt} and HI_{nt} in the constraint set (4.15).

Fire Channel Blind Sector Constraints:

$$B_{nit} = \theta_{nit} - H_{nt} + 360(1 - FA_{nit}) \qquad \forall i \in A, n \in N, t \in T, \tag{4.16}$$

$$B_{nit} - \frac{bss_{fb} + bse_{fb}}{2} \leq \frac{bss_{fb} - bse_{fb}}{2} - 1 + 360FG_{ifbt}$$

$$\forall (i,f) \in G, n \in N : f \in F_n, b \in BS, t \in T, \tag{4.17}$$

$$B_{nit} - \frac{bss_{fb} + bse_{fb}}{2} \geq \frac{bse_{fb} - bss_{fb}}{2} + 1 - 360FS_{ifbt}$$

$$\forall (i,f) \in G, n \in N : f \in F_n, b \in BS, t \in T, \tag{4.18}$$

$$Z_{ift} \leq 2 - (FS_{ifbt} + FG_{ifbt}) \qquad \forall (i,f) \in G, b \in BS, t \in T, \tag{4.19}$$

- Fire Channel Blind Sector Constraints prevent the assignment of fire channels to threats in blind areas.
- Constraint set (4.16) calculates the relative bearing (B_{nit}) of the threats from the ships.
- Constraint sets (4.17-4.18) check whether the relative bearing of the threats is within the defined blind sector starting-ending angle values, and constraint set (4.19) prevents radar assignment if the threat is in the blind sector.

Ship RCS/IR Signature Constraints:

$$\sum_{\alpha \in \Phi} BR_{i\alpha} = 1 \qquad \forall i \in A, \tag{4.20}$$

$$B_{(tg_i)i(rt_i)} = \sum_{\alpha \in \Phi} \alpha BR_{i\alpha} \qquad \forall i \in A, \tag{4.21}$$

$$PK_i = \sum_{\alpha \in \Phi} (1 - rcs_{i\alpha}) BR_{i\alpha} \qquad \forall i \in A, \tag{4.22}$$

- Showing minimum IR/RCS against the incoming threat can be seen as a defense resource since it can increase the survivability.
- The relative bearing of the threat from its target immediately before the hit ($B_{(tg_i)i(rt_i)}$) is determined by constraint sets (4.20-4.21).
- The hit probability of threat (PK_i) is calculated using constraint set (4.22) according to the RCS or Infrared signature in the final aspect of the target ship from the threat.

Variable Constraints

And finally constraint sets (4.23-4.34) are the variable constraints.

$$Y_{ift} \in \{0, 1\} \qquad \forall (i, f) \in G, t \in T, \tag{4.23}$$

$$Z_{ift} \in \{0, 1\} \qquad \forall (i, f) \in G, t \in T, \tag{4.24}$$

$$H_{nt} \in [0, 359] \qquad \forall n \in N, t \in T, \tag{4.25}$$

$$HI_{nt} \in [-mr_n, 359 + mr_n] \qquad \forall n \in N, t \in T, \tag{4.26}$$

$$HL_{nt} \in \{0, 1\} \qquad \forall n \in N, t \in T, \tag{4.27}$$

$$HG_{nt} \in \{0, 1\} \qquad \forall n \in N, t \in T, \tag{4.28}$$

$$FA_{nit} \in \{0, 1\} \qquad \forall i \in A, n \in N, t \in T, \tag{4.29}$$

$$B_{nit} \in [0, 359] \qquad \forall i \in A, n \in N, t \in T, \tag{4.30}$$

$$FS_{ifbt} \in \{0, 1\} \qquad \forall (i, f) \in G, b \in BS, t \in T, \tag{4.31}$$

$$FG_{ifbt} \in \{0, 1\} \qquad \forall (i, f) \in G, b \in BS, t \in T, \tag{4.32}$$

$$BR_{i\alpha} \in \{0, 1\} \qquad \forall i \in A, \alpha \in \Phi, \tag{4.33}$$

$$PK_i \geq 0 \qquad \forall i \in A, \tag{4.34}$$

During the modeling development process, run-time comparison is conducted for different modeling alternatives for Fire Channel Blind Sector and Ship Heading Constraints. As a result of these analyses, the model has taken its final form with the constraints described above.

4.7 Validation/Verification Tests of the Model

The NADP problem given in this chapter is solved using MATLAB and GAMS version 24.1. Scenario parameters are created using MATLAB and data exchange between MATLAB-GAMS is established with the GDXMRW tool included in GAMS. BARON is used as Mixed-Integer Nonlinear Programming (MINLP) solver.

Ten example scenarios are created in order to perform Validation/Verification tests and to see the performance of the model. The results are presented in Appendix B. It is seen that the model accurately represents the NADP problem.

CHAPTER 5

SOLUTION APPROACHES FOR THE STATIC NADP PROBLEM

The mathematical model proposed in the previous section is a mixed-integer nonlinear programming (MINLP) problem. As Lloyd and Witsenhausen (1986) [47] demonstrated the NP-completeness of the simpler form (WTA) of the NADP problem, it can be computationally expensive to solve exactly, even for small scenario sizes. A common approach for addressing this issue is to linearize the nonlinear equations in the model and obtain approximate solutions. After the linearization, decomposition of the mathematical model is applied as the first step toward developing a solution method. This process is observed to effectively reduce the previously high solution times, yet it still cannot reach a level sufficient for real-time applications. Therefore, heuristic solution methods have been developed based on the decomposition of the mathematical model. In this chapter, we present linearized model formulations, propose heuristic approaches for solving the static NADP problem, and examine the computational results.

5.1 Linearization of the Model

In the first trials, it is seen that the MINLP formulation of the NADP problem takes a very long time even for small problems. For this reason, linearization of the problem is needed in order to evaluate the success of the heuristic algorithms to be developed in the following process. The nonlinear parts of the objective function 1,2 and 3 models can be linearized using logarithms and piecewise linear functions. The details of the linearization processes are given in the Appendix C. The resulting linearized models are given below (plus the Constraint Sets 4.1-4.21,4.23-4.33):

Objective Function-1: Maximization of the probability of no-leaker for the entire TG.

$$max \sum_{i \in A} (c^1 b_i^1 + c^2 b_i^2 + c^3 b_i^3 + c^4 b_i^4) \tag{5.1}$$

$$s.t. - lnPK_i + \sum_{t \in T, f \in F} a_{ift} Y_{ift} \geq b_i^1 + b_i^2 + b_i^3 + b_i^4, \qquad \forall i \in A \tag{5.2}$$

$$lnPK_i = \sum_{\alpha \in \Phi} ln(1 - rcs_{i\alpha}) BR_{i\alpha} \qquad \forall i \in A, \tag{5.3}$$

$$0 \leq b_i^1 \leq Z_1, \qquad \forall i \in A \tag{5.4}$$

$$0 \leq b_i^2 \leq Z_2 - Z_1, \qquad \forall i \in A \tag{5.5}$$

$$0 \leq b_i^3 \leq Z_3 - Z_2, \qquad \forall i \in A \tag{5.6}$$

$$0 \leq b_i^4 \leq Z_4 - Z_3, \qquad \forall i \in A \tag{5.7}$$

Objective Function-2: Minimization of the maximum hit probability of the threats.

$$min\ lnW \tag{5.8}$$

$$s.t.\ lnPK_i - \sum_{t \in T, f \in F} a_{ift} Y_{ift} \leq lnW \qquad \forall i \in A \tag{5.9}$$

$$lnPK_i = \sum_{\alpha \in \Phi} ln(1 - rcs_{i\alpha}) BR_{i\alpha} \qquad \forall i \in A \tag{5.10}$$

Objective Function-3: Minimization of the value-weighted total hit probability of the threats.

$$min \sum_{i \in A} v_i (c^1 b_i^1 + c^2 b_i^2 + c^3 b_i^3 + +c^4 b_i^4) \tag{5.11}$$

$$s.t. - lnPK_i + \sum_{t \in T, f \in F} a_{ift} Y_{ift} \geq b_i^1 + b_i^2 + b_i^3 + b_i^4, \qquad \forall i \in A \tag{5.12}$$

$$lnPK_i = \sum_{\alpha \in \Phi} ln(1 - rcs_{i\alpha}) BR_{i\alpha} \qquad \forall i \in A, \tag{5.13}$$

$$0 \leq b_i^1 \leq Z_1, \qquad \forall i \in A \qquad (5.14)$$

$$0 \leq b_i^2 \leq Z_2 - Z_1, \qquad \forall i \in A \qquad (5.15)$$

$$0 \leq b_i^3 \leq Z_3 - Z_2, \qquad \forall i \in A \qquad (5.16)$$

$$0 \leq b_i^4 \leq Z_4 - Z_3, \qquad \forall i \in A \qquad (5.17)$$

5.2 A Greedy Algorithm for Getting An Initial Solution

As the problem size increases, even the linearized model cannot produce a solution within a reasonable time. Therefore, a simple Greedy Algorithm (GA) is developed to obtain an initial solution for the linearized models. This algorithm is our first heuristic solution approach to the NADP problem. The pseudocode for the algorithm is presented in Algorithm-1.

In this approach, threats are prioritized according to their values and hit probabilities. Then, the highest priority threat is selected, and the highest hit probability engagement to this threat within ts seconds[1] is chosen. All other engagement options that conflict with this selected engagement are then eliminated. If there is a weapon inventory and a possible engagement option left, this cycle continues. After determining the engagement decisions, heading decisions are made using a backward recursion dynamic programming approach (details are given in Section 5.6.2) in a way that will satisfy blind sector constraints and minimize the hit probabilities of the threats considering the IR/RCS signature at the time of hit.

[1] As a result of the experiments conducted on k, best greedy algorithm results are obtained when $k = ts$.

Algorithm 1 Greedy Algorithm for Getting An Initial Solution

Generate problem parameters

V: set of valid $\{i, f, t\}$ combinations of threat, fire channel and engagement time

d_j: remaining rounds on weapon system j

Initialize all decision variables as zero

while $|V| > 0$ and $\sum_{j \in J} d_j > 0$ **do**

Order the threats according to their priority values where

$priority_i = v_i \prod_{t \in T, f \in F} (1 - p_{ift})^{Y_{ift}}$

Select the highest priority threat i that there is an engagement alternative in the V set against it

Select the engagement against threat i ($\{i, f, t\}$ combination from the V set) which has the highest Pk value among the engagement alternatives within $k = ts$ seconds

$Y_{ift} = 1$

$Z_{if\rho} = 1, \forall \rho \in U_{ift}$; (illumination requirement of selected engagement)

$d_j = d_j - 1$ where $f \in F_j$

Update V set

-Delete $\{i, f, t\}$ combinations that violates SLS policy

-Delete $\{i, f, t\}$ combinations that violates weapon/radar setup times

-Delete $\{i, f, t\}$ combinations with overlapping illumination requirements

-Delete $\{i, f, t\}$ combinations if $d_j = 0$ where $f \in F_j$

end while

Determine H decision variables using a backward recursion DP Approach

Determine all other auxiliary decision variables

5.3 Comparison of the Exact and the Linearized Models

In this section, we present the computational results of the comparison between the exact and the linearized models. MATLAB is used to set up the scenario data, and GAMS version 24.1 is used to solve the nonlinear and linearized optimization models. The models are solved using a workstation with an Intel Xeon E-2246G CPU, a 3.6 GHz processor, and 16 GB of RAM. Experimentation scenarios are generated randomly using the parameter generation structure described in Appendix D.

Given that the NADP problem is NP-complete, it is unlikely that an exact solution can be obtained within a reasonable running time. Therefore, it is needed to linearize the nonlinear model. In our first experiment, we sought to assess the performance of the linearized model. We used BARON as the solver for the MINLP models and CPLEX as the solver for the linearized models. OBJ-1 models are used in this experiment. The running time limit is taken as 500 seconds for the linearized models and 1000 seconds for the MINLP models. We tested the models in 30 scenarios ranging from small to large.

As seen in Table 5.1, the MINLP model fails to produce even a feasible solution within the given time limit for 25 out of the 30 scenarios. The linearized model could not reach a feasible solution within the time limit for 5 scenarios, and reached the optimum solution for 14 scenarios. The primary goal of developing the greedy algorithm is to provide an initial solution for the nonlinear and linear models. As seen in Table 5.1, it can be said that the greedy algorithm produces good enough starting solutions in short period of time. When initializing with a hot start, the linearized model can achieve the optimal solution within the time limit for 21 scenarios. To evaluate the performance of the linearized model, we used the linearized model solution to initialize the nonlinear model. The results show that the nonlinear model cannot improve the linearized solutions within the time limit for any of the 30 scenarios. It is also observed that the linearized model finds the optimal solution for three scenarios (scenarios 1,2, 5), where the MINLP model can produce the optimal solution.

Table 5.2 shows the comparison results for smaller (1-3 ships, 1-2 threats, 1-6 weapon systems, and 1-5 radar systems) scenarios that MINLP model obtained optimal solutions. According to the results, the MINLP model reaches the optimum solution in 138 of the 500 scenarios within the time limit of 1000 seconds. The comparison results are calculated considering only these 138 scenarios. The linearized model successfully reached the optimal solution in 136 of these scenarios, with an approximation gap of only 0.004096%.

Based on these results, we can observe that the MINLP model solution takes too long even for small-sized problems, and the linearized model provides a successful approximation for the exact solution. Therefore, the linearized model is decided to be used for the comparison of the heuristic approaches in the subsequent experiments.

Table 5.1: Comparison Results of the Exact and the Linearized Model Solutions

			Linearized Model		**MINLP Model**		**Greedy Algorithm**		**Linearized Model with Greedy Hot Start**[b]		**MINLP Model with Linearized Solution Hot Start**[c]	
Scenario	**Number of Ships**	**Number of Threats**	$P_{noleaker}$	**Run Time**[a]	$P_{noleaker}$	**Run Time**	$P_{noleaker}$	**Run Time**	$P_{noleaker}$	**Run Time**	$P_{noleaker}$	**Run Time**
S-1	1	1	0.6115	0.10	0.6115	8.45	0.6115	0.10	0.6115	0.09	0.6115	0.69
S-2	1	1	0.7334	0.09	0.7334	5.21	0.7334	0.07	0.7334	0.09	0.7334	0.44
S-3	1	2	0.3365	0.12	0.3365	1000.45	0.3365	0.16	0.3365	0.11	0.3365	1000.17
S-4	1	3	0.2018	500.36	-	1000.47	0.1901	0.30	0.2018	500.36	0.2018	1001.00
S-5	2	1	0.5272	0.12	0.5272	5.15	0.5272	0.17	0.5272	0.12	0.5272	1.10
S-6	2	2	0.4687	0.21	0.4279	1000.28	0.4687	0.42	0.4687	0.20	0.4687	1000.31
S-7	2	3	0.4641	4.84	-	1000.28	0.4526	0.39	0.4641	4.40	0.4641	1000.59
S-8	2	4	0.0802	21.02	-	1000.35	0.0735	0.57	0.0802	7.39	0.0802	1000.71
S-9	3	2	0.7808	10.18	-	1000.47	0.7116	0.69	0.7808	3.01	0.7808	1001.01
S-10	3	3	0.7789	500.48	-	1000.73	0.5966	0.75	0.7915	500.48	0.7915	1001.62
S-11	3	4	0.5098	8.62	-	1000.61	0.1781	0.49	0.5098	8.08	0.5098	1001.18
S-12	3	5	0.1425	7.28	-	1000.39	0.1322	0.38	0.1425	6.83	0.1425	1000.85
S-13	4	3	0.8001	501.28	-	1001.44	0.6856	1.99	0.8121	465.42	0.8121	1003.36
S-14	4	4	0.2874	34.60	-	1000.57	0.2124	1.00	0.2874	55.20	0.2874	1001.24
S-15	4	5	0.0452	2.61	-	1001.07	0.0327	0.86	0.0452	3.07	0.0452	1002.26
S-16	4	6	0.1074	500.60	-	1000.76	0.0522	0.71	0.1115	338.94	0.1115	1001.75
S-17	5	4	0.6522	501.34	-	1001.43	0.5128	1.36	0.6587	501.32	0.6587	1003.47
S-18	5	5	0.3162	500.90	-	1001.06	0.2656	1.19	0.3163	500.89	0.3163	1002.47
S-19	5	6	0.2268	53.92	-	1000.96	0.1495	0.67	0.2268	54.08	0.2268	1002.27
S-20	5	7	0.4430	502.98	-	1145.60	0.2895	1.40	0.4517	502.96	0.4517	1145.10
S-21	6	5	0.4296	4.95	-	1001.06	0.2650	1.23	0.4296	5.32	0.4296	1002.47
S-22	6	6	0.2191	502.08	-	1001.83	0.0968	1.58	0.2231	80.73	0.2231	1005.05
S-23	6	7	0.2121	502.51	-	1002.07	0.2117	1.38	0.2280	172.40	0.2280	1006.01
S-24	6	8	- [d]	502.48	-	1287.60	0.0767	1.42	0.2639	503.37	0.2639	1287.37
S-25	7	6	0.3334	502.05	-	1001.97	0.2130	1.66	0.3587	125.89	0.3587	1005.16
S-26	7	7	-	501.75	-	1002.07	0.3723	1.89	0.5231	502.48	0.5231	1005.89
S-27	7	8	-	502.35	-	1007.64	0.0187	1.66	0.0525	502.99	0.0525	1007.30
S-28	8	6	0.2732	503.45	-	1001.23	0.2029	2.23	0.2752	168.58	0.2752	1003.57
S-29	8	8	-	504.21	-	3018.43	0.2343	2.73	0.2940	505.39	0.2940	3008.02
S-30	8	10	-	504.22	-	4987.21	0.1435	2.16	0.1435	506.05	0.1435	4984.20
	Average		-	272.72	-	1116.23	0.3016	1.05	0.3783	217.54	0.3783	1116.22

[a] Time in seconds.

[b] Greedy Algorithm solution is used as the initial solution.

[c] Linearized model solution is used as the initial solution.

− In this case, the solver could not reach a feasible solution within the time limit.

Table 5.2: Comparison Results of the Exact and the Linearized Model Solutions-2

	OBJ-1 Prob. of No Leaker	Run Time (seconds)
	Average Values	
Linearized Model	0.6463530	0.17
MINLP Model	0.6463794	51.76
	Percent Deviation % From the Optimal Solution (Average)	
Linearized Model	0.004096%	
	Number of Optimal Solutions	
Linearized Model	136/138	

5.4 Comparison of the Objective Functions

In this section, the results obtained using the three different objective functions (given in Section 4.5) are compared. The experiments are conducted using the linearized model with hot start using the greedy algorithm solutions. Linearization of OBJ-4 could not be performed; hence, that objective function is not used in the experiments. However, the values of the solutions obtained using the other three objective functions are also calculated in terms of the OBJ-4. The running time limit is set to 500 seconds for the 30 scenarios used in the previous experiment. The results of the comparison of objective functions are depicted in Table 5.3.

Since the OBJ-2 model provides the exact linearization of the original nonlinear model, it can be seen in the results that the linearized OBJ-2 model obtains the best solution in all of the scenarios in terms of objective function-2. However, it gives worse results than OBJ-1 and OBJ-3 models in terms of the other objective functions.

When OBJ-1 and OBJ-3 model results are compared, it is seen that both model solutions are quite close and it can be said that the models do not have a significant advantage over each other. Considering all objective functions, the OBJ-3 model provides slightly better results. This is expected since OBJ-1 does not consider the weights while OBJ-3 does. In terms of the running times of the models, the OBJ-3 model is slightly faster than the OBJ-1 model. Thus, the OBJ-3 model is decided to be the main model in the subsequent computational experiments.

Table 5.3: Comparison of the Objective Functions

	OBJ-1 Probability of No Leaker	OBJ-2 Min of Max Hit Prob.	OBJ-3 Value Weighted Threat Hit Prob.	OBJ-4 Value Weighted Ship Surv. Prob.	Run Time (seconds)
	Average Values				
Linearized Model (OBJ-1)	0.3783	0.3477 [a]	0.2380	0.7278	216.8606
Linearized Model (OBJ-2)	0.3355	0.3315	0.2645	0.7035	136.5292
Linearized Model (OBJ-3)	0.3747	0.3662	0.2301	0.7378	210.7127
	Percent Deviation (%) From the Best Solution (Average)				
Linearized Model (OBJ-1)	0.74	7.35	5.15	1.89	-
Linearized Model (OBJ-2)	17.69	0.00	18.07	5.65	-
Linearized Model (OBJ-3)	3.20	13.82	0.00	0.11	-
	Percent Deviation (%) From the Best Solution (Standart Deviation)				
Linearized Model (OBJ-1)	0.02	0.10	0.07	0.06	-
Linearized Model (OBJ-2)	0.19	0.00	0.16	0.08	-
Linearized Model (OBJ-3)	0.07	0.21	0.00	0.01	-
	Percent Deviation (%) From the Best Solution (Max)				
Linearized Model (OBJ-1)	0.08	0.31	0.20	0.34	-
Linearized Model (OBJ-2)	0.73	0.00	0.57	0.41	-
Linearized Model (OBJ-3)	0.34	0.82	0.00	0.03	-
	Number of Best Solutions				
Linearized Model (OBJ-1)	24	16	14	14	-
Linearized Model (OBJ-2)	6	30	5	5	-
Linearized Model (OBJ-3)	19	14	29	29	-

[a] The objective function values of the corresponding model solution are calculated in terms of each of the four objectives. For instance, this value represents the objective function 2 value of the solution of the linearized model with objective function 1.

5.5 Decomposition Approach for the Solution of the NADP Problem

Following the linearization of the model, our second step was to attempt the decomposition of the NADP model in pursuit of finding a solution method. The decisions within the NADP problem inherently fall into two primary categories: engagement planning, encompassing weapon scheduling (Y_{ift}) and sensor-threat allocations (Z_{ift}), and the management of ship headings. The decomposition of these activities into two sequential stages may have the potential to expedite the solution of the mathematical model. Figure 5.1 illustrates the bifurcation of the problem into two stages, elucidating which constraints should be considered in each respective phase.

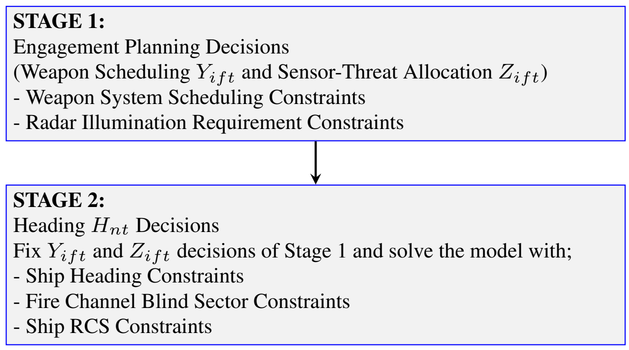


Figure 5.1: Decomposition of the NADP Problem

In Stage 1, the mathematical model solution yields engagement planning decisions (Y_{ift} and Z_{ift}), aiming to eliminate threats with maximum probability. The Z_{ift} variable is not included in the objective function; therefore, in some cases, it can be set to 1 in the solution even if no engagement is planned. Setting the Z_{ift} variable to 1 when it is not necessary can hinder finding the best solution in Stage 2. To prevent this, the Z_{ift} decisions are redefined based on the Y_{ift} decisions in the Stage 1 solution.

In Stage 2, maneuvering decisions (H_{nt}) are determined to ensure that these planned engagements are carried out without being affected by the blind sectors of the weapon/sensor systems and that the ship follows the heading in a way that will show minimum IR/RCS trace to the threat. Note that in this decomposition of the problem, we expect to obtain the optimal solution of the linearized model if there are no fire channel blind sector limitations overlapping with Stage 1 decisions. There may be some cases in which the ship cannot maneuver to its optimal heading due to blind-sector limitations.

Comparison results for the linearized model with decomposition are presented in Table 5.4. This experiment is conducted using the scenarios described in Table 5.1. As can be seen from the results, the decomposition approach can obtain solutions very close to those of the linearized model in terms of the objective function value in a remarkably shorter run time. Considering the scenarios that are completed within the time limit (20 out of 30), the decomposition approach mostly (15 out of 20) obtains the optimal solution. Only in a few scenarios, there is a minor deviation from the optimum due to the non-alignment of Stage 1 decisions with the fire channel blind sector limitations.

Table 5.4: Comparison Results of the Linearized Model with Decomposition

	Linearized Model		Linearized Model with Decomposition (Stage 1 + Stage 2)			
	OBJ-3	Run Time	OBJ-3	Stage 1 Run Time	Stage 2 Run Time	Total Run Time
Scenarios with optimal solution (20 out of 30)	0.2571	52.13	0.2573	3.02	3.25	6.27
Scenarios halted due to the time limit (8 out of 30)	0.1935	504.31	0.1880	94.51	8.63	103.13
All scenarios [a]	0.2390	181.32	0.2375	29.16	4.78	33.95

[a] Except for scenarios in which the linearized model could not reach a feasible solution within the time limit (2 out of 30). The decomposition approach achieved a feasible solution in all the scenarios. There is only one scenario in which the decomposition approach cannot be completed within the specified time limit.

As the scenario size increases, the speed performance of the decomposition approach can be observed. For scenarios in which the linearized model is halted due to the time limit, the decomposition approach is still able to complete within the time limit, thus resulting in better solutions. However, with an average run time of 103.13 seconds for these scenarios, the decomposition approach is not sufficiently fast to be applicable in real-world applications.

Note that the decomposition approach achieves these results in a single iteration. While optimal or near-optimal solutions can be obtained in a single iteration, this approach can be further enhanced by employing methods such as Dantzig-Wolf or

Benders Decomposition. In this case, we need to iterate between Stage 1 and Stage 2 solutions multiple times. However, as observed in the results, the decomposition approach in a single trial already exceeds our acceptable time limits for finding a solution.

At first glance, it may seem that Stage 1 can be further divided into two separate sub-stages in which Y_{ift} and Z_{ift} decisions can be determined sequentially. However, upon closer examination, it becomes apparent that this division is not feasible. Sensor threat allocation Z_{ift} decisions cannot be allocated independently to the initial sub-stage because they are not featured in the objective function. An alternative approach could be to determine the Y_{ift} decisions in the first sub-stage (Stage 1.1) using the weapon system scheduling constraint set. Subsequently, these decisions can be fixed, and the radar illumination requirement constraint set can be used to ascertain the Y_{ift} decisions. However, in this case, it is observed that the maximum number of simultaneous firing channel limits are generally violated during Stage 1.1, leading to the generation of infeasible solutions in Stage 1.2. For instance, in the experiments conducted using the scenarios outlined in Table 5.1, 23 out of 30 scenarios resulted in infeasible solutions. Therefore, it is concluded that in the mathematical model solution, it is necessary to simultaneously determine the Y_{ift} and Z_{ift} decisions in Stage 1.

As a result, these findings underscore the need for a heuristic solution method that can produce solutions as swiftly and effectively as required in real-world applications. To achieve this goal, we draw upon the decomposition approach presented here as a starting point.

5.6 Heuristic Solution Approaches

5.6.1 Solution of the Stage 1 (Engagement Planning) Problem

In this subsection, we start with one fire channel - one threat scenario, which is the simplest version of the NADP problem. Subsequently, studies are continued by increasing the complexity step-by-step as multiple fire channels to one threat, one fire channel to multiple threats, and multiple fire channels to multiple threats.

5.6.1.1 One Fire Channel - One Threat (1x1 NADP) Problem

In this scenario case, there is one threat coming toward a single ship having one weapon/sensor system. The ship can plan its engagement by allocating all its defense resources only to this threat.

Following the detection of the threat, the combat management system first determines all the engagement alternatives (V set) that can be employed against this threat. The decisions to be made at this stage are which of these engagement alternatives will be chosen to eliminate the threat with the highest probability. While making these decisions, constraints such as SLS policy, setup times, illumination requirements, and inventory levels specified in Section 4 are taken into account.

In the solution approach, an engagement network is constructed by considering the problem constraints of these engagement alternatives. The graph created for an example scenario is shown in Figure 5.2. For this scenario, there are 5 different engagement alternatives to be fired between the time periods 3 to 7 against the incoming threat using the weapon system with 2 available inventories. Y_{116} and Y_{117} engagements can also be made after the Y_{113} engagement, while only Y_{117} engagement can be made after the Y_{114} engagement. In the optimal solution of this example scenario, the first engagement is started at time period 3 and the second one is started at time period 7.

The objective function of 1x1 NADP problem is given below:

$$min\left(PK_1 \prod_{t \in T, f \in F} (1 - p_{1ft})^{Y_{1ft}}\right)$$

Note that this equation can represent our four objective function alternatives exactly since there is only one ship and one threat. Taking the logarithm of the objective function does not change the optimal solution:

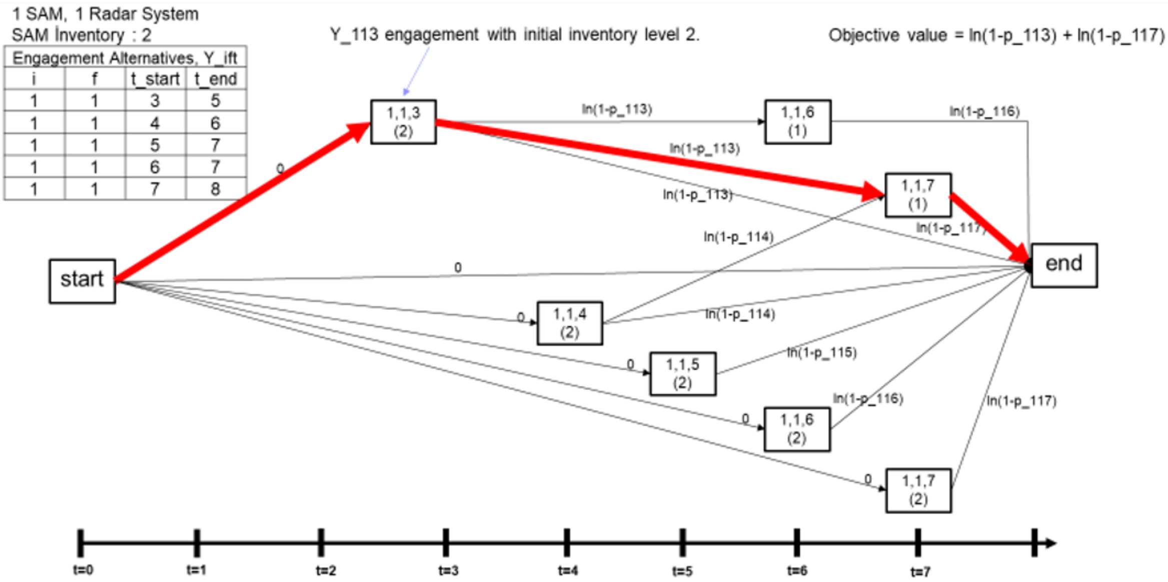


Figure 5.2: Network Representation of an Example 1x1 NADP Engagement Planning Scenario

$$min\left(\underbrace{lnPK_1}_{stage2} + \underbrace{\sum_{t\in T, f\in F} ln(1-p_{1ft})Y_{1ft}}_{stage1}\right)$$

In the engagement graph, each node represents an engagement alternative. Suppose the cost of the edges leaving the nodes is the natural logarithm of the probability of failure of the respective engagement ($ln(1-p_{ift})$). As can be seen in the equation we reached above, when we solve the shortest path problem between the start and end nodes over the network created in this way, we find the solution that minimizes the survival probability of the threat.

Note that, the engagement network is a directed acyclic graph with negative edge weights. Since cost values of the edges are negative, the Bellman-Ford Algorithm [55] can be used to find the shortest path in the network. This solution approach provides the optimal weapon scheduling.

5.6.1.2 Multiple Fire Channels - One Threat (Mx1 NADP) Problem

In the Mx1 NADP scenario case, there are multiple fire channels that can be allocated to one incoming threat. Note that there may be multiple ships or a single ship having multiple weapon/sensor systems in this scenario category. Since there is one incoming threat as in the 1x1 NADP scenario case, the same solution approach (finding the shortest path over the engagement network) explained above can be applied to the Mx1 NADP problem. And this solution approach provides the optimal weapon scheduling also for the Mx1 NADP scenario case.

The graph created for a Mx1 NADP example scenario is shown in Figure-5.3. For this scenario, there are 5 different engagement alternatives to be fired between the time periods 3 to 7 against the incoming threat using the weapon systems with 2 available inventories. Y_{127} engagement can also be made after the Y_{113} engagement, while no additional firing can be made after the Y_{114}, Y_{125}, Y_{126}, and Y_{127} engagements. In the optimal solution of this example scenario, the first engagement is started by fire channel-1 at time period 3 and the second one is started by fire channel-2 at time period 7.

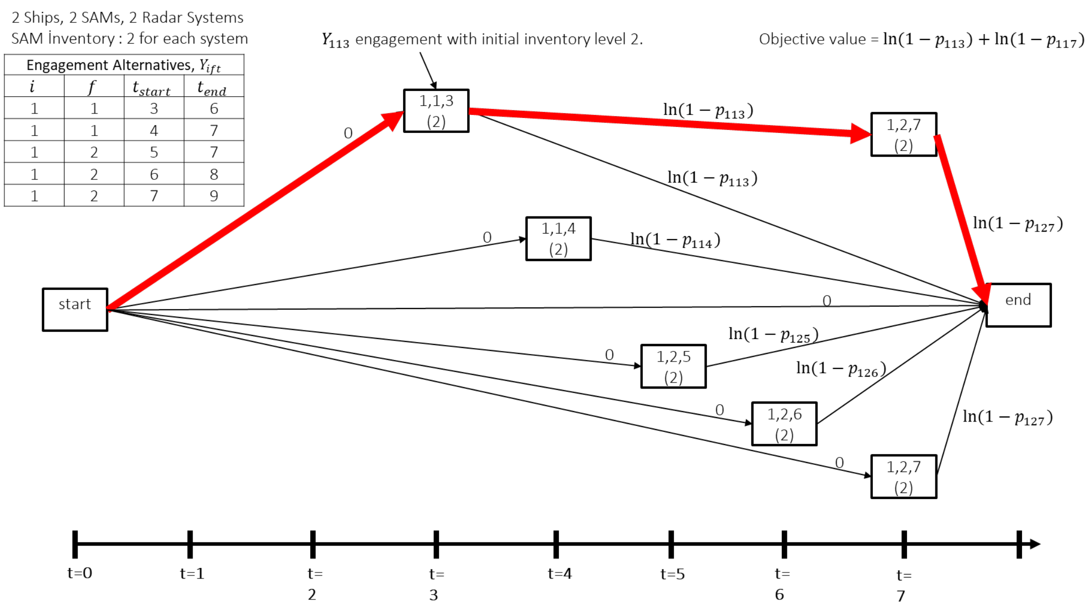


Figure 5.3: Network Representation of an Example Mx1 NADP Engagement Planning Scenario

5.6.1.3 One Fire Channel - Multiple Threats (1xN NADP) Problem

In this case, there is only one fire channel (single ship) and multiple incoming threats. Therefore, one fire channel has to counter multiple threats in the 1xN NADP problem. Different from 1x1 and Mx1 NADP cases, both weapon scheduling (Y_{ift}) and sensor-threat allocation (Z_{ift}) decisions need to be determined in this scenario case.

In a complex problem such as NADP, finding a feasible solution for sensor allocation and weapon scheduling is a challenging process. It is needed to find a solution that satisfies many constraints while exploring a vast solution space. To solve this problem in this scenario case, we developed a genetic algorithm (GA) approach. The GA approach can be highly beneficial in this case as it provides an effective way to search through the solution space and finds a solution that complies with the problem's constraints. By utilizing selection, crossover, and mutation operators, the algorithm can generate and evaluate numerous potential solutions while ensuring they satisfy all the constraints.

To successfully complete an engagement, tracking radars must be assigned to the engaged threats during the required illumination time periods. Considering the problem constraints, the GA is applied to determine sensor-threat allocation decisions. An initial population is generated using the greedy algorithm and candidate solutions are improved using crossover and mutation operators iteratively. The fitness function is used to estimate the objective value of the solution.

A chromosome in this algorithm represents the threat assignments of the sensors for each time period. Figure 5.4 shows the chromosome representation of the GA. In this approach, there are sensors in rows and time periods in columns. Sensors can be allocated to threats for each time period. Note that, this solution approach is developed to address the sensor-threat assignment problem effectively in scenarios with multiple sensors, as well as those with a single sensor.

In the GA crossover operator, the offspring chromosome matrices are generated by mixing the rows of the parent chromosome matrices. For each offspring, a subset of rows from one parent matrix is combined with the remaining rows from the other parent matrix. This generates new sensor-to-threat assignment solutions by preserving

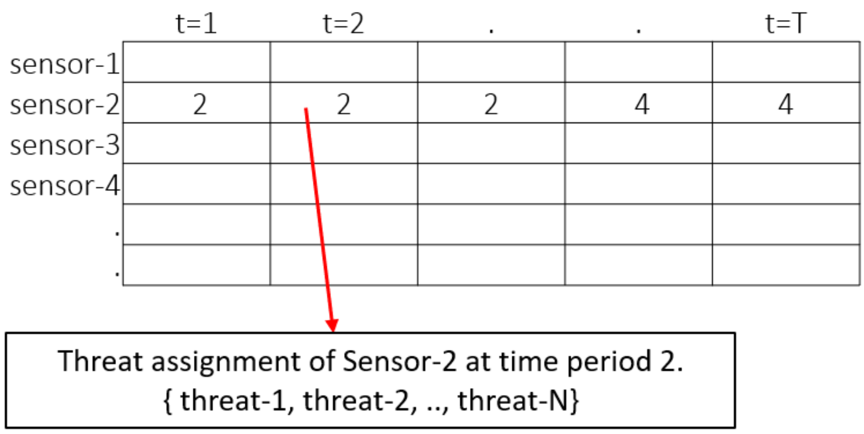


Figure 5.4: Chromosome representation of Sensor-Threat Allocation Solution

the structure. The new offspring inherit the characteristics of both parents, potentially leading to better allocation strategies.

The mutation operator works by altering the sensor-threat allocation decisions represented on the chromosomes. In the mutation process, an engagement alternative is selected randomly. The corresponding sensor-threat allocations are then modified to ensure that the selected engagement becomes part of the alternative engagements considered by the algorithm. This process allows the mutation operator to introduce diversity into the population.

V_i set (engagement alternatives for threat i) for each threat can be determined using the sensor-threat allocations coded in the solution chromosome. In the fitness function, engagements are selected in a similar way as in the greedy algorithm. Threats are prioritized, and the earliest possible engagement option is chosen against the selected threat. This process continues in a loop until there are no remaining engagement alternatives. The objective function value, calculated based on these engagements, serves as the fitness value of the solution.

Finally, the best solution of the GA provides sensor-threat allocation and weapons scheduling decisions for the Stage 1 problem. The pseudocode of the algorithm is given in Algorithm 2. The parameter set used in the GA is given in Appendix E.

Algorithm 2 Stage 1 Genetic Algorithm

Generate initial population

- Use Greedy Algorithm with different k values and generate solutions
- If the initial population size is not provided using the Greedy Algorithm, then generate random solutions (select random engagements and allocate sensors to threats accordingly)

for $n_{iteration} = 1$ to Max Number of Iterations **do**

  for $n_{Crossover} = 1$ to Number of Crossovers **do**

    - Select two solutions randomly from the population
    - Create two offsprings (take allocation plans randomly from parents)
    - For each offspring, change some sensor-threat allocation decisions using the mutation rate
    - Calculate the fitness value of the offsprings
    - Add offsprings to the population if not duplicate

  end for

  - Generate random new solutions and add to the population if not duplicate
  - Sort the individuals according to their fitness values and keep the population size number of individuals
  - If there is no improvement in the best fitness value for the convergence limit number of iterations, then stop.

end for

Fitness Function:

Order the threats ascending in terms of $\frac{HitTime_i}{Value_i}$

for $\forall i \in A$ **do**

  Determine the engagement alternatives (V_i set) using the chromosome (sensor-threat allocation decisions)

  while V_i set is not empty **do**

    - Select the earliest starting engagement in the V_i set and delete conflicting engagements from V and V_i sets considering all problem constraints such as SLS policy, setup times, illumination requirements, and inventory levels.

  end while

end for

- Calculate obj-3 (weighted average hit probability of the threats) value

5.6.1.4 Multiple Fire Channels - Multiple Threats (MxN NADP) Problem

As in the 1xN NADP problem, both sensor-threat allocation and weapon scheduling decisions need to be determined in this scenario case. In the GA solution approach, the chromosome structure is designed to handle scenarios with multiple sensors without requiring additional modifications for a MxN NADP scenario. Therefore, the GA solution approach presented for the 1xN NADP problem can be used for the MxN NADP problem.

5.6.2 Solution of the Stage 2 Problem

In Stage 2 of the algorithm, maneuvering decisions (H_{nt}) are made to minimize the probability of being hit by showing the least IR/RCS trace to the incoming threat, without hindering the planned engagements of the ship due to the blind-sector limitations. A dynamic programming (DP) approach with backward recursion can be used to solve this problem. The heading that the ship will take throughout the scenario time (ts) is determined by calculating the gain value of all heading alternatives backward from the hit time of the threat. The DP formulation for this problem is as follows:

Let $L(\alpha)^t$ be the maximum survival probability of the ship that can be obtained if the ship's heading is α at time period t.

State space, S: {0, 1, 2, ..., 358, 359}

Action space, D: $\{-mr_n, -mr_n + 1, .., mr_n\}$ degree of change in heading.

Backward recursion:

$$L(\alpha)^t = \left(\prod_{i \in A}(1 - HIT_i^t(\alpha))\right) max_{\theta \in D}\left(L(\alpha+\theta)^{t+1}\right) O^t(\alpha) \tag{5.18}$$

$$HIT_i^t(\alpha) = \begin{cases} MISS_i(1 - rcs_{i\alpha}), rt_i = t \\ 1, \text{ otherwise} \end{cases} \tag{5.19}$$

$$MISS_i = \prod_{t \in T, f \in F} (1 - p_{ift})^{Y_{ift}} \tag{5.20}$$

And the boundary condition:

$$L(\alpha)^{ts+1} = 1 \tag{5.21}$$

where rt_i is the hit time of threat i, $rcs_{i\alpha}$ is the miss probability of threat i when the target ship's heading is α immediately before the hit, $MISS_i$ denotes the survival probability of threat i after all engagements, and $HIT_i^t(\alpha)$ is the overall hit probability of threat i if the ship's heading is α at time period t. $O^t(\alpha)$ takes the value of 0 if there is an obstruction in one of the active engagements because of a blind sector when the ship's heading is α at time period t.

When problem stages (Stage 1 and Stage 2) are handled separately in one-threat scenarios, both proposed solution approaches can provide optimal solutions for their own problem cases. However, considering both stages together, the proposed heuristic algorithm does not guarantee an optimal solution. There may be some cases in which the ship cannot maneuver to its optimal heading due to blind-sector limitations. In the engagement planning stage, the blind sector limitations of ship routing are not considered. Therefore, there may be infrequent cases where this algorithm does not reach the optimal solution.

For multiple threat scenarios, an optimal solution cannot be guaranteed in Stage 1 since a heuristic algorithm is being used. However given the engagement decisions, the proposed backward recursion DP algorithm can determine the optimal heading solution for both one-threat and multiple-threats scenarios.

5.6.3 Two and Three-Stage Heuristic Algorithms

Thus, we have explained our solution approaches for the static NADP problem, which we decomposed into two main stages: engagement planning and heading. By utilizing the proposed methods, a three-stage heuristic algorithm ($HA^{GA+EN+DP}$) is created. The pseudo-code of the $HA^{GA+EN+DP}$ algorithm is presented in Algorithm 3.

Algorithm 3 Heuristic Algorithm with Genetic Algorithm + Engagement Network + Dynamic Programming ($HA^{GA+EN+DP}$)

Generate problem parameters
V: set of valid $\{i, f, t\}$ combinations of threat, fire channel and engagement time
Initialize all decision variables as zero

Stage 1.1: Sensor-Threat Allocation Decisions
if $|A| > 1$ **then**
    -Determine Z decision variables using the Genetic Algorithm
    -Determine V_i set according to Z decisions
end if

Stage 1.2: Weapon Scheduling Decisions
Order the threats ascending in terms of $\frac{HitTime_i}{Value_i}$
for $\forall i \in A$ **do**
    -Create Engagement Graph using V_i set
    -Solve Shortest Path Problem using Bellman-Ford Algorithm
    -Determine Y decision variables using the shortest path solution
end for

Stage 2: Heading Decisions
for $\forall n \in N$ **do**
    -Determine H decision variables using a backward recursion DP Approach
end for

As discussed in Section 5.5, dividing Stage 1 into two sub-stages in the mathematical model solution does not prove useful for our purpose. However, in a heuristic solution approach, addressing sensor-threat allocation and weapon scheduling decisions sequentially can yield effective solutions.

After the allocation of the sensors to the threats by the GA, the engagement network approach can be employed to determine weapon scheduling decisions. After determining the Z_{ift} decisions, we can always find a feasible Y_{ift} set, but otherwise, this is not guaranteed. There may not be a feasible Z_{ift} solution for all Y_{ift} decisions. In this algorithm, Stage 1 is decomposed into two sequential sub-stages for the NADP scenarios with multiple threats. Stage 1.1 GA solution approach provides V_i set for each threat by using the sensor-threat assignments of the genetic algorithm's best solution. For the scenario cases with single threat, Stage 1.1 is not needed and weapon scheduling decisions can be determined using Stage 1.2.

In the GA, the sensor-threat allocations and the engagements are being determined using a fitness function that evaluates the objective function value of the solutions. Therefore skipping the engagement network approach in the Stage 1.2 of the algorithm and using the fast engagement planning procedure of the fitness function could be an alternative way to determine the engagement decisions. Thus our third heuristic algorithm for the NADP problem is given in Algorithm 4:

Algorithm 4 Heuristic Algorithm with Genetic Algorithm + Dynamic Programming (HA^{GA+DP})

Generate problem parameters
V: set of valid $\{i, f, t\}$ combinations of threat, fire channel and engagement time
Initialize all decision variables as zero

Stage 1.1: Sensor-Threat Allocation Decisions
- Determine Z decision variables using the Genetic Algorithm
- Determine V_i set according to Z decisions

Stage 1.2: Weapon Scheduling Decisions
- Use the GA fitness value calculation procedure and determine weapon scheduling decisions using V_i set

Stage 2: Heading Decisions
for $\forall n \in N$ **do**
- Determine H decision variables using a backward recursion DP Approach

end for

5.6.4 Performance of the Heuristic Algorithms

We first present the results for 1x1 NADP scenarios, which represent the simplest and the most basic case of the NADP problem. In these scenarios, there is only one incoming threat that needs to be intercepted by one available fire channel. Experimentation results on 50 different scenarios are presented in Table 5.5.

Table 5.5: Comparison of the Heuristic Algorithms in 1 Fire Channel 1 Threat (1x1 NADP) Scenarios

	OBJ-1 Prob. of No Leaker	OBJ-2 Min of Max Hit Prob.	OBJ-3 Value Weighted Threat Hit Prob.	OBJ-4 Value Weighted Ship Surv. Prob.	Run Time (seconds)
	Average Values				
Greedy Algorithm	0.7678	0.2322	0.2322	0.7678	0.16
Linearized Model	0.7755	0.2245	0.2245	0.7755	0.18
Decomposition Approach	0.7755	0.2245	0.2245	0.7755	1.01
$HA^{GA+EN+DP}$	0.7755	0.2245	0.2245	0.7755	0.20
HA^{GA+DP}	0.7669	0.2331	0.2331	0.7669	0.58
	Percent Deviation % From the Best Solution (Average)				
Greedy Algorithm	0.93	10.91	10.91	0.93	
Linearized Model	0.00	0.00	0.00	0.00	
Decomposition Approach	0.00	0.24	0.24	0.00	
$HA^{GA+EN+DP}$	0.00	0.02	0.02	0.00	
HA^{GA+DP}	1.08	10.79	10.79	1.08	
	Number of Best Solutions				
Greedy Algorithm	31	31	31	31	
Linearized Model	50	50	50	50	
Decomposition Approach	48	48	48	48	
$HA^{GA+EN+DP}$	48	48	48	48	
HA^{GA+DP}	37	37	37	37	

The running time limit is taken as 500 seconds for the linearized model and decomposition approach GAMS solutions. As seen in the results, the running times for each model and algorithm are under one second on average. Due to the small size of the scenarios, the GAMS solutions are also very fast. Our analysis reveals that the decomposition approach and the $HA^{GA+EN+DP}$ algorithm provide the best solution in 48 out of the 50 scenarios, with the average objective values being very close to the linearized model GAMS solutions. Since all resources can be allocated to the single incoming threat, Stage 1.1 is not needed for these scenarios. Stage 1.2 can provide optimal engagement planning decisions if Stage 2 heading decisions are not considered. However, since the decomposition approach and $HA^{GA+EN+DP}$ algorithm handle all stages separately, there may be small deviations from the optimal solutions infrequently. They can provide optimal solutions for 1x1 NADP scenarios if there are no fire channel blind sector limitations after the Stage 1 decisions. In this particular experiment, we encountered two scenarios where the blind sector limitations of the fire channels resulted in negligible deviations from the optimal solution. The slightly higher percent deviation in the decomposition approach is due to an approximation error in one of the scenarios. In summary, we can say that the $HA^{GA+EN+DP}$ algorithm mostly achieves optimal results for 1x1 NADP scenario cases.

We also tested the HA^{GA+DP} algorithm, which provides engagement plans using the GA. This solution approach provides the best results in 37 out of 50 scenarios. The average of the objective values is close to the average of the best objective values (with an absolute deviation of only 0.0086). Our findings also reveal that the greedy algorithm produces good solutions for this problem scenario.

The results for the Mx1 NADP scenarios are presented in Table 5.6. In these scenario cases, there is a single threat to be countered by multiple fire channels. Just like in the 1x1 NADP experiment, there is no need to have Stage 1.1 in the $HA^{GA+EN+DP}$ algorithm since all resources can be allocated to a single target.

We observe similar outcomes in the Mx1 NADP scenarios as well. The $HA^{GA+EN+DP}$ algorithm and the decomposition approach can achieve the best results in all scenarios. The HA^{GA+DP} algorithm can also provide close results with a small margin of error.

When it comes to the single fire channel against multiple threat scenarios, the problem starts to become more complex. In this case, time period assignments of the available single weapon-radar option need to be divided among incoming threats. The results of the 1xN NADP scenario experiment are presented in Table 5.7.

The running times of the GAMS solutions for these scenarios start to increase as

Table 5.6: Comparison of the Heuristic Algorithms in Multiple Fire Channels 1 Threat (Mx1 NADP) Scenarios

	OBJ-1 Prob. of No Leaker	OBJ-2 Min of Max Hit Prob.	OBJ-3 Value Weighted Threat Hit Prob.	OBJ-4 Value Weighted Ship Surv. Prob.	Run Time (seconds)
	Average Values				
Greedy Algorithm	0.7853	0.2147	0.2147	0.9221	0.49
Linearized Model	0.8154	0.1846	0.1846	0.9328	0.27
Decomposition Approach	0.8154	0.1846	0.1846	0.9328	0.97
$HA^{GA+EN+DP}$	0.8154	0.1846	0.1846	0.9328	0.74
HA^{GA+DP}	0.8140	0.1860	0.1860	0.9324	1.12
	Percent Deviation % From the Best Solution (Average)				
Greedy Algorithm	3.47	37.50	37.50	1.13	
Linearized Model	0.00	0.00	0.00	0.00	
Decomposition Approach	0.00	0.00	0.00	0.00	
$HA^{GA+EN+DP}$	0.00	0.00	0.00	0.00	
HA^{GA+DP}	0.16	2.46	2.46	0.04	
	Number of Best Solutions				
Greedy Algorithm	21	21	21	21	
Linearized Model	50	50	50	50	
Decomposition Approach	50	50	50	50	
$HA^{GA+EN+DP}$	50	50	50	50	
HA^{GA+DP}	44	44	44	44	

expected. The average run time of the linearized model significantly increases to 48.17 seconds in the 1xN NADP scenarios. The decomposition approach, with an average runtime of 3.49 seconds, achieves the same average objective function value as the linearized model and even better results in terms of average percent deviation. The reason for the decomposition approach outperforming the linearized model on average is due to the fact that in some scenarios, the linearized model fails to reach the optimal solution within the runtime limit of 500 seconds.

It is observed that the heuristic algorithms exhibit a more reasonable increase in execution times. Furthermore, when considering the objective function value, it can be observed that the heuristic algorithms produce results that are close to the results of the GAMS models, with an average deviation of 2.89% for $HA^{GA+EN+DP}$ and

Table 5.7: Comparison of the Heuristic Algorithms in 1 Fire Channel Multiple Threats (1xN NADP) Scenarios

	OBJ-1 Prob. of No Leaker	OBJ-2 Min of Max Hit Prob.	OBJ-3 Value Weighted Threat Hit Prob.	OBJ-4 Value Weighted Ship Surv. Prob.	Run Time (seconds)
	Average Values				
Greedy Algorithm	0.2534	0.5367	0.3939	0.2534	0.25
Linearized Model	0.2927	0.4674	0.3618	0.2927	48.17
Decomposition Approach	0.2934	0.4648	0.3618	0.2934	3.49
$HA^{GA+EN+DP}$	0.2847	0.4725	0.3671	0.2847	1.25
HA^{GA+DP}	0.2839	0.4724	0.3680	0.2839	1.23
	Percent Deviation % From the Best Solution (Average)				
Greedy Algorithm	17.04	23.36	11.62	17.04	
Linearized Model	0.19	2.51	0.14	0.19	
Decomposition Approach	0.22	1.00	0.09	0.22	
$HA^{GA+EN+DP}$	2.11	4.44	2.89	2.11	
HA^{GA+DP}	2.45	4.35	3.38	2.45	
	Number of Best Solutions				
Greedy Algorithm	7	21	6	7	
Linearized Model	46	39	46	46	
Decomposition Approach	42	34	42	42	
$HA^{GA+EN+DP}$	24	34	25	24	
HA^{GA+DP}	23	34	24	23	

3.38% for HA^{GA+DP}. This demonstrates the effectiveness of heuristic algorithms in providing solutions that closely approximate the solutions obtained from the GAMS model.

The results for multiple fire channels and multiple threats scenarios can be seen in Table 5.8. Detailed results are presented in the Appendix F.

The GAMS average running time increases to 174.21 seconds for the linearized model and 61.85 seconds for the decomposition approach, whereas the heuristic algorithm's running time reaches up to approximately 7 seconds. Meanwhile, the Greedy algorithm consistently maintains an average running time of around 1 second. As observed in the results of 1xN NADP, the linearized model obtains slightly worse results than the decomposition approach due to the 500-second runtime limit. It is notewor-

Table 5.8: Comparison of the Heuristic Algorithms in Multiple Fire Channels Multiple Threats (MxN NADP) Scenarios

	OBJ-1 Prob. of No Leaker	OBJ-2 Min of Max Hit Prob.	OBJ-3 Value Weighted Threat Hit Prob.	OBJ-4 Value Weighted Ship Surv. Prob.	Run Time (seconds)
	Average Values				
Greedy Algorithm	0.3209	0.4187	0.2520	0.7017	1.07
Linearized Model	0.4084	0.3172	0.2084	0.7477	174.21
Decomposition Approach	0.4087	0.3139	0.2075	0.7459	61.85
$HA^{GA+EN+DP}$	0.3986	0.3239	0.2122	0.7452	7.39
HA^{GA+DP}	0.3969	0.3243	0.2135	0.7442	7.34
$HA^{GA+EN+DP}$ (1 sec limit)	0.3892	0.3314	0.2184	0.7362	2.33
	Percent Deviation % From the Best Solution (Average)				
Greedy Algorithm	28.32	43.19	29.55	7.63	
Linearized Model	1.25	1.13	0.88	0.34	
Decomposition Approach	0.88	1.96	0.18	0.13	
$HA^{GA+EN+DP}$	4.59	4.66	3.26	0.64	
HA^{GA+DP}	5.19	4.81	4.01	0.78	
$HA^{GA+EN+DP}$ (1 sec limit)	9.11	6.29	7.31	1.36	
	Number of Best Solutions				
Greedy Algorithm	6	15	6	6	
Linearized Model	33	38	31	31	
Decomposition Approach	38	39	41	39	
$HA^{GA+EN+DP}$	22	34	22	22	
HA^{GA+DP}	19	33	19	20	
$HA^{GA+EN+DP}$ (1 sec limit)	16	29	16	16	

thy that the objective function values between the heuristic algorithms and the GAMS solutions do not exhibit a notable difference, indicating that close solutions can be obtained in a much shorter time frame using heuristic algorithms.

Additionally, we wanted to assess the performance of the $HA^{GA+EN+DP}$ algorithm when limiting Stage 1.1 to 1 second in this experiment. As the scenario size increases, it may be necessary to restrict the runtime of the heuristic algorithms. The iterations of the GA in Stage 1.1 can be dynamically constrained based on the threat status and urgency of countermeasures. According to the results, when Stage 1.1 is constrained to 1 second, the $HA^{GA+EN+DP}$ algorithm produces solutions with an average percent deviation of 7.31 in average runtime of 2.33 seconds.

These results highlights the efficiency and effectiveness of heuristic algorithms in quickly generating comparable solutions without compromising solution quality for the NADP problem. Therefore, the small deviation observed in the objective function values further emphasizes the value of heuristic algorithms as valuable tools for addressing and solving this complex problem.

CHAPTER 6

DYNAMIC NADP PROBLEM

6.1 Introduction

In the dynamic NADP problem, which represents the war environment more realistically than the static version, the need for the first engagement planning arises when the first threat is reported, while the friendly units are in a zero-threat environment. In this study, it is assumed that all friendly units have the capability of full coordination and data sharing. In other words, it is accepted that the detection of a threat by a sensor of a platform is instantly learned by all other platforms in the task group via the link.

After the detection of the first threat and the generation of an engagement plan against this threat, the solution needs to be updated in the case of any change in the war environment. These are the situations that may require an update in the current assignment and schedule plan:

- Detection of a new threat
- Miss/destruction of the threat after a weapon-threat engagement
- Disappearance of a threat
- Destruction of a friendly unit
- Breakdown of a weapon/radar system
- Change in the direction of a threat
- Change in other problem parameters

When a new situation that requires a solution update is encountered, the engagements in the current plan that need to be executed immediately or in a very short call are fixed, and the solution is updated accordingly.

Unlike the static NADP problem, in a dynamic problem, it is assumed that if a friendly ship is hit during the scenario, the weapon/sensor systems onboard become unusable, better reflecting the actual situation. This assumption acknowledges that the functionality of weapon and sensor systems may be compromised if a friendly ship is targeted and hit within the scenario. This dynamic aspect introduces a more practical perspective, emphasizing the evolving nature of the naval air defense environment. In contrast, the static problem overlooks these implications, assuming a constant state of system readiness, regardless of the unfolding events within the scenario. The static problem considers all threats and plans for the entire scenario period under the assumption that weapon/sensor systems will never be hit.

In the static problem, consideration of ship damage is impractical because these scenarios depict a single moment in time, lacking the evolution of events. Thus, the dynamic model provides a more comprehensive and realistic representation of the challenges faced in the naval air defense planning. For example, if the first arriving threat hits its target, the ship will be out of combat, and any engagements that can be made by the ship will be canceled.

To gain a better understanding of the dynamic NADP problem, we can examine Figure 6.1. In this representation, the development of events and engagement planning processes can be observed second by second, providing detailed insight into the dynamics of the scenario.

The deviation from the current plan may not be taken into account since the new engagement plan is generated using only the possible engagement alternatives. Therefore, in this work, a bi-objective version of this problem is not studied. Our effectiveness objective function is used as a single objective function in the model. Feasible planning is made by taking into account all problem limitations such as weapon setup time and radar illumination requirements. Since threat ASM systems can move very fast, counter-engagement must be initiated without delay. Therefore, renewed decisions must be determined in a feasible time frame to have an applicable decision

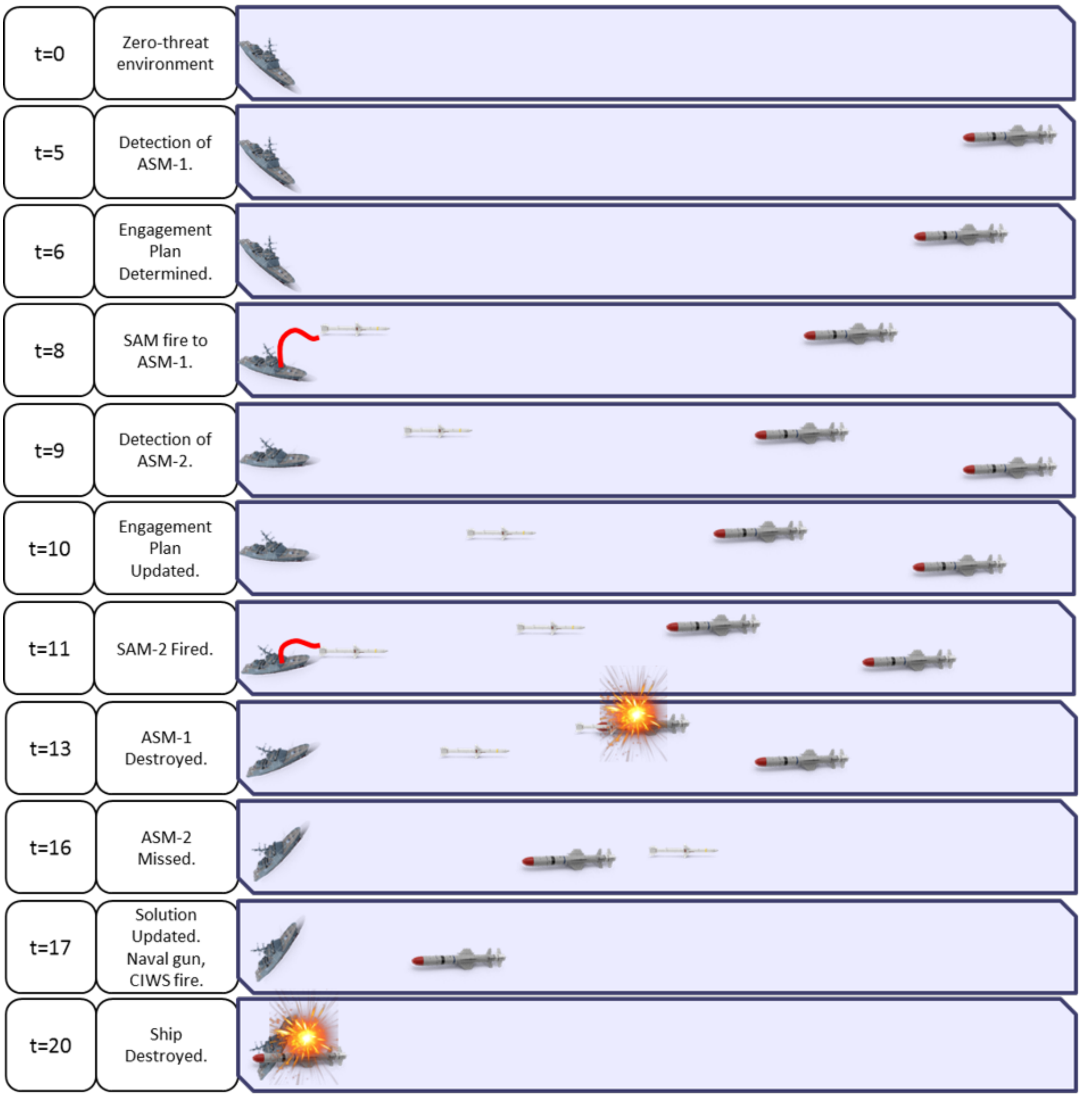


Figure 6.1: Example Development of Events in Dynamic NADP Problem

support.

6.2 Mathematical Formulation of the Dynamic NADP Problem

In this section, our aim is to elucidate the dynamic NADP problem by presenting the mathematical formulation. Thus, differences compared to the static problem formulation can be highlighted. As the first modification, it is needed to track the status of platforms' survival throughout the scenario's progression within the dynamic problem formulation. Therefore, the inclusion of the following additional decision variable and constraint set in the formulation is necessary.

Additional Decision Variable:

PS_{ift} = Probability of the platform, on which the firing channel f is located, not being damaged until the time period that is required to complete the engagement {i,f,t}

Additional Constraint Set:

$$PS_{ift} = \prod_{\substack{i \in A : \{f \in F_n : tg_i = n, \\ rt_i <= max(\rho) : \rho \in U_{ift}\}}} \left(1 - PK_i \prod_{t' \in T, f' \in F} (1 - p_{if't'} PS_{if't'})^{Y_{if't'}}\right),$$

$$\forall (i, f) \in G, t \in T \quad (6.1)$$

This constraint set calculates the survival probability of the platform until the latest time required to complete the i,f,t engagement. To achieve this, the miss probabilities of all threats expected to be on the platform until the last illumination time of the engagement i,f,t are multiplied. The hit probabilities of the i,f,t engagements are updated by multiplying them with the survival probabilities of the engaging platform.

Update in the Objective Function (OBJ-3)

$$min \sum_{i \in A} v_i \left(PK_i \prod_{t \in T, f \in F} (1 - p_{ift} PS_{ift})^{Y_{ift}}\right)$$

The modification in the objective function-3 can be seen above. For engagements to be carried out successfully, the platform must not be hit by any threat until the last required illumination time for the engaging weapon. Therefore, the single shot kill probability of the engagements is multiplied by the probability of the platforms surviving to carry out these engagements.

Note that, at any point in the scenario progression, previously given and immutable engagement decisions (such as missiles already in flight or engagements that have begun the firing process) can be accounted for in the mathematical formulation by fixing the relevant decision variables. These previously fixed decisions are also considered in the calculation of the platforms' survival probabilities.

The introduction of a new decision variable, additional constraint set and modified objective function in the dynamic problem significantly increases the complexity of the mathematical formulation. Due to the infeasibility of solving this formulation within a reasonable timeframe, the mathematical formulation solution is not employed in the computational experiments conducted for the dynamic problem.

6.3 Simulation Structure for Dynamic NADP Problem

A simulation structure is developed to test the algorithms for the dynamic NADP problem. From the initiation of a scenario until the time on target of the last threat, all time points of events are visited, and the progression of the naval warfare environment is simulated. The steps of the simulation structure are as follows:

Algorithm 5 Dynamic NADP Simulation Steps

```
for run = 1 to n_max_iteration do
    Generate problem parameters
    E: Events list that creates a change in the problem scenario
    E ← {}
    Add threat detection times to E
    Add threat ship hit times to E
    V: List of possible engagement alternatives
    V ← {}
    for t = 1, 2, .., ts do
        for each event ∈ E at time t do
            if Event: New Threat detection then
                Update V
            else if Event: Weapon-Threat Encounter then
                Determine if the threat is destroyed by the weapon
                Update V, E
            else if Event: Threat-Ship Hit then
                Determine if the ship is hit by the threat
                Update V, E
            else if Event: Breakdown of a Radar/Weapon System then
                Update V, E
            else if Event: Threat Route Change or Threat Disappearance then
                Update V, E
            else if Event: Weapon Engagement Start against a Threat then
                Update E
            end if
        end for
        if V set is updated then
            Solve NADP problem and determine new solution
            Update E
        end if
        Update the threat/SAM positions and ship headings
    end for
    Calculate obj3 value of the run
    Calculate CI of the obj3 values
      CI = mean(obj3) ± tscore_{α/2, t−1} · σ(obj3)/√t
    if (CI_upper − CI_lower)/2 <= 0.05 then
        Exit the runs
    end if
end for
```

$$CI = \overline{obj3} \pm tscore_{\frac{\alpha}{2},t-1}\frac{\sigma(obj3)}{\sqrt{t}}$$

In the simulation of the dynamic NADP problem, the V and E lists are updated, if needed, at each step. If the V list is updated in response to a new situation, the engagement plan must be updated accordingly. Consequently, a new solution is determined at the current time based on the available data, and the new engagement plan continues to be implemented over time. Events such as the breakdown of radar/weapon systems, threat disappearance, or threat route changes are not considered in the simulation analysis within the scope of this study. Their occurrences can be simulated using predetermined probabilities. However, introducing stochastic parameters for these events increases variability of the results used to determine the success of the algorithms. Therefore, these events never occur in the experiments of this study.

Additionally, the retargeting capability, which is not considered in the static problem, can be modeled in the dynamic problem. This means that if an airborne SAM's target is neutralized before impact, the SAM can be redirected to other threats according to its current trajectory and speed. Retargeting can be introduced as a new engagement alternative in the updated scenario. However, due to the reasons outlined above, the retargeting capability is not considered in the experiments.

The expected objective function value achievable using the solution algorithm for a given scenario can be obtained by evaluating all possible situations that may occur in the scenario. Starting from time $t = 0$, the expected objective function value can be computed by considering all possible situations through the decision tree analysis approach. However, this approach is suitable for small-scale problems because the probability space expands with the size of the problem, making it time-consuming to find the expected value for larger scenarios. Therefore, in our analysis, while calculating the expected value for small problems, we employed a Monte Carlo simulation approach for larger problems using multiple runs to estimate the average objective function value.

In Monte Carlo simulation scenarios, runs are carried out for each scenario until the mean of the objective function values fall within the $+ - 0.05$ interval for a 95% confidence level. The number of runs for each scenario is limited to 100.

Additionally, to track the development of events within the scenario simulation and to validate the developed simulation environment, a visualization tool is implemented using MATLAB plotting tools. Figure 6.2 displays a snapshot of a scenario using this visualization tool.

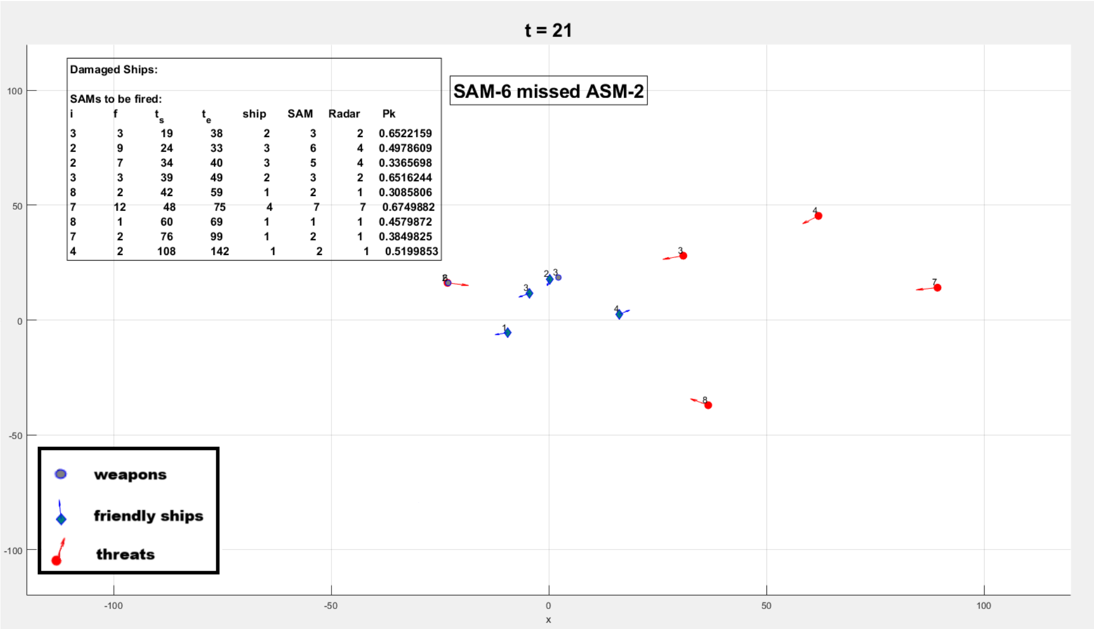


Figure 6.2: Snapshot of a Scenario in MATLAB

6.4 Heuristic Solution Approaches for the Dynamic NADP Problem

6.4.1 $HA^{GA+EN+DP}$ Algorithm

In the dynamic NADP environment, each new situation can be handled as a static problem with existing information at hand. Therefore the solution approaches developed for the static problem can also be used to obtain a solution for the dynamic NADP problem.

The first solution approach that can be employed for the Dynamic NADP problem may be the $HA^{GA+EN+DP}$ algorithm, which is developed for the Static NADP problem, explained in Section 5.6.3. Since the $HA^{GA+EN+DP}$ algorithm provides more successful results than the HA^{GA+DP} algorithm and there is no significant difference in runtime, the HA^{GA+DP} algorithm is not used in the experiments conducted for the dynamic problem. Additionally, because the experiments require a very long runtime, the linearized models (GAMS solutions) are not used in the dynamic problem experiments. As a straightforward approach, the Greedy algorithm is considered among the alternative solution approaches in the experiments conducted in this section.

6.4.2 $DHA^{GA+EN+DP}$ Algorithm

As previously elucidated in the preceding sections, the most significant contribution of this study to the literature is the mathematical definition of the NADP problem, its decomposition for more effective solutions, and the development of a three-stage heuristic algorithm based on this decomposition. These developments made on the static problem are largely applicable to its dynamic counterpart. In this context, the $HA^{GA+EN+DP}$ algorithm proposed for the static problem is modified to achieve better results in the dynamic scenarios. Referred to as the $DHA^{GA+EN+DP}$ (dynamic version of $HA^{GA+EN+DP}$) algorithm, this algorithm incorporates the necessary adjustments, as outlined in Section 6.1 and 6.2, to consider the survival states of the ships within the scenario. The pseudocode of the $DHA^{GA+EN+DP}$ algorithm is given in Algorithm 6:

In this approach, the decisions made during the solution process consider the subsequent effects of being hit by threats. In the genetic algorithm of Stage 1, the survival probabilities of the ships are considered when calculating the fitness value of the solutions. As engagement plans are created for each threat, the kill probabilities of subsequent engagements from the targeted ship are updated based on the survival probability of that ship. The pseudocode for the modified fitness function calculation procedure is provided in Algorithm 7:

Thus, compared with the static version, the modified version of the $HA^{GA+EN+DP}$ algorithm is expected to provide more successful engagement decisions in a dynamic environment.

6.4.3 MHA^{EN+DP} Algorithm

The $DHA^{GA+EN+DP}$ algorithm encompasses the three-stage solution approach developed for this problem. In scenarios without time stringency, it is expected to provide the best solutions. However, particularly for large-scale problems, the so-

Algorithm 6 Dynamic Heuristic Algorithm with Genetic Algorithm + Engagement Network + Dynamic Programming ($DHA^{GA+EN+DP}$)

Generate problem parameters
V: set of valid $\{i, f, t\}$ combinations of threat, fire channel and engagement time
Initialize all decision variables as zero

Stage 1.1: Sensor-Threat Allocation Decisions
if $|A| > 1$ **then**
  -Determine Z decision variables using the Genetic Algorithm with modified fitness function
  -Determine V_i set according to Z decisions
end if

Stage 1.2: Weapon Scheduling Decisions
Order the threats ascending in terms of $\frac{HitTime_i}{Value_i}$
for $\forall i \in A$ **do**
  -Create Engagement Graph using V_i set
  -Solve Shortest Path Problem using Bellman-Ford Algorithm
  -Determine Y decision variables using the shortest path solution
  -Update p_{ift} single shot kill probabilities of the engagement alternatives that can be fired from the target ship of threat i after $HitTime_i$ according to the hit probability of the threat i.
end for

Stage 2: Heading Decisions
for $\forall n \in N$ **do**
  -Determine H decision variables using a backward recursion DP Approach
end for

lution time of the genetic algorithm approach, where defense resources are allocated to threats, can be excessively high and may exceed our practical usage limits. Therefore, to have a solution approach that can generate faster solutions for large-scale problems, the MHA^{EN+DP} algorithm is developed.

The pseudocode of the MHA^{EN+DP} algorithm is presented in Algorithm 8:

In this algorithm, incoming threats are addressed one by one based on their priority, determined by the value ($Value_i$) and the time on target ($HitTime_i$) of the threats.

Algorithm 7 Fitness Function of Stage 1 Genetic Algorithm for Dynamic NADP

<u>Fitness Function:</u>

-Order the threats ascending in terms of $\frac{HitTime_i}{Value_i}$

for $\forall i \in A$ **do**

-Determine the engagement alternatives (V_i set) using the chromosome (sensor-threat allocation decisions)

while V_i set is not empty **do**

- Select the earliest starting engagement in the V_i set and delete conflicting engagements from V and V_i sets considering all problem constraints such as SLS policy, setup times, illumination requirements, and inventory levels.

end while

-Update p_{ift} single shot kill probabilities of the engagement alternatives that can be fired from the target ship of threat i after $HitTime_i$ according to the hit probability of the threat i.

end for

- Calculate obj-3 (weighted average hit probability of the threats) value

Accordingly, the threats that are close in time to hitting the target and have higher values are prioritized as the top-priority threats in the algorithm. Later in Section 6.7, threat prioritization approach using machine learning models is proposed and compared. An engagement plan is sequentially formulated for each threat by considering all resources available on friendly ships. The earliest engagement in the generated plan is selected, and a similar planning process is conducted for the next prioritized threat. The algorithm results in an engagement plan that consists of one engagement for each threat. New solutions are generated based on hit or miss outcomes in subsequent events.

Compared with the $DHA^{GA+EN+DP}$ algorithm, this algorithm employs a myopic approach, focusing on the top-priority threat. Genetic algorithm approach in Stage 1.1 is not utilized in this algorithm. Sequentially, for each threat in the priority order, an engagement network is constructed, and the engagement plan is determined. The earliest engagement in the solution is then selected. In contrast, the $DHA^{GA+EN+DP}$ algorithm considers all threats and plans for the entire scenario period.

The major advantage of this approach is its ability to provide much faster solutions compared to the 3-stage approach. This ensures an effective solution that can be

Algorithm 8 Myopic Heuristic Algorithm with Engagement Network + Dynamic Programming (MHA^{EN+DP})

Generate/Update problem parameters

V: set of all valid $\{i, f, t\}$ engagement combinations of threat, fire channel and engagement start time

Stage-1: Engagement Planning Decisions

Order the threats ascending in terms of $\frac{HitTime_i}{Value_i}$

for $\forall i \in A$ **do**

  If there is no active engagement going towards ASM i

    -$V_i \in V$: set of valid engagements against ASM i

    If the previously planned engagement against ASM i is in set V_i

      -Select the previously planned engagement

    else

      -Create Engagement Graph using V_i set

      -Solve Shortest Path Problem using Bellman-Ford Algorithm

      -Select the first engagement for ASM i using the shortest path solution

    end if

      -Delete conflicting engagement alternatives with the selected engagement from the V set

  end if

  -Update p_{ift} single shot kill probabilities of the engagement alternatives that can be fired from the target ship of threat i after $HitTime_i$ according to the hit probability of the threat i.

end for

Stage-2: Heading Decisions

for $\forall n \in N$ **do**

  -Determine H decision variables using a backward recursion DP Approach

end for

utilized in scenarios with limited time for rapid decision making.

However, the non-allocation of resources to targets can result in inferior results in cases with limited resources compared with the $DHA^{GA+EN+DP}$ algorithm. In this approach, when planning for one threat, it is possible to select a weapon/sensor that is the sole alternative for another threat. In such cases, the number of available engagement alternatives may be limited in the subsequent planning phases, and consequently, a relatively worse objective function value may be obtained.

On the other hand, generating engagement plans for targets sequentially and selecting the earliest engagement in this plan allows for quicker reactions (not in terms of the solution run-times, but about selecting early starting engagements) in some cases compared to the $DHA^{GA+EN+DP}$ algorithm. This provides the opportunity to reserve more engagement alternatives for future emerging threats. In the $DHA^{GA+EN+DP}$ solution approach, a plan for the known targets is generated throughout the entire scenario duration. However, potential threats that may emerge later are not considered. In some situations, there may be instances where later engagement alternatives with higher probabilities of success may be selected to achieve the best objective function value. In this case, the probability of countering a subsequently emerging threat may be reduced. In the next section, we examine the experimental results related to these aspects.

6.5 Computational Experiments for the Dynamic NADP Problem

6.5.1 Performance of the Solution Approaches in Small-Mid Size Scenarios

The first experiment for the Dynamic NADP problem was carried out on 20 scenarios that have appropriate parameter sizes for expected value calculation. The scenarios involve 1-5 ships and 2-5 threats, and their parameters are generated using the parameter generation structure detailed in Appendix D. By evaluating all possible outcomes in the generated scenarios, expected objective function values are computed. The primary aim of this experiment is to investigate the performances of solution approaches on results where the confidence interval is zero. Additionally, algorithm run-times for each stage in the scenarios are recorded.

Table 6.1 and 6.2 present the average OBJ-3 objective function values for the scenarios and the average percent deviations from the best solution. In addition to the static/dynamic three-stage (GA+EN+DP) and myopic approaches described in Section 6.4, the Greedy Algorithm mentioned in Section 5.2 is also tested.

The results in Table 6.1 demonstrate that modifying the three-stage algorithm for a dy-

Table 6.1: Dynamic NADP Computational Experiments on Small-Mid Size Scenarios-1 (Expected Values)

			Average OBJ-3 Value Weighted Threat Hit Probability		**Average Percent Deviation % From the Best Solution**	
	Number of Ships	**Number of Threats**	$HA^{GA+EN+DP}$	$DHA^{GA+EN+DP}$	$HA^{GA+EN+DP}$	$DHA^{GA+EN+DP}$
S-1	1	2	0.148	0.148	0.00%	0.00%
S-2	1	3	0.526	0.412	27.64%	0.00%
S-3	1	4	0.626	0.604	3.54%	0.00%
S-4	2	2	0.172	0.172	0.00%	0.00%
S-5	2	3	0.415	0.415	0.00%	0.00%
S-6	2	3	0.298	0.255	16.65%	0.00%
S-7	2	4	0.346	0.346	0.00%	0.00%
S-8	2	5	0.113	0.108	4.74%	0.00%
S-9	3	2	0.120	0.120	0.00%	0.00%
S-10	3	3	0.224	0.224	0.00%	0.00%
S-11	3	4	0.498	0.498	0.00%	0.00%
S-12	3	4	0.409	0.406	0.69%	0.00%
S-13	3	5	0.447	0.439	1.84%	0.00%
S-14	4	2	0.104	0.095	9.22%	0.00%
S-15	4	3	0.106	0.106	0.00%	0.00%
S-16	4	4	0.168	0.168	0.12%	0.00%
S-17	4	5	0.165	0.165	0.00%	0.00%
S-18	5	3	0.230	0.204	13.01%	0.00%
S-19	5	4	0.073	0.076	0.00%	3.97%
S-20	5	5	0.153	0.152	0.59%	0.00%
Average OBJ-3 Value			**0.2670**	**0.2556**	**3.90%**	**0.20%**
Number of Best Solution			**10**	**19**		
Average Run Time (s)			**1.83**	**1.92**		

namic environment has enhanced its performance. The $DHA^{GA+EN+DP}$ algorithm achieved an average OBJ-3 value of 0.2556, providing the best solutions in 19 out of 20 scenarios. In comparison, the $HA^{GA+EN+DP}$ algorithm produced solutions with an average OBJ-3 value of 0.2670, deviating by 3.90% from the best solutions on average. These results indicate that the dynamic version, $DHA^{GA+EN+DP}$, is more effective in making engagement decisions in dynamic environment compared to its static counterpart, $HA^{GA+EN+DP}$.

Further, Table 6.2 illustrates the comparison of the $DHA^{GA+EN+DP}$ algorithm with the myopic solution approach and the greedy algorithm. For the 20 scenarios tested, the $DHA^{GA+EN+DP}$ algorithm achieved the best solution in 18 instances. The MHA^{EN+DP} algorithm achieved the best solution in 11 scenarios, with an average OBJ-3 value of 0.2591 and an average deviation of 3.52% from the best solutions. The Greedy al-

Table 6.2: Dynamic NADP Computational Experiments on Small-Mid Size Scenarios-1 (Expected Values)

			Average OBJ-3 Value Weighted Threat Hit Probability			**Average Percent Deviation % From the Best Solution**		
	Number of Ships	**Number of Threats**	$DHA^{GA+EN+DP}$	MHA^{EN+DP}	**Greedy Algorithm**	$DHA^{GA+EN+DP}$	MHA^{EN+DP}	**Greedy Algorithm**
S-1	1	2	0.148	0.148	0.149	0.00%	0.00%	0.81%
S-2	1	3	0.412	0.412	0.482	0.00%	0.00%	16.76%
S-3	1	4	0.604	0.605	0.613	0.00%	0.05%	1.37%
S-4	2	2	0.172	0.172	0.195	0.00%	0.00%	13.57%
S-5	2	3	0.415	0.415	0.415	0.00%	0.00%	0.00%
S-6	2	3	0.255	0.274	0.314	0.00%	7.45%	23.00%
S-7	2	4	0.346	0.346	0.395	0.00%	0.00%	13.98%
S-8	2	5	0.108	0.151	0.145	0.00%	40.61%	34.39%
S-9	3	2	0.120	0.120	0.120	0.00%	0.00%	0.00%
S-10	3	3	0.224	0.224	0.224	0.00%	0.00%	0.18%
S-11	3	4	0.498	0.498	0.510	0.00%	0.00%	2.53%
S-12	3	4	0.406	0.406	0.409	0.00%	0.00%	0.69%
S-13	3	5	0.439	0.442	0.488	0.00%	0.59%	11.11%
S-14	4	2	0.095	0.098	0.137	0.00%	3.14%	43.82%
S-15	4	3	0.106	0.088	0.117	19.34%	0.00%	32.58%
S-16	4	4	0.168	0.167	0.189	0.72%	0.00%	13.48%
S-17	4	5	0.165	0.169	0.207	0.00%	2.00%	24.98%
S-18	5	3	0.204	0.204	0.215	0.00%	0.29%	5.65%
S-19	5	4	0.076	0.085	0.122	0.00%	11.59%	60.74%
S-20	5	5	0.152	0.159	0.222	0.00%	4.75%	46.41%
Average OBJ-3 Value			**0.2556**	**0.2591**	**0.2833**	**1.00%**	**3.52%**	**17.30%**
Number of Best Solution			**18**	**11**	**2**			
Average Run Time (s)			**1.92**	**0.68**	**0.66**			

gorithm, however, yielded solutions with an average OBJ-3 value of 0.2833 and an average deviation of 17.30% from the best solutions.

Note that, despite being the most comprehensive algorithm, since it is a heuristic algorithm, there may be occasional minor deviations from the best solution. Additionally, it is observed that the MHA^{EN+DP} algorithm performs better than the $HA^{GA+EN+DP}$, and it is slightly inferior to the $DHA^{GA+EN+DP}$.

Examining the solution times, while the $DHA^{GA+EN+DP}$ algorithm provides solutions in an average of 1.92 seconds, the MHA^{EN+DP} algorithm offers faster solutions, approximately in 0.68 seconds, which is close to the run-time of the Greedy algorithm.

6.5.2 Performance of the Solution Approaches in Large Size Scenarios

In this experiment, we aim to observe the performance of $DHA^{GA+EN+DP}$ and MHA^{EN+DP} algorithms in large-size scenarios. Due to the large scenario sizes and large number of possible outcomes from the beginning to the end of each scenario, calculating the expected value of the objective function can take long time. Therefore, in this experiment, multiple runs are conducted for each scenario, and at each decision point containing different possibilities, a random number is generated to determine the flow of events (such as hit or miss). Using the dynamic NADP simulation environment, runs are carried out for each scenario until the mean of the OBJ-3 values is within the ± 0.05 intervals for a 95% confidence interval. The minimum number of runs is set to 50. Since the scenario sizes are large, it is observed that the variances of the results are small. Thus, it is seen that the desired confidence interval is reached in all scenarios within 50 runs. In this experiment, a maximum time of 5 seconds has been set for the iterations of the genetic algorithm.

When examining the results presented in Table 6.3, it is observed that the $DHA^{GA+EN+DP}$ algorithm produces solutions with an average OBJ-3 objective function value of 0.278 in a total run time of 5.89 seconds. The MHA^{EN+DP} algorithm, on the other hand, reaches solutions with an average objective function value of 0.306 within an average run time of 2.57 seconds. Additionally, the average percent deviation from the best solution is observed to be 1.39% for the $DHA^{GA+EN+DP}$ algorithm and 12.75% for the MHA^{EN+DP} algorithm.

It can be said that the $DHA^{GA+EN+DP}$ algorithm is more successful than the MHA^{EN+DP} algorithm in terms of the average objective function values in these scenarios. However, when examining the run times, it is noticed that the solution times of the $DHA^{GA+EN+DP}$ algorithm may not be practical. For a real combat environment where rapid decisions are crucial, an average solution time of 5.89 seconds may be considered an excessively long duration. Detailed results on solution times are presented in Table 6.4:

In Table 6.4, the average and maximum duration values for each sub-stage in algorithm solutions are presented. When we look at the $DHA^{GA+EN+DP}$ Algorithm, for Stage 1.1, while the average time required to generate the initial population of the

Table 6.3: Dynamic NADP Computational Experiments on Large Size Scenarios

	Number of Ships	Number of Threats	$DHA^{GA+EN+DP}$						MHA^{EN+DP}					Average Percent Deviation % From the Best Solution	
			OBJ-3	**Number of Runs**	**CI**	**Run Time Stage 1.1**	**Run Time Stage 1.2**	**Run Time Stage 2**	**OBJ-3**	**Number of Runs**	**CI**	**Run Time Stage 1**	**Run Time Stage 2**	$DHA^{GA+EN+DP}$	MHA^{EN+DP}
S-1	5	10	0.235	50.00	0.043	1.97	0.00	2.03	0.256	50.00	0.038	0.01	2.11	0.00%	9.07%
S-2	5	12	0.167	50.00	0.041	3.24	0.00	1.79	0.229	50.00	0.042	0.09	1.93	0.00%	37.13%
S-3	5	15	0.450	50.00	0.043	2.89	0.00	1.07	0.442	50.00	0.041	0.03	1.30	1.83%	0.00%
S-4	6	12	0.159	50.00	0.040	3.89	0.01	1.65	0.143	50.00	0.035	0.18	2.00	11.40%	0.00%
S-5	6	15	0.238	50.00	0.039	5.00	0.04	2.06	0.237	50.00	0.038	0.26	2.09	0.68%	0.00%
S-6	6	18	0.396	50.00	0.040	3.78	0.00	1.88	0.418	50.00	0.036	0.06	2.01	0.00%	5.55%
S-7	7	14	0.245	50.00	0.034	2.95	0.00	3.39	0.321	50.00	0.041	0.03	3.56	0.00%	30.92%
S-8	7	21	0.372	50.00	0.035	3.30	0.00	2.28	0.407	50.00	0.034	0.01	2.31	0.00%	9.41%
S-9	8	16	0.219	50.00	0.032	4.01	0.01	3.15	0.262	50.00	0.037	0.40	3.18	0.00%	19.55%
S-10	8	24	0.297	50.00	0.033	5.59	0.00	3.91	0.344	50.00	0.039	0.05	4.07	0.00%	15.91%
Average			**0.278**	**50.00**	**0.038**	**3.66**	**0.01**	**2.32**	**0.306**	**50.00**	**0.038**	**0.11**	**2.46**	**1.39%**	**12.75%**

Table 6.4: Dynamic NADP Computational Experiments on Large Size Scenarios (Detailed Run Times)

	Number of Ships	Number of Threats	$DHA^{GA+EN+DP}$										MHA^{EN+DP}					
			Stage 1.1 GA Generate Initial Population		**Stage 1.1 GA Iterations**		**Stage 1.2 Construct EN for Each Threat**		**Stage 1.2 Solve SP for Each Threat**		**Stage 2 Solve DP for Each Ship**		**Stage 1 Construct EN for Each Threat**		**Stage 1 Solve SP for Each Threat**		**Stage 2 Solve DP for Each Ship**	
			Mean	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**	**Mean**	**Max**
S-1	5	10	0.56	0.92	1.46	3.23	0.00	0.00	0.00	0.01	0.41	0.67	0.01	0.09	0.00	0.02	0.42	0.60
S-2	5	12	0.83	1.97	2.46	4.21	0.00	0.01	0.00	0.02	0.36	0.55	0.06	1.05	0.03	0.44	0.39	0.62
S-3	5	15	0.84	2.73	2.04	3.99	0.00	0.00	0.00	0.02	0.21	0.39	0.01	0.09	0.01	0.08	0.26	0.50
S-4	6	12	1.75	3.62	2.30	4.13	0.00	0.15	0.00	0.06	0.27	0.46	0.13	1.39	0.06	0.54	0.33	0.63
S-5	6	15	3.38	6.95	1.74	4.05	0.02	0.99	0.01	0.30	0.34	0.51	0.14	3.50	0.05	1.06	0.35	0.55
S-6	6	18	1.78	4.00	2.01	4.10	0.00	0.03	0.00	0.02	0.31	0.49	0.03	0.58	0.01	0.14	0.33	0.57
S-7	7	14	1.05	3.79	2.04	4.06	0.00	0.04	0.00	0.03	0.48	0.67	0.01	0.12	0.01	0.07	0.51	0.71
S-8	7	21	1.22	4.69	2.11	3.93	0.00	0.02	0.00	0.02	0.33	0.53	0.00	0.03	0.00	0.02	0.33	0.57
S-9	8	16	1.72	5.29	2.46	4.24	0.00	0.13	0.00	0.07	0.39	0.58	0.29	3.52	0.11	1.36	0.40	0.58
S-10	8	24	3.76	9.74	1.81	4.10	0.00	0.01	0.00	0.03	0.49	0.70	0.02	0.16	0.02	0.10	0.51	0.84
Average			**1.69**	**4.37**	**2.04**	**4.00**	**0.004**	**0.14**	**0.004**	**0.06**	**0.36**	**0.55**	**0.07**	**1.05**	**0.03**	**0.38**	**0.38**	**0.62**

genetic algorithm is 1.69 seconds, the average of the maximum times observed in scenarios is 4.37 seconds. For example, for the largest scenario, Scenario-10, the initial population is generated on average in 3.76 seconds, and in the worst-case scenario, it can take up to 9.74 seconds just for this task. It is also observed that the iteration process of the genetic algorithm takes an average of 2.04 seconds. After the sensor allocation decisions using the genetic algorithm in Stage 1.1, it is seen that engagement planning decisions are made very quickly in Stage 1.2. The construction of the engagement network and then solving the shortest path problem takes an average of 0.004 seconds for each threat. These operations for each threat in this stage cannot be performed in parallel but are done sequentially in order of priority. Finally, in Stage 2, determining heading decisions using dynamic programming takes an average of 0.36 seconds per ship. Unlike Stage 1.2, this stage's solution process is independent for each ship and can be run in parallel as simultaneous processes. In other words, Stage 2 can be applied independently by each ship solving its own heading solution. This ensures that after the engagement decisions are distributed to the ships, deciding which route to go throughout the scenario time takes an average of 0.36 seconds.

In terms of solution times, the situation is much more encouraging for the MHA^{EN+DP} algorithm. In line with the purpose of this algorithm, especially Stage 1 is being solved quickly. It takes 0.07 seconds to create the engagement network and 0.03 seconds to solve the shortest path problem for each threat. In some scenarios, when there are too many engagement alternatives for a threat, building the engagement network can take a little longer due to the large number of possible outcomes. For example, in Scenario 9, it is seen that this process is completed in a maximum of 3.52 seconds. Similarly, as the network is created larger, the stage of solving the shortest path problem can take a bit longer accordingly. Again, in Scenario 9, it is observed that the worst-case solution time for the SP problem is 1.36 seconds. At this point, as the scenario size grows, simple measures can be taken (such as eliminating some repetitive or low-pk alternatives) to ensure that the size of the engagement network does not become too large. Thus, rare elongations in solution times can be prevented. The solution time of Stage 2 is similar to Stage 2 of $DHA^{GA+EN+DP}$ algorithm.

Additionally, parameter tuning experiments can be conducted for the Stage 1.1 GA in $DHA^{GA+EN+DP}$, potentially compromising a bit on the effectiveness of the objective function to reduce the solution time to a practical level for real-world applications.

In conclusion, when the results for large-size scenarios are evaluated, it is seen that the $DHA^{GA+EN+DP}$ algorithm provides the best solutions, but in terms of feasible solution times, the MHA^{EN+DP} algorithm is more advantageous.

6.5.3 Comparison of the Solution Approaches in Close Distance Threats Case

In this experiment, the algorithms are compared for the scenario environment in which threats start approaching to the ships from a closer distance (20 NM) and there is limited time for counter-engagement. The results for small size scenarios are presented in Table 6.5. When examining the results, it is observed that the success of the MHA^{EN+DP} algorithm is increased compared to its normal distance version, and the $DHA^{GA+EN+DP}$ and the MHA^{EN+DP} algorithms achieve close objective function averages. Among the scenarios, the largest difference between MHA^{EN+DP} and $DHA^{GA+EN+DP}$ algorithms is observed in Scenario-17. Detailed examination of the solutions for Scenario-17 reveals that the $DHA^{GA+EN+DP}$ algorithm starts engagements a bit later than the MHA^{EN+DP} algorithm in the engagement plans generated against incoming threats. Therefore, fewer alternatives remain available against newly emerging threats. Although rare, in such situations, the success of the $DHA^{GA+EN+DP}$ algorithm can decrease because it does not consider future threats.

In Table 6.6, the experimentation results for larger scenarios are presented. In this experiment, a simulation environment simulating a swarm attack from close range against the ships is created. While the $DHA^{GA+EN+DP}$ algorithm can achieve an average OBJ-3 value of 0.361, the MHA^{EN+DP} algorithm yields a value of 0.389. In larger scenarios, where resource allocation is critical, it is observed that the $DHA^{GA+EN+DP}$ algorithm achieves better results. When compared to the results at normal distances (Table 6.3), the average percent deviation of the MHA^{EN+DP} algorithm decreases from 12.75% to 9.11%.

The solution time of the $DHA^{GA+EN+DP}$ algorithm in this experiment remains above the applicable level, with an average of 3.01 seconds for Stage 1.1. The average solution time for the MHA^{EN+DP} Algorithm Stage 1 is 0.35 seconds. For Scenario-

Table 6.5: Dynamic NADP Computational Experiments on Small-Mid Size Scenarios (Close Distance Threats Case)

			Average OBJ-3 Value Weighted Threat Hit Probability				**Percent Deviation % From the Best Solution (Average)**			
	Number of Ships	**Number of Threats**	$HA^{GA+EN+DP}$	$DHA^{GA+EN+DP}$	MHA^{EN+DP}	**Greedy Algorithm**	$HA^{GA+EN+DP}$	$DHA^{GA+EN+DP}$	MHA^{EN+DP}	**Greedy Algorithm**
S-1	1	2	0.228	0.216	0.216	0.243	5.61%	0.00%	0.00%	12.70%
S-2	1	3	0.341	0.326	0.329	0.341	4.73%	0.00%	1.04%	4.73%
S-3	1	4	0.696	0.617	0.617	0.696	12.75%	0.00%	0.00%	12.75%
S-4	2	2	0.277	0.277	0.276	0.276	0.11%	0.11%	0.00%	0.00%
S-5	2	3	0.306	0.303	0.303	0.314	1.26%	0.13%	0.00%	3.74%
S-6	2	3	0.532	0.532	0.532	0.532	0.00%	0.00%	0.00%	0.00%
S-7	2	4	0.333	0.333	0.338	0.370	0.09%	0.00%	1.47%	11.05%
S-8	2	5	0.495	0.494	0.494	0.521	0.12%	0.00%	0.00%	5.46%
S-9	3	2	0.011	0.011	0.011	0.028	0.00%	0.00%	0.00%	151.33%
S-10	3	3	0.215	0.215	0.215	0.229	0.00%	0.00%	0.00%	6.75%
S-11	3	4	0.257	0.257	0.254	0.328	1.26%	1.26%	0.00%	29.35%
S-12	3	4	0.101	0.100	0.101	0.121	0.50%	0.00%	0.80%	20.44%
S-13	3	5	0.142	0.141	0.148	0.200	0.64%	0.00%	4.83%	41.93%
S-14	4	2	0.122	0.122	0.122	0.126	0.00%	0.00%	0.00%	3.78%
S-15	4	3	0.240	0.240	0.243	0.240	0.00%	0.00%	1.00%	0.00%
S-16	4	4	0.205	0.205	0.198	0.269	3.64%	3.64%	0.00%	35.88%
S-17	4	5	0.178	0.178	0.136	0.174	31.37%	30.93%	0.00%	28.06%
S-18	5	3	0.041	0.040	0.042	0.073	2.74%	0.00%	5.49%	82.04%
S-19	5	4	0.223	0.223	0.223	0.253	0.00%	0.00%	0.00%	13.50%
S-20	5	5	0.259	0.261	0.254	0.236	9.61%	10.67%	7.54%	0.00%
Average			**0.2600**	**0.2545**	**0.2525**	**0.2785**	**3.72%**	**2.34%**	**1.11%**	**23.18%**
Number of Best Solution			**6**	**14**	**13**	**4**				
Average Run Time (s)			**1.41**	**1.47**	**0.50**	**0.44**				

9, average solution time is 3.24 seconds which is relatively long compared to the others. When examining the details for this scenario, it is observed that the process of constructing the engagement network during a solution update operation takes a very long time (69 seconds) due to the large number of engagement alternatives. As mentioned earlier, measures to reduce the number of alternatives can be taken to prevent such extreme situations.

6.5.4 The Value of Having Future Threat Information

For the same scenarios as in Table 6.5, further experimentations are conducted using the $DHA^{GA+EN+DP}$ algorithm in a simulated environment where all threat information is assumed to be available from the beginning of the scenario. The aim is to reveal the value of having future threat information. When examining the results, it is observed that with all threat information available from the beginning of the scenario, and the defense planning is conducted accordingly, the average objective function value can be reduced to 0.2489. For this scenario, it is observed that an average improvement of 3.82% ($4.26\% - 0.44\%$) in the objective function value is achieved. Particularly, it can be seen that having future threat information contributes to improving the solution for scenarios 12, 16, and 20. Thus, it has been demonstrated that possessing advanced intelligence and long-range surveillance systems contributes to more effective defense planning.

Table 6.6: Dynamic NADP Computational Experiments on Large Size Scenarios (Close Distance Threats Case)

	Number of Ships	Number of Threats	$DHA^{GA+EN+DP}$						MHA^{EN+DP}					Average Percent Deviation % From the Best Solution	
			OBJ-3	**Number of Runs**	**CI**	**Run Time Stage 1.1**	**Run Time Stage 1.2**	**Run Time Stage 2**	**OBJ-3**	**Number of Runs**	**CI**	**Run Time Stage 1**	**Run Time Stage 2**	**Dynamic** $HA^{GA+EN+DP}$	**Myopic** HA^{EN+DP}
S-1	5	10	0.285	50.00	0.045	1.80	0.00	1.08	0.274	50.00	0.040	0.01	1.12	4.02%	0.00%
S-2	5	12	0.247	50.00	0.046	2.81	0.00	0.80	0.281	50.00	0.046	0.04	0.78	0.00%	13.77%
S-3	5	15	0.445	50.00	0.039	3.07	0.00	0.90	0.491	50.00	0.043	0.01	0.91	0.00%	10.33%
S-4	6	12	0.248	50.00	0.043	3.19	0.00	0.88	0.285	54.00	0.050	0.05	0.86	0.00%	14.71%
S-5	6	15	0.376	50.00	0.046	3.64	0.01	1.18	0.436	50.00	0.038	0.09	1.24	0.00%	15.83%
S-6	6	18	0.413	50.00	0.040	2.77	0.00	1.18	0.474	50.00	0.030	0.02	1.18	0.00%	14.70%
S-7	7	14	0.369	50.00	0.047	2.68	0.00	2.14	0.363	50.00	0.044	0.01	2.11	1.57%	0.00%
S-8	7	21	0.480	50.00	0.033	2.68	0.00	1.28	0.510	50.00	0.033	0.01	1.30	0.00%	6.34%
S-9	8	16	0.274	50.00	0.037	3.73	0.01	1.84	0.316	50.00	0.041	3.24	1.81	0.00%	15.41%
S-10	8	24	0.476	50.00	0.043	3.70	0.00	1.89	0.464	50.00	0.030	0.01	1.86	2.41%	0.00%
Average			**0.361**	**50.00**	**0.042**	**3.01**	**0.00**	**1.32**	**0.389**	**50.40**	**0.039**	**0.35**	**1.32**	**0.80%**	**9.11%**

Table 6.7: Computational Experiments on the Value of Having Future Threat Information

	Number of Ships	Number of Threats	Average OBJ-3 Value Weighted Threat Hit Probability $DHA^{GA+EN+DP}$	$DHA^{GA+EN+DP}$ (With Future Threat Information)	Percent Deviation % From the Best Solution (Average) $DHA^{GA+EN+DP}$	$DHA^{GA+EN+DP}$ (With Future Threat Information)
S-1	1	2	0.216	0.216	0.00%	0.00%
S-2	1	3	0.326	0.334	0.00%	2.61%
S-3	1	4	0.617	0.617	0.00%	0.00%
S-4	2	2	0.277	0.276	0.11%	0.00%
S-5	2	3	0.303	0.303	0.00%	0.00%
S-6	2	3	0.532	0.532	0.00%	0.00%
S-7	2	4	0.333	0.331	0.54%	0.00%
S-8	2	5	0.494	0.494	0.02%	0.00%
S-9	3	2	0.011	0.011	0.00%	0.00%
S-10	3	3	0.215	0.214	0.28%	0.00%
S-11	3	4	0.257	0.257	0.00%	0.00%
S-12	3	4	0.100	0.081	23.83%	0.00%
S-13	3	5	0.141	0.142	0.00%	0.64%
S-14	4	2	0.122	0.122	0.00%	0.00%
S-15	4	3	0.240	0.240	0.00%	0.00%
S-16	4	4	0.205	0.198	3.28%	0.00%
S-17	4	5	0.178	0.133	33.68%	0.00%
S-18	5	3	0.040	0.042	0.00%	5.49%
S-19	5	4	0.223	0.223	0.00%	0.00%
S-20	5	5	0.261	0.212	23.37%	0.00%
Average			**0.2545**	**0.2489**	**4.26%**	**0.44%**

6.5.5 The Value of Having Full Coordination in the Task Group

The developed algorithms assume a full coordination among friendly ships, where detected threat information are shared, and counter engagement planning is collaboratively conducted by all ships. If full coordination capability is absent among ships, and each ship independently plans its defense against incoming threats, the results shown in Table 6.8 emerge.

In the case where the $DHA^{GA+EN+DP}$ algorithm is used, and all weapons are considered area-defense systems, the average OBJ-3 value is observed as 0.1572. However, when the weapons are considered only as self-defense systems, this value increases to 0.2689. Thus, when the same solution approach is used, the contribution of using weapons in full coordination is seen as 0.1117 $(0.2689 - 0.1572)$ in the average OBJ-3 value. Additionally, if the Greedy Algorithm is used instead of the newly developed

Table 6.8: Computational Experiments on the Value of Having Full Coordination in the Task Group

			$DHA^{GA+EN+DP}$ (All weapons are area-defense systems.)		$DHA^{GA+EN+DP}$ (All weapons are self-defense systems.)		Greedy Algorithm (All weapons are self-defense systems.)	
	Number of Ships	**Number of Threats**	**Average OBJ-3**	**Average OBJ-4**	**Average OBJ-3**	**Average OBJ-4**	**Average OBJ-3**	**Average OBJ-4**
s-1	3	6	0.0627	0.9000	0.1494	0.7644	0.3060	0.6017
s-2	3	9	0.2710	0.4211	0.3378	0.3100	0.4051	0.2515
s-3	4	4	0.2067	0.8895	0.3190	0.8473	0.3325	0.8414
s-4	4	8	0.1375	0.7692	0.1870	0.7220	0.2400	0.6800
s-5	5	10	0.1079	0.7983	0.3514	0.5567	0.3876	0.6204
Average			**0.1572**	**0.7556**	**0.2689**	**0.6401**	**0.3342**	**0.5990**

algorithm, the loss in the average objective function value is calculated as 0.1770 $(0.3342 - 0.1572)$. This experiment clearly demonstrates the significance of having full coordination capability and using efficient algorithms in naval combat scenarios.

Similarly, Table 6.9 presents the results when threats are distributed according to the sectors from which they approach, as shown in Figure 6.3, where 3 ships face 9 threats approaching from 3 different sectors. In the $DHA^{GA+EN+DP}$ algorithm solution, full coordination among ships is assumed, while in the Greedy Algorithm solution, sectors are assigned to ships, and each ship can engage only the threats coming from its assigned sectors.

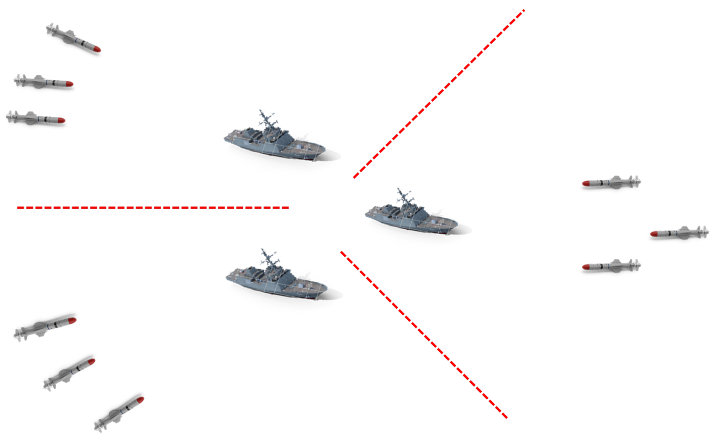


Figure 6.3: Example Sector Allocation Scenario

According to the results, having full coordination capability and employing an effec-

Table 6.9: Computational Experiments on the Value of Having Full Coordination in the Task Group-2

	Number of Ships	Number of Threats	$DHA^{GA+EN+DP}$ (Full Coordination) OBJ-3	$DHA^{GA+EN+DP}$ (Full Coordination) OBJ-4	Greedy Algorithm (Engage Only in the Assigned Sector) OBJ-3	Greedy Algorithm (Engage Only in the Assigned Sector) OBJ-4
S-1	3	9	0.189	0.574	0.339	0.359
S-2	3	9	0.394	0.340	0.515	0.320
S-3	3	9	0.322	0.351	0.474	0.172
S-4	3	9	0.331	0.573	0.489	0.283
S-5	3	9	0.418	0.210	0.508	0.130
S-6	3	9	0.459	0.336	0.500	0.243
S-7	3	9	0.495	0.184	0.503	0.174
S-8	3	9	0.461	0.171	0.602	0.067
S-9	3	9	0.360	0.364	0.523	0.129
S-10	3	9	0.528	0.122	0.594	0.040
Average			**0.396**	**0.323**	**0.505**	**0.192**

tive solution method increase the average OBJ-3 by 0.109 $(0.505 - 0.396)$. In terms of OBJ-4, it increases the weighted survival probabilities of friendly ships by 0.131 $(0.323 - 0.192)$.

6.6 Implementation of a Modified-SLS Firing Policy

The Shoot-Look-Shoot (SLS) firing policy is a conventional strategy employed in naval air defense systems. This policy is designed to optimize the use of interceptors by minimizing their expenditure and maximizing their effectiveness per engagement, thereby reducing the overall cost of defense given the high price of each interceptor. However, during combat, preserving the ship's integrity becomes paramount, and sometimes, more aggressive firing policies may be required to ensure the survival of the platform.

In the mathematical model of the NADP problem defined in this study, the SLS firing policy is the standard approach. This policy restricts engagements to at most one per target at any given time. A modified version of the SLS policy can be utilized to enhance the flexibility and effectiveness of air defense operations. This new policy relaxes the constraint of at most one engagement per target, allowing up to two simultaneous engagements for a single target. Although this approach can increase the probability of successfully neutralizing threats, it also accelerates the consumption of resources, potentially leaving the defense vulnerable to possible subsequent threats.

To implement the modified-SLS firing policy in the mathematical model, the right-hand side of the Constraint Set (4.3) (Section 4.6) can be increased to 2 as presented below. And the value of the parameter μ_{if}, upper bound on the number of engagements from fire channel f against threat i being used in the Constraint Set (4.2), should be determined according to the firing policy. These adjustments allow for the representation of the modified-SLS firing policy.

$$\sum_{t \in T_{if}} Y_{ift} \leq \mu'_{if} \qquad \forall (i, f) \in G, \tag{6.2}$$

$$\sum_{(f,\rho) \in J_{it}} Y_{if\rho} \leq 2 \qquad \forall i \in A, t \in T \tag{6.3}$$

Table 6.10 shows the results of the experiment comparing the conventional SLS and the modified-SLS firing policies. For the solution approach, the dynamic GA+DP method is utilized. Due to the potentially large number of nodes in the engagement network when using the modified-SLS approach, creating this network becomes time-consuming. Therefore, in this experiment, DHA^{GA+DP} algorithm (dynamic version of Algorithm 4, $DHA^{GA+EN+DP}$ without Stage 1.2) is used.

When the constraint of one engagement per target is increased to two, the average OBJ-3 value for 30 scenarios decreases from 0.2529 to 0.2059, indicating a 0.05 reduction in the weighted average probability of threats hitting the friendly ships. However, as expected, the modified-SLS policy consumes more ammunition, with ships having 15% less ammunition remaining at the end of the scenario compared to the SLS policy. In scenarios 2, 3, and 23, all ammunition was depleted by the end of the engagements, likely resulting in some threats not being intercepted, leading to poorer OBJ-3 values compared to the SLS policy.

The SLS firing policy and the modified-SLS firing policy offer distinct advantages

Table 6.10: Comparison of the SLS vs. Modified-SLS Firing Policies

			DHA^{GA+DP}						DHA^{GA+DP} with Mod-SLS policy					
	Number of Ships	**Number of Threats**	**OBJ-1**	**OBJ-3**	**OBJ-4**	**Run Time**	**Inv Left Average**	**Inv Left Percentage**	**OBJ-1**	**OBJ-3**	**OBJ-4**	**Run Time**	**Inv Left Average**	**Inv Left Percentage**
S-1	1	2	0.815	0.148	0.815	1.68	2.449	40.8%	0.889	0.070	0.889	1.85	1.011	16.8%
S-2	1	3	0.161	0.412	0.161	1.37	0.000	0.0%	0.079	0.501	0.079	1.43	0.000	0.0%
S-3	1	4	0.035	0.604	0.035	1.08	0.000	0.0%	0.020	0.577	0.020	1.25	0.000	0.0%
S-4	2	2	0.686	0.172	0.828	1.50	2.360	39.3%	0.783	0.117	0.883	2.45	0.925	15.4%
S-5	2	3	0.221	0.415	0.568	1.24	6.353	63.5%	0.449	0.248	0.750	1.61	3.164	31.6%
S-6	2	3	0.382	0.255	0.688	1.27	1.027	20.5%	0.402	0.260	0.657	1.61	0.000	0.0%
S-7	2	4	0.049	0.346	0.675	1.31	1.150	28.8%	0.052	0.318	0.715	1.59	0.732	18.3%
S-8	2	5	0.640	0.108	0.790	3.84	7.225	45.2%	0.745	0.091	0.837	4.14	4.296	26.8%
S-9	3	2	0.788	0.120	0.949	1.32	10.600	75.7%	0.944	0.028	0.988	1.76	8.948	63.9%
S-10	3	3	0.395	0.224	0.777	1.43	4.969	55.2%	0.514	0.124	0.876	2.26	3.219	35.8%
S-11	3	4	0.125	0.498	0.583	1.85	7.522	57.9%	0.103	0.445	0.590	2.19	6.609	50.8%
S-12	3	4	0.148	0.406	0.502	1.57	1.142	19.0%	0.183	0.348	0.535	1.90	0.000	0.0%
S-13	3	5	0.042	0.439	0.516	1.97	3.418	42.7%	0.051	0.394	0.561	2.30	1.674	20.9%
S-14	4	2	0.814	0.099	0.950	2.41	12.687	74.6%	0.949	0.027	0.986	3.17	11.495	67.6%
S-15	4	3	0.721	0.106	0.930	1.39	17.918	81.4%	0.929	0.025	0.983	1.83	15.133	68.8%
S-16	4	4	0.626	0.168	0.888	3.99	13.381	66.9%	0.800	0.089	0.941	4.60	10.489	52.4%
S-17	4	5	0.469	0.166	0.835	1.95	12.458	62.3%	0.775	0.057	0.934	3.47	8.131	40.7%
S-18	5	3	0.506	0.204	0.859	3.25	13.821	76.8%	0.557	0.174	0.880	3.05	11.151	61.9%
S-19	5	4	0.733	0.076	0.927	4.33	10.280	60.5%	0.757	0.065	0.936	4.91	8.177	48.1%
S-20	5	5	0.440	0.154	0.843	3.97	10.860	54.3%	0.427	0.162	0.831	3.89	9.376	46.9%
S-21	2	6	0.080	0.442	0.222	1.77	0.941	10.5%	0.040	0.445	0.166	1.69	0.309	3.4%
S-22	2	7	0.032	0.455	0.197	1.04	1.881	20.9%	0.015	0.365	0.187	1.22	0.432	4.8%
S-23	2	8	0.000	0.378	0.280	2.07	0.361	5.2%	0.000	0.418	0.230	1.85	0.001	0.0%
S-24	3	6	0.780	0.058	0.915	4.74	5.041	26.5%	0.800	0.057	0.903	5.26	3.341	17.6%
S-25	3	7	0.360	0.158	0.790	3.72	4.461	24.8%	0.540	0.071	0.873	4.15	2.421	13.4%
S-26	3	8	0.080	0.309	0.475	1.72	4.261	28.4%	0.040	0.296	0.445	1.92	0.741	4.9%
S-27	3	9	0.000	0.378	0.353	1.73	4.629	28.9%	0.020	0.363	0.413	2.58	2.241	14.0%
S-28	4	6	0.482	0.151	0.815	2.06	4.616	25.6%	0.480	0.133	0.836	2.39	4.901	27.2%
S-29	4	7	0.000	0.380	0.426	3.36	3.230	19.0%	0.043	0.359	0.484	3.77	1.088	6.4%
S-30	4	8	0.118	0.265	0.565	2.50	10.845	49.3%	0.180	0.184	0.743	2.48	8.261	37.6%
Average			**0.4398**	**0.2559**	**0.7059**	**2.14**	**6.98**	**48.3%**	**0.5203**	**0.2059**	**0.7435**	**2.56**	**5.23**	**33.3%**

and disadvantages, which present a trade-off between resource conservation and engagement effectiveness. The SLS policy is advantageous in conserving resources and reducing costs but may fall short in scenarios where threats are more persistent or difficult to neutralize with a single interceptor. On the other hand, the modified-SLS policy enhances the probability of successful engagements by allowing multiple interceptors per target, but this comes at the cost of faster resource depletion and higher defense costs.

This experiment provides an important insight into the characteristics of NADP problem solution approaches. Strategic decisions on which firing policy to adopt must consider the specific combat environment, including the anticipated volume and persistence of threats, the availability of resupply, and the overall mission objectives. An intelligent decision support system could dynamically adapt firing policies based on real-time assessments of these factors, optimizing both the immediate and long-term effectiveness of the naval defense.

Meanwhile, when engaging the same target simultaneously with different weapon systems, there is a risk of interference between the weapons based on their characteristics. While the SLS firing policy does not encounter this issue, the modified SLS policy must account for this potential interference. In practical implementations, this factor can be incorporated into the solution by considering interference risks in the scheduling of engagements. This can be achieved through adjustments in the genetic algorithm's fitness value calculation or during the construction of the engagement network, ensuring that interference is avoided.

In the air defense literature, various engagement policies exist in addition to the SLS policy, such as the Shoot-Shoot-Look (SSL) policy. The SSL policy is typically employed by launching two SAMs simultaneously, particularly to increase the survival probability of the ship during the last engagement opportunity against a threat. In our study, the modified-SLS policy allows for two simultaneous engagements, similar to SSL, if doing so improves the objective function value. To fully implement the SSL policy, an additional constraint can be introduced in the model, requiring two engagements on the same target within a specific time frame. However, when comparing the SSL policy to our modified-SLS approach, we can assert that the SSL policy would lead to higher ammunition consumption and, due to the additional constraint, potentially resulting worse objective function values.

6.7 Threat Prioritization Using Machine Learning Models

In the MHA^{EN+DP} Algorithm, threats are prioritized based on the calculated ratio $(\frac{HitTime_i}{Value_i})$ which considers the hit time and the value of threats, and engagement plans are determined one by one according to this priority order. This ratio ensures that threats with shorter time to hit and higher value are prioritized. In this section, the aim is to demonstrate the potential improvement in the objective function value when threat priorities are determined using machine learning models based on a set of attributes describing the naval air defense warfare environment. This approach can enable a more appropriate allocation of defense resources by analyzing the changing threat environment.

In other words, this section attempts the use of machine learning models trained on a dataset generated through simulation, as an alternative to comparing a simple ratio for the threat prioritization problem that can be expressed with much more complex relationships. The pseudocode of the MHA^{EN+DP} Algorithm with modified threat prioritization is presented in Algorithm 9 (changes are highlighted):

In this algorithm, the machine learning model determines the priority order of the incoming threats by conducting pairwise comparisons between the threats that need a solution update. The machine learning model identifies which of the two threats should be prioritized and thus creates a priority order through pairwise comparisons.

To train the models, a dataset consisting of 2700 instances is created using the dynamic NADP simulation environment developed in Section 6.3. Note that, increasing the number of instances can help improve the model's performance. Each row of this dataset contains features related to the threats and the defense resources in a randomly generated scenario. For the generation of the dataset, feature data characterizing the current environment are recorded during the simulation flow at the moment when a solution is needed for at least two threats.

The feature set used to train the machine learning models is listed below. The features representing the severity of the threats and the adequacy of defense resources available against these threats are selected for the model setup.

The feature set of the model:

1. Difference in the threat values ($v_{ASM1} - v_{ASM2}$)
2. Difference in time remaining until the threat hit times
3. Difference in the number of possible engagement alternatives against the threats
4. Difference in the number of different fire channels available against the threats
5. Difference in the sum of P_{ift} probabilities of engagements that can be made by

Algorithm 9 MHA^{EN+DP} Algorithm with Modified Threat Prioritization

Generate/Update problem parameters

V: set of all valid $\{i, f, t\}$ engagement combinations of threat, fire channel and engagement start time

Stage-1: Engagement Planning Decisions

Order the threats using the machine learning model

-Make pairwise comparisons between the threats that need solution update

for $\forall i \in A$ **do**

If there is no active engagement going towards ASM i

-$V_i \in V$: set of valid engagements against ASM i

If the previously planned engagement against ASM i is in set V_i

-Select the previously planned engagement

else

-Create Engagement Graph using V_i set

-Solve Shortest Path Problem using Bellman-Ford Algorithm

-Select the first engagement for ASM i using the shortest path solution

end if

-Delete conflicting engagement alternatives with the selected engagement from the V set

end if

-Update p_{ift} single shot kill probabilities of the engagement alternatives that can be fired from the target ship of threat i after $HitTime_i$ according to the hit probability of the threat i.

end for

Stage-2: Heading Decisions

for $\forall n \in N$ **do**

-Determine H decision variables using a backward recursion DP Approach

end for

the target ship after the threat hit time

6. Difference in the maximum number of engagements that can be made against the threats

7. Difference in the earliest possible engagement end time against the threats

8. Difference in the earliest possible engagement start time against the threats

9. Difference in ($\frac{HitTime_i}{Value_i}$) ratios of the threats

10. Earliest engagement end time for Threat-1 minus earliest engagement start time for Threat-2

11. Difference in minimum engagement durations that can be made against the threats

12. Difference in the intersection rates of engagements that can be made against the threats

13. Difference in the survival probabilities of threats (quick estimate calculation)

14. Difference in the weighted survival probabilities of threats (quick estimate calculation)

15. Difference in the inventory levels that can be used against the threats

16. Difference in total ammunition inventory of the target ship of the threats

17. Difference in the objective function values that can be obtained when the threats are prioritized (quick estimate calculation)

For each pair of threats, expected values of the objective function are calculated assuming each threat is prioritized in turn. The difference between these values ($OBJ3(ASM_{ab}) - OBJ3(ASM_{ba})$) is recorded as a penalty/reward score. Here, $OBJ3(ASM_{ab})$ represents the objective function value achievable when threat ASM_a is prioritized over ASM_b. This approach allows for a quantitative assessment of the impact of prioritizing one threat over another.

In this section, the primary focus is not extensively on the intricate processes of feature selection and model parameter tuning. Although these elements are critical for optimizing the performance of machine learning models, the primary aim of this section is to demonstrate the potential contribution of machine learning approaches to solving the NADP problem, rather than to dive deeply into the nuances of model optimization.

The selection of the feature set is driven by the need to effectively represent the complexity of the naval air defense environment. The features are chosen based on their relevance in describing the threat scenario and the capabilities of defense systems. This approach ensures that the machine learning model can make informed predictions about threat prioritization, which is crucial for effective defense planning. However, comprehensive refinement of the feature set is not the main focus; instead, the chosen features are intended to sufficiently capture the scenario dynamics to test the feasibility of machine learning applications in this field.

Similarly, parameter tuning in machine learning is often a meticulous process that involves adjusting various settings that control the learning process of the models (such as the learning rate, the number of trees in a forest model, or the depth of the trees). These parameters significantly affect the model's accuracy and efficiency. However, for the purposes of this study, the standard parameter settings are used to establish a baseline for the machine learning models. This approach allows for an initial assessment of the effectiveness of the models without the need for extensive tuning.

In this study, the following machine learning models, which are readily available as built-in functions in MATLAB [56] [57], are used with default parameters:

1. Support Vector Regression (SVR)
   - Statistics and Machine Learning Toolbox (fitrsvm)
2. Support Vector Machine (SVM)
   - Statistics and Machine Learning Toolbox (fitcsvm)
3. Linear Regression (LR)
   - Statistics and Machine Learning Toolbox (fitlm)
4. K-Nearest Neighbors (KNN)
   - Statistics and Machine Learning Toolbox (fitcknn)
5. Decision Trees (DT)
   - Statistics and Machine Learning Toolbox (fitrtree)

6. Random Forest (RF)
   - Statistics and Machine Learning Toolbox (TreeBagger)

7. Pattern Recognition Neural Networks (PRNN)
   - Deep Learning Toolbox (patternnet)

8. Multilayer Perceptron (MLP)
   - Deep Learning Toolbox (fitnet)

The dataset consisting of 2700 instances is used to train these models.

6.7.1 The Contribution of Using Machine Learning Models for the Threat Prioritization

The data set used in this study is generated based on randomly created scenarios, with the number of ships ranging from 1 to 4 and the number of threats ranging from 2 to 4. The first computational experiment to assess the performance of machine learning models is initially conducted on 50 scenarios of these dimensions. The results of this experiment is presented in Table 6.11. The MHA^{EN+DP} Algorithm is used as the solution method, and comparisons are made using nine different threat prioritization approaches: the basic calculated ratio of hit time to threat value, and eight machine learning models listed above. These nine approaches are compared to evaluate the threat prioritization process.

Upon examining the results, it can be observed that the eight machine learning models trained with default parameters perform better than the simple ratio approach in threat prioritization. The baseline algorithm using the simple ratio shows an average deviation of 2.58% from the best solution, lagging behind in 16 out of 50 scenarios. On the other hand, the best-performing model, Decision Trees model, achieves the best solution in 41 scenarios with a deviation of only 0.75%.

In terms of the run times, it can be observed that using the approach with machine learning models results in an increase in Stage 1 solution time ranging from 0.01 to 0.07 seconds on average compared to the baseline algorithm. The use of machine learning models requires the preparation of the feature set and making the pairwise

Table 6.11: Computational Experiments-1 on the Contribution of Using Machine Learning Models for the Threat Prioritization

	Average OBJ-3	**Average Percent Deviation (%)**	**# of Best Solutions**	**Run Time Stage 1**
Ratio	0.2402	2.58%	34	0.07
DT	0.2361	0.75%	41	0.14
RF	0.2360	0.78%	40	0.14
KNN	0.2366	0.98%	38	0.09
SVM	0.2372	1.35%	37	0.10
SVR	0.2376	1.46%	35	0.13
MLP	0.2375	1.52%	35	0.14
PRNN	0.2381	1.79%	37	0.10
LM	0.2379	1.96%	38	0.08

comparisons. Since the model training is already completed, it is assumed that the machine learning model is readily available. When we input the feature set consisting of 17 feature values, the result (which threat is prioritized) can be determined very quickly by the model.

To replicate this analysis on larger scenarios, the DT model and the simple ratio approach are compared in 25 scenarios, generated with 4 ships and 4 threats, as seen in Table 6.12. The DT model, which yielded the best results in the previous analysis, continues to perform successfully compared to the simple ratio approach. The DT model produces the best solutions in nearly all scenarios, achieving a deviation of only 0.06%. The simple ratio approach gets an average deviation of 2.91%.

The average Stage 1 run time, which is 0.10 seconds in the baseline algorithm, is observed to increase to an average of 0.24 seconds for the algorithm with the machine learning approach. According to these results, it is observed that the machine learning approach does not render the algorithm's solution time unfeasible, but rather leads to an improvement in the objective function with an acceptable increase in time.

These results indicate that machine learning models in threat prioritization can provide improvements over the basic ratio approach, even when using default model configurations. By demonstrating that, the study aims to highlight the potential for more detailed future research that could focus on optimizing these models to achieve even greater efficiencies.

Table 6.12: Computational Experiments-2 on the Contribution of Using a Machine Learning Model (DT Approach) for the Threat Prioritization

	Average OBJ-3		Average Percent Deviation (%)		Run Time Stage 1	
	Ratio	DT	Ratio	DT	Ratio	DT
S-1	0.1776	0.1776	0.00%	0.00%	0.01	0.27
S-2	0.0626	0.0626	0.00%	0.00%	0.27	0.41
S-3	0.1870	0.1870	0.00%	0.00%	0.02	0.16
S-4	0.0516	0.0524	0.00%	1.55%	0.10	0.27
S-5	0.1169	0.1169	0.00%	0.00%	0.00	0.13
S-6	0.1701	0.1701	0.00%	0.00%	0.01	0.20
S-7	0.1657	0.1657	0.00%	0.00%	0.01	0.20
S-8	0.0828	0.0828	0.00%	0.00%	0.05	0.22
S-9	0.1371	0.1371	0.00%	0.00%	1.43	1.50
S-10	0.0781	0.0781	0.00%	0.00%	0.04	0.19
S-11	0.0443	0.0351	26.21%	0.00%	0.08	0.26
S-12	0.3278	0.3227	1.58%	0.00%	0.01	0.17
S-13	0.1386	0.1362	1.76%	0.00%	0.03	0.22
S-14	0.1412	0.1412	0.00%	0.00%	0.02	0.13
S-15	0.1964	0.1761	11.53%	0.00%	0.03	0.14
S-16	0.0764	0.0764	0.00%	0.00%	0.01	0.14
S-17	0.0958	0.0958	0.00%	0.00%	0.02	0.11
S-18	0.1294	0.1292	0.15%	0.00%	0.05	0.14
S-19	0.2534	0.2534	0.00%	0.00%	0.01	0.14
S-20	0.4757	0.4757	0.00%	0.00%	0.00	0.10
S-21	0.0463	0.0352	31.53%	0.00%	0.05	0.23
S-22	0.2889	0.2889	0.00%	0.00%	0.02	0.08
S-23	0.2540	0.2540	0.00%	0.00%	0.01	0.14
S-24	0.1808	0.1808	0.00%	0.00%	0.00	0.13
S-25	0.0903	0.0903	0.00%	0.00%	0.18	0.36
Average	**0.1588**	**0.1569**	**2.91%**	**0.06%**	**0.10**	**0.24**

CHAPTER 7

IMPLEMENTATION OF THE NADP DECISION SUPPORT AUTOMATION

The NADP solution algorithm is developed to be used in combat management systems to effectively protect naval task group platforms from aerial threats. The NADP algorithm can optimize and automate maritime air defense planning. It encompasses the rapid planning of defense resources, including ships' maneuvering decisions and the allocation and timing of weapons and sensors against threats.

In this study, the NADP problem is expressed as a "Mixed-Integer Nonlinear Programming (MINLP) Model" that includes constraints considering weapon/sensor assignments, sequence-dependent setup times, weapon/sensor blind sectors, and the infrared/radar signatures of ships, aimed at minimizing the hit probabilities of threats. This problem, belonging to the NP-hard complexity class, takes impractically long time even in small scenarios to solve exactly. Hence, obtaining a quick and effective solution that can meet real-time operational needs in the rapidly evolving maritime warfare environment is of critical importance.

The centralized use of the NADP algorithm in combat management systems can significantly enhance the air defense capabilities of naval task groups. The results of the simulation analysis shown in previous chapters demonstrate the contribution of the developed algorithm in effectively protecting ships from threats, improving the coordination between the friendly forces, and providing a fast defense response in the dynamic threat environment.

In the face of air attacks involving simultaneous and numerous threats (as can be seen in the recent conflicts between Israel and Iran, on April 12, 2014 [1]), it is critically important to have fast solution algorithms that can provide decision support automation and enable air defense systems to demonstrate a coordinated and effective defensive response.

With recent advancements in communication/link capabilities, naval platforms have begun to possess network-enabled operational capabilities, and the need to solve this problem has become significant. In the past (or still for platforms without network-enabled operational capabilities), threat allocation in naval task groups usually involved methods based on self-defense or sector allocation as shown in Figure 6.3. However, such methods do not ensure effective utilization of defense resources.

Our study offers a solution approach to the increasingly complex and challenging nature of air defense in the maritime warfare environment. This algorithm can be integrated into the naval ships' combat management systems to enhance the effectiveness of air defense operations, providing quick and flexible decision support to meet operational needs.

In a network-enabled naval task group, all friendly platforms can demonstrate a fully coordinated air defense reaction. Thus, the overall defense effectiveness of the task group can be increased. As seen in Figure 7.1, platforms in a network-enabled task group can share data/information/orders and demonstrate a coordinated defensive reaction against threats. Data sharing can be facilitated using satellite or radio communication systems.

The central use of the dynamic NADP Algorithm in combat management systems can be performed through the steps shown in Figure 7.2.

1. **Data Collection Related to the Warfare Environment:** Data from radars, sonars, and other sensors located on all friendly air/land/sea platforms are collected.

2. **(Decision) New Solution Required?:** After the data collection, it is determined whether a new solution is needed. A new solution is required when one of the conditions explained in Section 6.1 is encountered.

3. **Creation of the Algorithm Input Parameters:** The central combat management system integrates data from all connected air/land/sea platforms and its

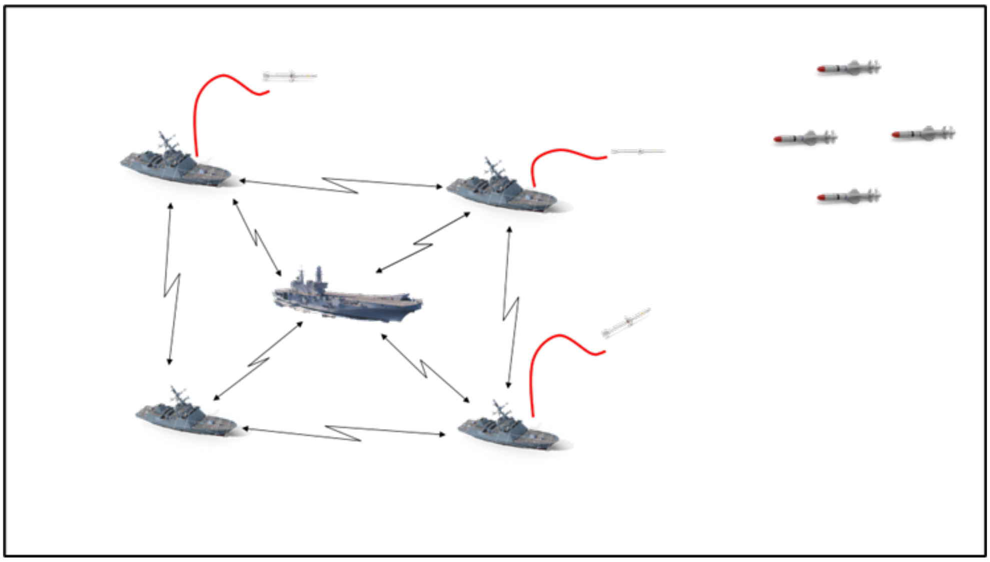


Figure 7.1: Network-Enabled Naval Task Group

own sensors. The collected data are converted into a format that can be used by the NADP Algorithm to create input parameters for the solution approach. At this point, an evaluation of all defense resources and threats within the task group is carried out to create the model parameters. This parameter set is continuously updated by the combat management system. Steps related to data collection and algorithm parameter creation are presented in Figure 7.3.

4. **Problem Solution and Engagement Planning (NADP Algorithm Stage-1):** The algorithm assigns current defense resources (guided missiles, naval gun systems, electronic warfare tools, etc.) to threats in a way that optimizes the objective function and schedules timing. Detailed steps involve threat allocation to sensors using a genetic algorithm-based solution approach in Stage 1.1 of the algorithm, and engagement scheduling decisions are determined by creating an engagement network and solving the shortest path problem in Stage 1.2.

5. **Problem Solution and Maneuver Decisions (NADP Algorithm Stage-2):** In Stage 2, the algorithm plans the maneuvering decisions of the ships using a recursive backward dynamic programming approach. During this stage, it aims to ensure that engagements are not obstructed by weapon and sensor blind sectors and maneuvers are conducted to minimize the sensor signatures against threats.

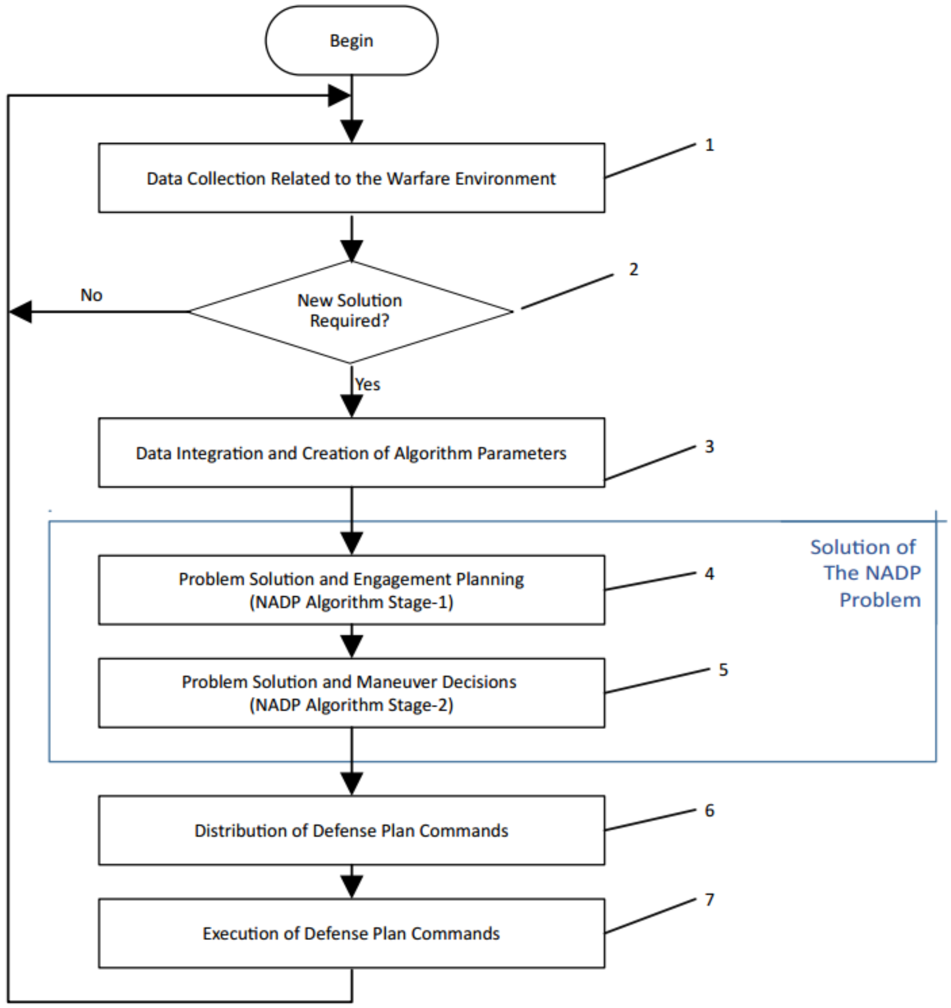


Figure 7.2: A Dynamic NADP DSS Implementation Steps

6. **Distribution of Defense Plan Commands:** After the algorithm solution, engagement plans and maneuvering decisions are distributed to the relevant defense resources and ships in the task group. This process is carried out both locally (for systems on the central ship) and over the network (for other ships in the task group).

7. **Execution of Defense Plan Commands:** Platforms implement defensive reactions. The war environment is dynamic, and the combat environment continuously changes. The NADP solution is constantly updated with new information about the warfare environment.

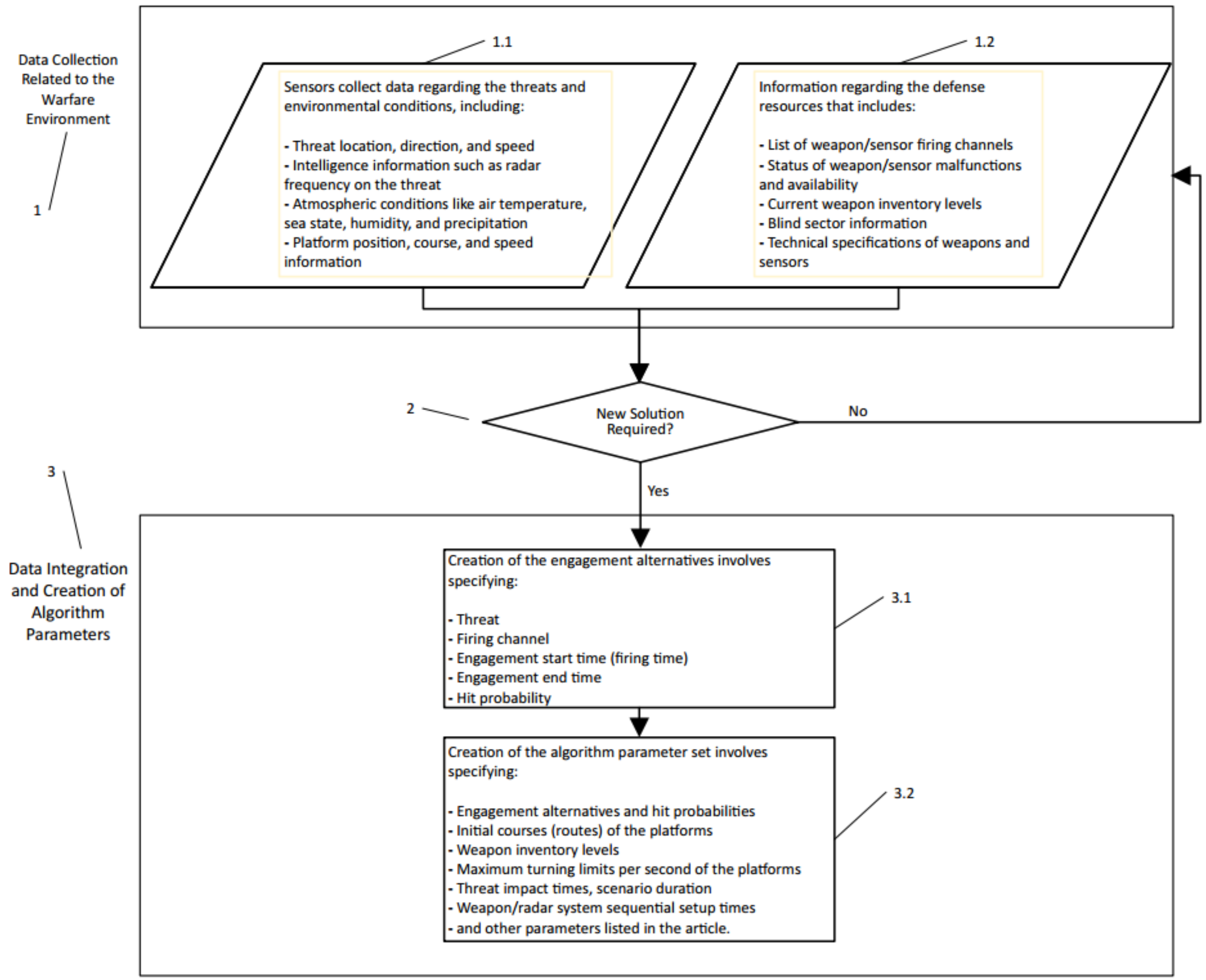


Figure 7.3: Data Collection and Algorithm Input Creation

CHAPTER 8

CONCLUSION

In this study, we addressed the NADP problem, which involves protecting friendly naval assets from aerial threats in a maritime environment. In the first part of the thesis, we focused on the static version of the problem. We developed a MINLP model for the static NADP problem and proposed heuristic solution approaches to overcome its computational complexity.

The computational results obtained from our study provide valuable insights and conclusions regarding the NADP problem and the effectiveness of the proposed heuristic algorithms. The NADP problem is a challenging and complex optimization problem due to its NP-completeness and the inclusion of various realistic features such as sensor assignment requirements, weapon and sensor blind sectors, sequence-dependent setup times, and ship's infrared/radar signature. Solving the problem exactly using the MINLP model is computationally expensive, even for small scenario sizes. Therefore, we firstly linearized the nonlinear model to obtain approximate solutions. The linearized model proved to be a successful approximation, providing effective solutions within a reasonable time for a significant number of scenarios.

Following the linearization, our initial step in developing a solution method was applying the decomposition of the mathematical model. While this process effectively reduces the solution times, it still falls short of reaching a level suitable for real-time applications. Therefore, we developed the heuristic algorithms to solve the NADP problem efficiently utilizing the decomposition of the mathematical model. The Greedy Algorithm produced good initial solutions for the linearized models, while the $HA^{GA+EN+DP}$ and HA^{GA+DP} algorithms provided fast and effective solutions for most NADP scenarios. The heuristic algorithms demonstrated their ability to generate solutions that closely approximated the solutions obtained from the linearized model, with small deviations in the objective function values. The heuristic algorithms also significantly reduced the computation time compared to the linearized model, making them practical and efficient tools for solving the NADP problem in real-life situations.

In the second part of the study, we investigated the NADP problem in a dynamic warfare environment. Unlike the static version, the dynamic NADP problem accounts for real-world implications such as the loss of functionality in weapon/sensor systems if a friendly ship is hit. This chapter introduces heuristic solution approaches, including the $DHA^{GA+EN+DP}$ algorithm, to address the need for rapid decision-making in evolving naval warfare scenarios. Computational experiments reveal that while this algorithm provides superior solutions, the MHA^{EN+DP} algorithm proves to be more efficient in terms of solution times for larger scenarios. The importance of full coordination among friendly ships and the value of possessing future threat information are underscored through comprehensive experiments, shedding light on effective naval air defense planning strategies.

Comparison of SLS firing policy and modified-SLS firing policy provided an important insight into the NADP solution characteristics. The trade-off between the SLS and modified-SLS firing policies highlights the critical balance between conserving resources and maximizing engagement effectiveness. While the SLS policy ensures efficient use of interceptors and cost savings, it may fall short against persistent threats. Conversely, the modified-SLS policy enhances the probability of neutralizing targets but at the expense of accelerated resource depletion and higher costs. These findings underscore the importance of adaptive firing strategies that consider real-time combat conditions, resupply opportunities, and anticipated threat levels to optimize naval defense operations.

Additionally, it is proved that machine learning models can enhance the threat prioritization process by analyzing a set of attributes that describe the naval air defense warfare environment, leading to improved allocation of defense resources. Computational experiments indicate that machine learning models outperform the simple ratio method, demonstrating the potential benefits of integrating machine learning into NADP solution approaches.

Lastly, the implementation of the NADP Decision Support Automation topic is addressed, and steps are outlined for how NADP solution approaches can be applied into a Network-Enabled Naval Task Group. Thus, this study sheds light on how a real-world implementation can be achieved.

Overall, this study contributes to addressing the increasingly complex and challenging nature of air defense in the maritime environment. The proposed heuristic algorithms provide fast and efficient solutions for the NADP problem by optimizing the allocation of defense resources to minimize threats to friendly naval assets. These algorithms can be embedded into the command and control systems of naval ships, providing decision support and enhancing the effectiveness of air defense operations.

Future research can focus on extending the use of machine learning models in the NADP solution to include engagement planning decisions alongside threat prioritization. Methods can be developed to integrate real-time adjustments to engagement schedules based on evolving threat environments and operational defense resource statuses. Exploring machine learning algorithms may provide optimal engagement strategies through iterative learning. Also, developing the modified-SLS policy with advanced resource allocation rules that consider various factors such as threat prioritization, ammunition type, resupply capability and future threat expectation may lead to more resilient and efficient naval defense operations.

Furthermore, the findings and methodologies developed in this study for naval air defense can be extended to various other defense scenarios. For instance, the solution approach can be adapted to protect a naval task group from asymmetric surface threats or to safeguard a land base against aerial and ground threats.

REFERENCES

[1] Wikipedia.org, "2024 iranian strikes in israel." Accessed: 2024-05-20.

[2] A. Kline, D. Ahner, and R. Hill, "The weapon-target assignment problem," *Computers & Operations Research*, vol. 105, pp. 226–236, 2019.

[3] A. S. Manne, "A target-assignment problem," *Operations Research*, vol. 6, no. 3, pp. 346–351, 1958.

[4] O. Kwon, K. Lee, and S. Park, "Targeting and scheduling problem for field artillery," *Computers & industrial engineering*, vol. 33, no. 3-4, pp. 693–696, 1997.

[5] Z.-J. Lee, C.-Y. Lee, and S.-F. Su, "An immunity-based ant colony optimization algorithm for solving weapon–target assignment problem," *Applied Soft Computing*, vol. 2, no. 1, pp. 39–47, 2002.

[6] Z.-J. Lee, S.-F. Su, and C.-Y. Lee, "Efficiently solving general weapon-target assignment problem by genetic algorithms with greedy eugenics," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 33, no. 1, pp. 113–121, 2003.

[7] X. Zeng, Y. Zhu, L. Nan, K. Hu, B. Niu, and X. He, "Solving weapon-target assignment problem using discrete particle swarm optimization," vol. 1 of *2006 6th World Congress on Intelligent Control and Automation*, pp. 3562–3565, IEEE, 2006.

[8] R. K. Ahuja, A. Kumar, K. C. Jha, and J. B. Orlin, "Exact and heuristic algorithms for the weapon-target assignment problem," *Operations research*, vol. 55, no. 6, pp. 1136–1146, 2007.

[9] O. Kwon, K. Lee, D. Kang, and S. Park, "A branch-and-price algorithm for a targeting problem," *Naval Research Logistics (NRL)*, vol. 54, no. 7, pp. 732–741, 2007.

[10] G. Arslan, J. R. Marden, and J. S. Shamma, "Autonomous vehicle-target assignment: A game-theoretical formulation," *Journal of Dynamic Systems, Measurement, and Control*, vol. 129, pp. 584–596, 04 2007.

[11] O. Karasakal, "Air defense missile-target allocation models for a naval task group," *Computers & Operations Research*, vol. 35, no. 6, pp. 1759–1770, 2008.

[12] L. Wu, H.-y. Wang, F.-x. Lu, and P. Jia, "An anytime algorithm based on modified ga for dynamic weapon-target allocation problem," 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence), 2008.

[13] A. M. Madni and M. Andrecut, "Efficient heuristic approach to the weapon-target assignment problem," *Journal of Aerospace Computing, Information, and Communication*, vol. 6, no. 6, pp. 405–414, 2009.

[14] Y.-H. Cha and Y.-D. Kim, "Fire scheduling for planned artillery attack operations under time-dependent destruction probabilities," *Omega*, vol. 38, no. 5, pp. 383–392, 2010.

[15] O. Karasakal, N. E. Özdemirel, and L. Kandiller, "Anti-ship missile defense for a naval task group," *Naval Research Logistics (NRL)*, vol. 58, no. 3, pp. 304–321, 2011.

[16] C. Zhu, J. Luo, W. Zhang, and Z. Liu, "Ol-dec-mdp model for multiagent online scheduling with a time-dependent probability of success," *Mathematical Problems in Engineering*, vol. 2014, 2014.

[17] R. Taghavi and M. Ranjbar, "Weapon scheduling in naval combat systems for maximization of defense capabilities," *Iranian Journal of Operations Research*, vol. 6, no. 2, pp. 87–99, 2015.

[18] L. Zhanwu, C. Yizhe, K. Yingxin, Y. Haiyan, X. An, and L. You, "Approach to wta in air combat using iafsa-ihs algorithm," *Journal of Systems Engineering and Electronics*, vol. 29, no. 3, pp. 519–529, 2018.

[19] P. Zhao, J. Wang, and L. Kong, "Decentralized algorithms for weapon-target assignment in swarming combat system," *Mathematical Problems in Engineering*, vol. 2019, 2019.

[20] A. C. Andersen, K. Pavlikov, and T. A. Toffolo, "Weapon-target assignment problem: Exact and approximate solution algorithms," *Annals of Operations Research*, vol. 312, no. 2, pp. 581–606, 2022.

[21] Y. Lu and D. Z. Chen, "A new exact algorithm for the weapon-target assignment problem," *Omega*, p. 102138, 2019.

[22] E. Sonuç, "A modified crow search algorithm for the weapon-target assignment problem," *An International Journal of Optimization and Control: Theories & Applications (IJOCTA)*, vol. 10, no. 2, pp. 188–197, 2020.

[23] W. Luo, J. Lü, K. Liu, and L. Chen, "Learning-based policy optimization for adversarial missile-target assignment," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 7, pp. 4426–4437, 2021.

[24] J. Zhang, M. Kong, G. Zhang, and Y. Huang, "Weapon–target assignment using a whale optimization algorithm," *International Journal of Computational Intelligence Systems*, vol. 16, no. 1, p. 62, 2023.

[25] E. Acar, S. Hatipoğlu, and İ. Yılmaz, "A quantum algorithm for solving weapon target assignment problem," *Engineering Applications of Artificial Intelligence*, vol. 125, p. 106668, 2023.

[26] H.-W. Park and H.-L. Choi, "Weapon-target assignment and firing scheduling for rapid engagement with heterogeneous interceptors," *International Journal of Aeronautical and Space Sciences*, vol. 24, no. 3, pp. 890–904, 2023.

[27] X. Chang, J. Shi, Z. Luo, and Y. Liu, "Adaptive large neighborhood search algorithm for multi-stage weapon target assignment problem," *Computers & Industrial Engineering*, vol. 181, p. 109303, 2023.

[28] S. Zou, X. Shi, and S. Song, "Moea with adaptive operator based on reinforcement learning for weapon target assignment," *Electronic Research Archive*, vol. 32, no. 3, pp. 1498–1532, 2024.

[29] D. Khosla, "Hybrid genetic approach for the dynamic weapon-target allocation problem," *Battlespace digitization and network-centric warfare*, vol. 4396, pp. 244–259, 2001.

[30] B. Xin, J. Chen, Z. Peng, L. Dou, and J. Zhang, "An efficient rule-based constructive heuristic to solve dynamic weapon-target assignment problem," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 41, no. 3, pp. 598–606, 2010.

[31] B. Xin, J. Chen, J. Zhang, L. Dou, and Z. Peng, "Efficient decision makings for dynamic weapon-target assignment by virtual permutation and tabu search heuristics," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 40, no. 6, pp. 649–662, 2010.

[32] C. Leboucher, H. Shin, S. Le Ménec, A. Tsourdos, and A. Kotenkoff, "Optimal weapon target assignment based on a geometric approach," *IFAC Proceedings Volumes*, vol. 46, no. 19, pp. 341–346, 2013.

[33] C. Leboucher, H.-S. Shin, P. Siarry, R. Chelouah, S. Le Ménec, and A. Tsourdos, "A two-step optimisation method for dynamic weapon target assignment problem," *Recent Advances on Meta-Heuristics and Their Application to Real Scenarios*, pp. 109–129, 2013.

[34] D. K. Ahner and C. R. Parson, "Optimal multi-stage allocation of weapons to targets using adaptive dynamic programming," *Optimization Letters*, vol. 9, no. 8, pp. 1689–1701, 2015.

[35] M. T. Davis, M. J. Robbins, and B. J. Lunday, "Approximate dynamic programming for missile defense interceptor fire control," *European Journal of Operational Research*, vol. 259, no. 3, pp. 873–886, 2017.

[36] N. Gülpınar, E. Çanakoğlu, and J. Branke, "Heuristics for the stochastic dynamic task-resource allocation problem with retry opportunities," *European Journal of Operational Research*, vol. 266, no. 1, pp. 291–303, 2018.

[37] J. Li, B. Xin, P. M. Pardalos, and J. Chen, "Solving bi-objective uncertain stochastic resource allocation problems by the cvar-based risk measure and decomposition-based multi-objective evolutionary algorithms," *Annals of Operations Research*, pp. 1–28, 2019.

[38] A. Silav, O. Karasakal, and E. Karasakal, "Bi-objective missile rescheduling for

a naval task group with dynamic disruptions," *Naval Research Logistics (NRL)*, vol. 66, no. 7, pp. 596–615, 2019.

[39] D. S. Summers, M. J. Robbins, and B. J. Lunday, "An approximate dynamic programming approach for comparing firing policies in a networked air defense environment," *Computers & Operations Research*, vol. 117, p. 104890, 2020.

[40] J. Zhengrong, L. Faxing, and W. Hangyu, "Multi-stage attack weapon target allocation method based on defense area analysis," *Journal of Systems Engineering and Electronics*, vol. 31, no. 3, pp. 539–550, 2020.

[41] A. Silav, E. Karasakal, and O. Karasakal, "Bi-objective dynamic weapon-target assignment problem with stability measure," *Annals of Operations Research*, pp. 1–19, 2021.

[42] O. Karasakal, E. Karasakal, and A. Silav, "A multi-objective approach for dynamic missile allocation using artificial neural networks for time sensitive decisions," *Soft Computing*, vol. 25, no. 15, pp. 10153–10166, 2021.

[43] L. Kong, J. Wang, and P. Zhao, "Solving the dynamic weapon target assignment problem by an improved multiobjective particle swarm optimization algorithm," *Applied Sciences*, vol. 11, no. 19, p. 9254, 2021.

[44] S. Li, X. He, X. Xu, T. Zhao, C. Song, and J. Li, "Weapon-target assignment strategy in joint combat decision-making based on multi-head deep reinforcement learning," *IEEE Access*, 2023.

[45] C. Liu, J. Li, Y. Wang, Y. Yu, L. Guo, Y. Gao, Y. Chen, and F. Zhang, "A time-driven dynamic weapon target assignment method," *IEEE Access*, vol. 11, pp. 129623–129639, 2023.

[46] O. Karasakal, *Optimal air defense strategies for naval task group*. PhD thesis, Middle East Technical University, 2004.

[47] S. P. Lloyd and H. S. Witsenhausen, "Weapons allocation is np-complete," 1986 summer computer simulation conference, pp. 1054–1058, 1986.

[48] Z. Bogdanowicz, N. Coleman, *et al.*, "Sensor-target and weapon-target pairings

based on auction algorithm," Proceedings of the 11th WSEAS international conference on applied mathematics, pp. 92–96;, Citeseer, 2007.

[49] H. Chen, Z. Liu, Y. Sun, and Y. Li, "Particle swarm optimization based on genetic operators for sensor-weapon-target assignment," vol. 2 of *2012 Fifth International Symposium on Computational Intelligence and Design*, pp. 170–173;, IEEE, 2012.

[50] B. Xin, Y. Wang, and J. Chen, "An efficient marginal-return-based constructive heuristic to solve the sensor–weapon–target assignment problem," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 12, pp. 2536–2547, 2018.

[51] A. Benaskeur, É. Bossé, and D. Blodgett, "Combat resource allocation planning in naval engagements," TR 2005-486, Defence Research and Development Canada Valcartier (Quebec), 2007.

[52] D. E. Blodgett, B. Chaib-draaa, P. Plamondon, P. Kropf, and E. Bossé, "A method to optimize ship maneuvers for the coordination of hardkill and softkill weapons within a frigate," 7th International Command and Control Research Technology Symposium, pp. 1–15, 2002.

[53] A. Benaskeur, É. Bossé, and D. Blodgett, "Multi-agent coordination techniques for naval tactical combat resources management," TR 2006-784, Defence Research and Development Canada Valcartier (Quebec), 2008.

[54] M. Cerruti, M. Pastorino, A. Randazzo, F. Perra, and A. Guagnano, "A radar cross section and radar performance evaluation tool for the early stage ship design (essd) phase," *OCEANS 2015 - Genova*, pp. 1–5, 2015.

[55] R. Bellman, "On a routing problem," *Quarterly of applied mathematics*, vol. 16, no. 1, pp. 87–90, 1958.

[56] MathWorks, *MATLAB Statistics and Machine Learning Toolbox User's Guide*. The MathWorks, Inc., Natick, MA, USA, 2024. Accessed: 2024-05-11.

[57] M. H. Beale, M. T. Hagan, and H. B. Demuth, *MATLAB Deep Learning Toolbox User's Guide*. The MathWorks, Inc., Natick, MA, USA, 2024. Accessed: 2024-05-11.

APPENDICES

A Air Defense Systems of a Warship

In order to describe the problem better, Air Defense Systems of some warships will be explained briefly in this appendix. Figure A.1 shows an Ada class corvette of Turkish Navy.

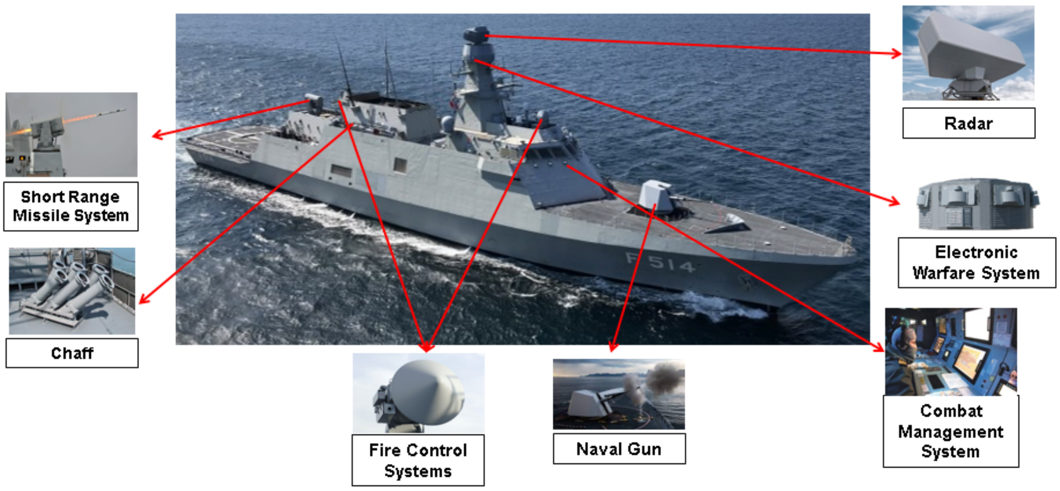


Figure A.1: A Milgem Class Corvette

This ship has surveillance radar systems. Radars are used for the detection of the threats.

EW systems is used to detect and identify the threats. Depending on their capabilities, these systems can also disrupt the threat ASM's track on the ownship and lead the threat to destroy itself. EW systems can work as jamming systems.

This ship has a Chaff countermeasure system. Chaff is deployed at a position between the ownship and the incoming threat and it can distract radar-guided-missiles by creating a false target. EW systems and chaff countermeasure systems are named as soft kill systems.

Fire Control Systems are used for tracking and illumination. Some of the weapon systems are guided by fire control radars to perform the engagements.

Naval gun is artillery mounted on warships.

This ship has a short range missile system. Hence it can be said that this ship has only self defense weapon systems.

And CMS is the brain of the ship. It integrates all of the ship's weapons, data, sensors and other equipment into a single system and it enables fast and correct combat management decisions.

A Gabya class frigate can be seen in Figure A.2. Different from the previous Ada class corvette, she has CIWS and Medium range missile systems.

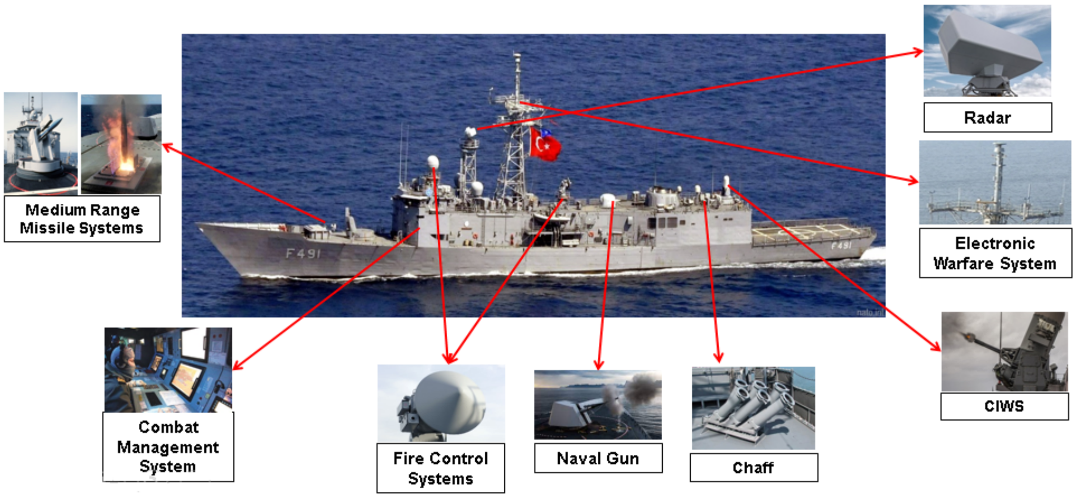


Figure A.2: A Gabya Class Frigatte

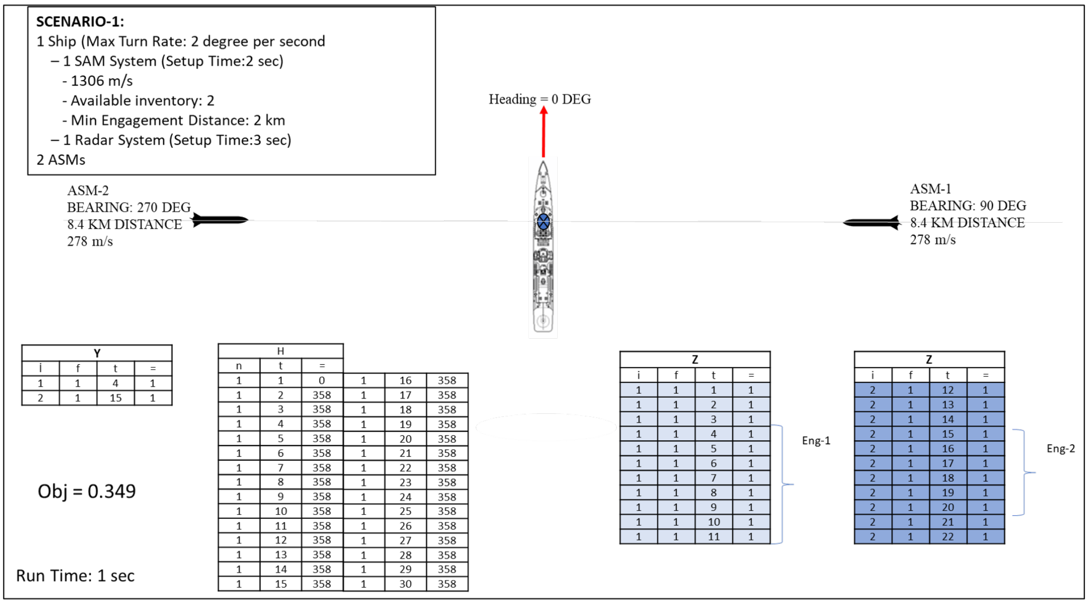
CIWS is a self-defense system. It has a very short range. It provides an ultra-high fire rate of shells and it represents the "last chance" protection against missiles.

Medium range missile systems can cover other friendly units within their effective ranges. Gabya Class Frigates can provide area defense for other ships.

Platforms have to provide their defenses by using their limited resources effectively. Efficient use of defense resources is critical and that would affect the final survival of the ship. Therefore, all defense systems in the task group, Soft Kill or Hard Kill, need to be considered in the NADP problem.

B Example Scenarios

In the first scenario described in Figure B.1, there are two incoming ASMs and one ship. The ship has one SAM system with 2 available inventory and one radar system. The solution obeys the radar/weapon setup times, radar illumination, and all other requirements.



Y			
i	f	t	=
1	1	4	1
2	1	15	1

H					
n	t	=			
1	1	0	1	16	358
1	2	358	1	17	358
1	3	358	1	18	358
1	4	358	1	19	358
1	5	358	1	20	358
1	6	358	1	21	358
1	7	358	1	22	358
1	8	358	1	23	358
1	9	358	1	24	358
1	10	358	1	25	358
1	11	358	1	26	358
1	12	358	1	27	358
1	13	358	1	28	358
1	14	358	1	29	358
1	15	358	1	30	358

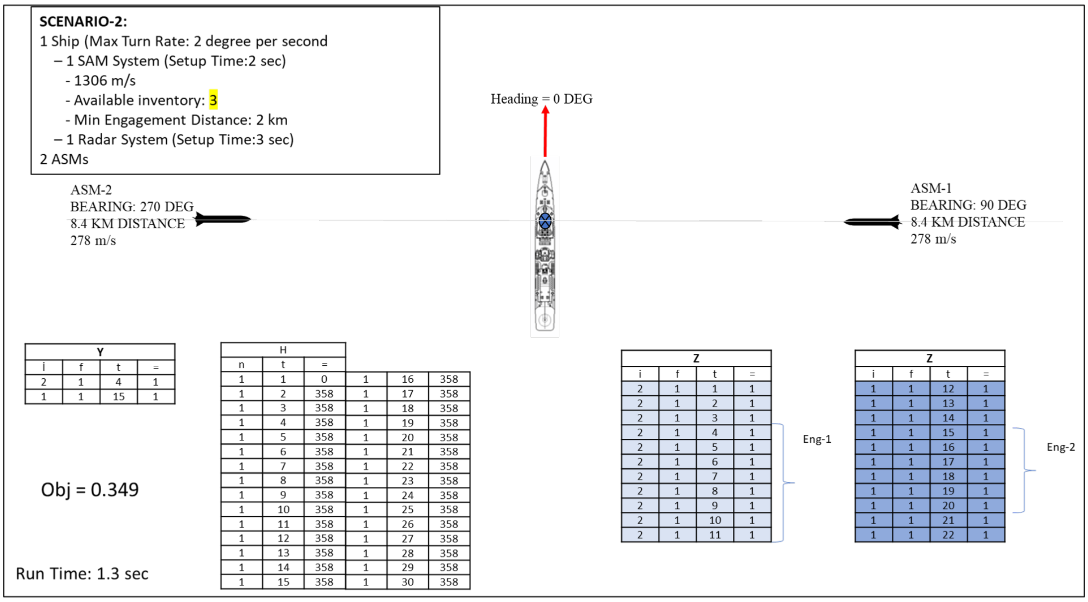
Z			
i	f	t	=
1	1	1	1
1	1	2	1
1	1	3	1
1	1	4	1
1	1	5	1
1	1	6	1
1	1	7	1
1	1	8	1
1	1	9	1
1	1	10	1
1	1	11	1

Z			
i	f	t	=
2	1	12	1
2	1	13	1
2	1	14	1
2	1	15	1
2	1	16	1
2	1	17	1
2	1	18	1
2	1	19	1
2	1	20	1
2	1	21	1
2	1	22	1

Figure B.1: Scenario-1

As can be seen in the tables above, first engagement starts against ASM-1 at time slot 4 and second engagement starts against ASM-2 at time slot 15. Ship's heading is 358 throughout the scenario. The radar is assigned to ASM-1 between time slots 1-11 and to ASM-2 between time slots 12-22. OBJ-1 is used as the objective function of the model.

In the second scenario, the available inventory increases to 3. However, the ship can not use the third ASM since there is not enough time to complete engagement before the ASMs enter the SAM minimum engagement range.



Y			
i	f	t	=
2	1	4	1
1	1	15	1

H					
n	t	=			
1	1	0	1	16	358
1	2	358	1	17	358
1	3	358	1	18	358
1	4	358	1	19	358
1	5	358	1	20	358
1	6	358	1	21	358
1	7	358	1	22	358
1	8	358	1	23	358
1	9	358	1	24	358
1	10	358	1	25	358
1	11	358	1	26	358
1	12	358	1	27	358
1	13	358	1	28	358
1	14	358	1	29	358
1	15	358	1	30	358

Z			
i	f	t	=
2	1	1	1
2	1	2	1
2	1	3	1
2	1	4	1
2	1	5	1
2	1	6	1
2	1	7	1
2	1	8	1
2	1	9	1
2	1	10	1
2	1	11	1

Z			
i	f	t	=
1	1	12	1
1	1	13	1
1	1	14	1
1	1	15	1
1	1	16	1
1	1	17	1
1	1	18	1
1	1	19	1
1	1	20	1
1	1	21	1
1	1	22	1

Figure B.2: Scenario-2

In the third scenario, the radar system has blind sectors. As can be seen in the solution, the ship maneuvers toward the port side (below 350) to be able to make an engagement to ASM-1.

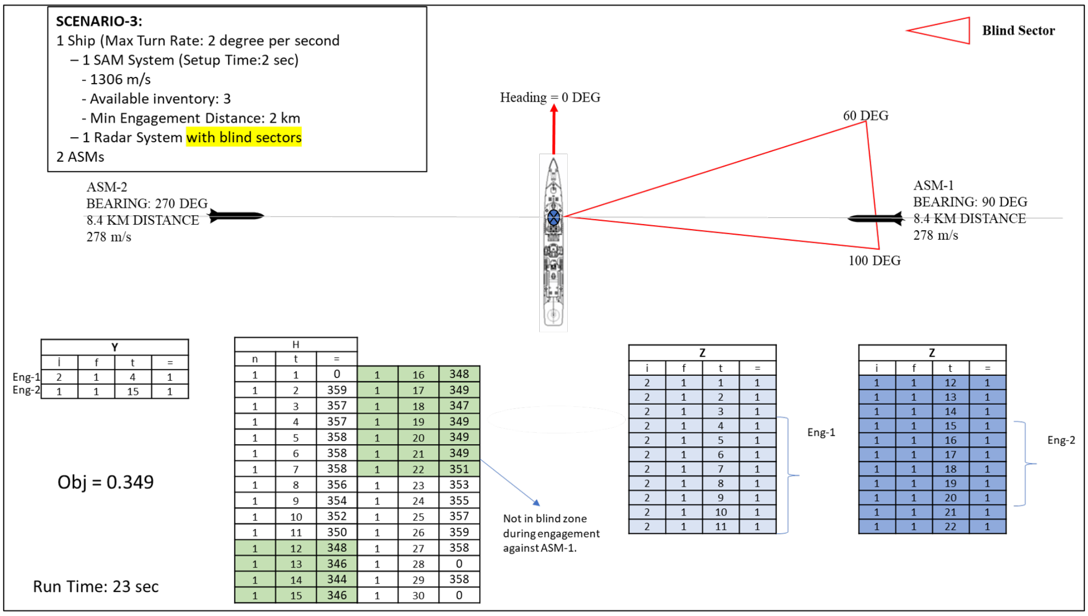


Figure B.3: Scenario-3

In the fourth scenario, the weapon system's minimum engagement range is decreased to 1 kilometer. In this case, the ship has enough time to use the third SAM.

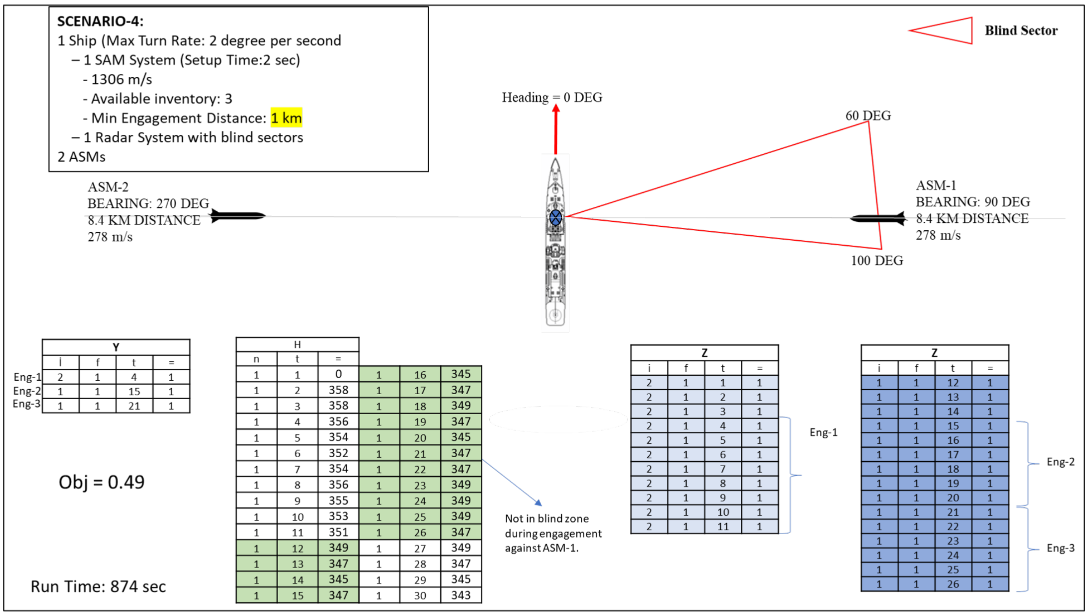
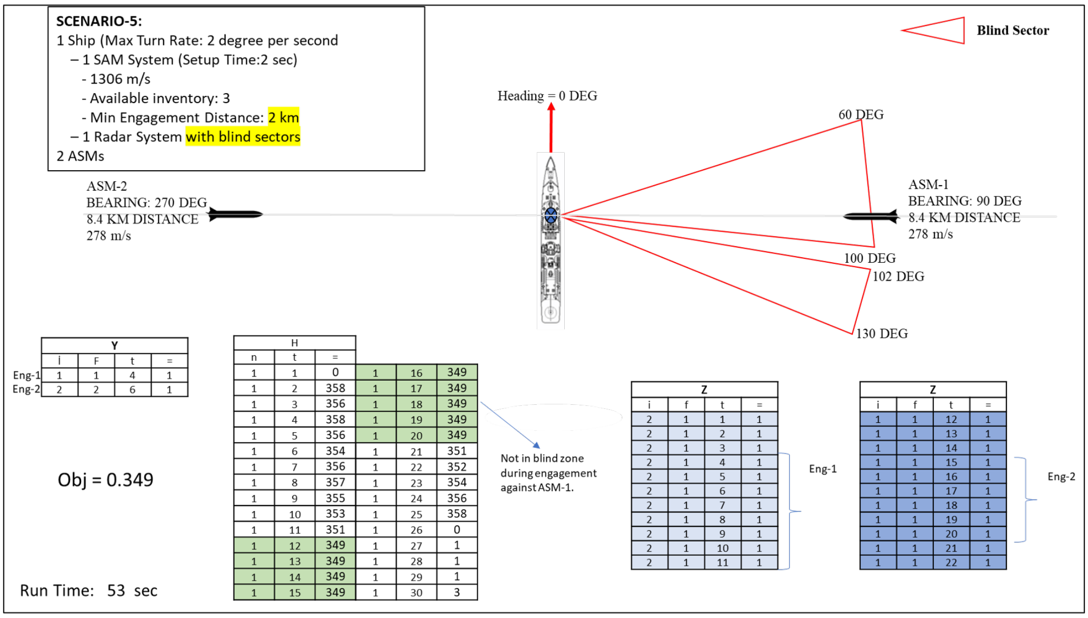


Figure B.4: Scenario-4

In the fifth scenario, the radar system has more blind sectors. She needs to have heading exactly 349 degrees. And this can be seen in the solution.



Y	i	F	t	=
Eng-1	1	1	4	1
Eng-2	2	2	6	1

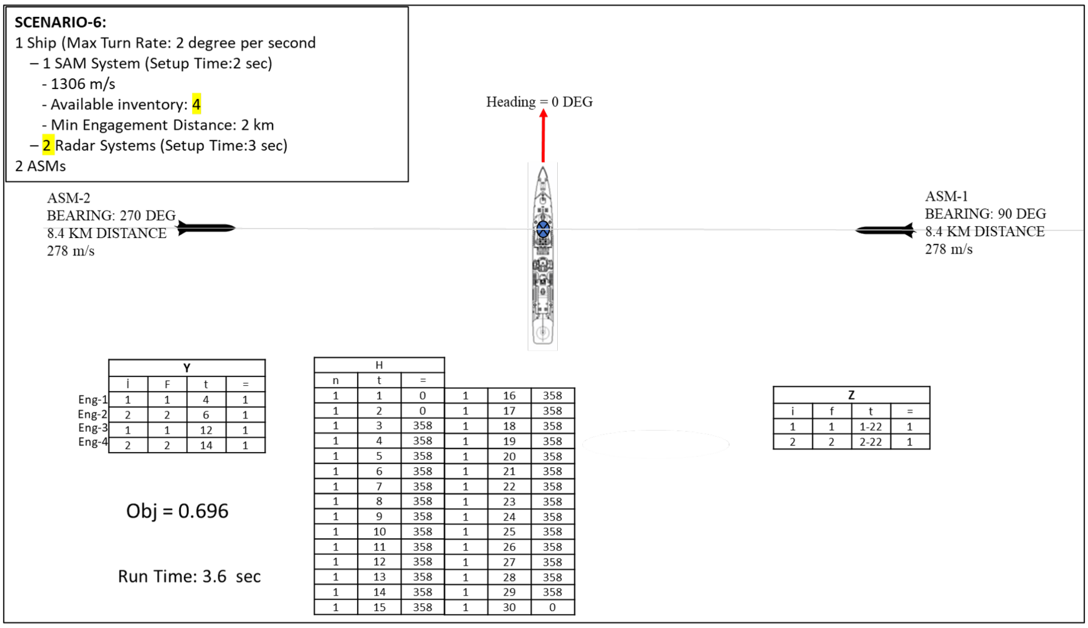
H: n	t	=	n	t	=
1	1	0	1	16	349
1	2	358	1	17	349
1	3	356	1	18	349
1	4	358	1	19	349
1	5	356	1	20	349
1	6	354	1	21	351
1	7	356	1	22	352
1	8	357	1	23	354
1	9	355	1	24	356
1	10	353	1	25	358
1	11	351	1	26	0
1	12	349	1	27	1
1	13	349	1	28	1
1	14	349	1	29	1
1	15	349	1	30	3

Z: i	f	t	=
2	1	1	1
2	1	2	1
2	1	3	1
2	1	4	1
2	1	5	1
2	1	6	1
2	1	7	1
2	1	8	1
2	1	9	1
2	1	10	1
2	1	11	1

Z: i	f	t	=
1	1	12	1
1	1	13	1
1	1	14	1
1	1	15	1
1	1	16	1
1	1	17	1
1	1	18	1
1	1	19	1
1	1	20	1
1	1	21	1
1	1	22	1

Figure B.5: Scenario-5

In the sixth scenario, the ship has two radar systems and 4 available SAM inventory. The solution obeys the radar/weapon setup times, radar illumination, and all other requirements.



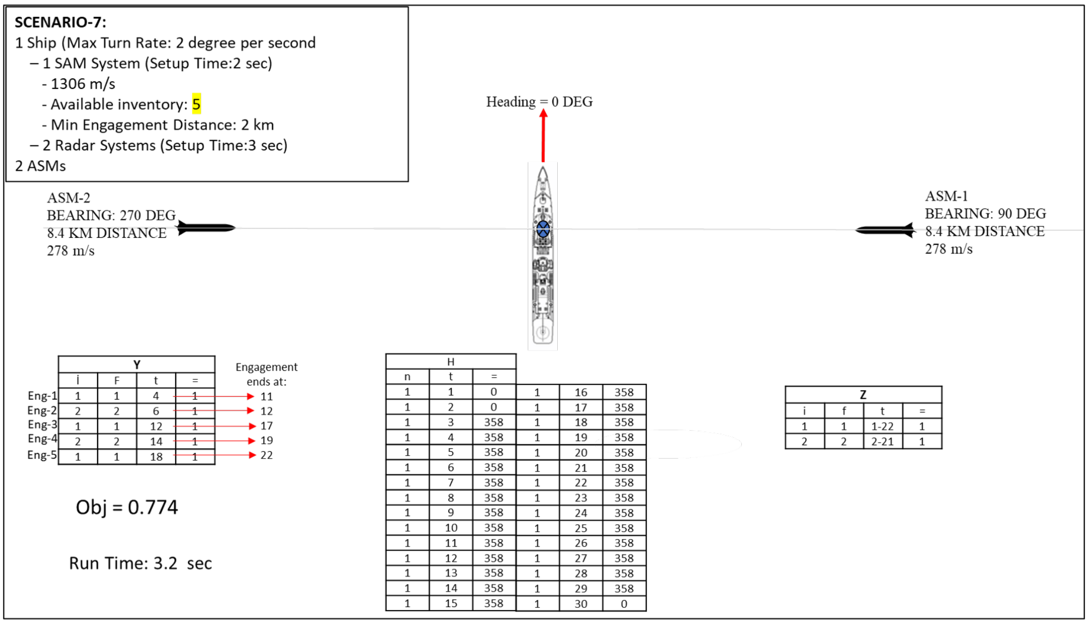
Y	i	F	t	=
Eng-1	1	1	4	1
Eng-2	2	2	6	1
Eng-3	1	1	12	1
Eng-4	2	2	14	1

H: n	t	=	n	t	=
1	1	0	1	16	358
1	2	0	1	17	358
1	3	358	1	18	358
1	4	358	1	19	358
1	5	358	1	20	358
1	6	358	1	21	358
1	7	358	1	22	358
1	8	358	1	23	358
1	9	358	1	24	358
1	10	358	1	25	358
1	11	358	1	26	358
1	12	358	1	27	358
1	13	358	1	28	358
1	14	358	1	29	358
1	15	358	1	30	0

Z: i	f	t	=
1	1	1-22	1
2	2	2-22	1

Figure B.6: Scenario-6

In the seventh scenario, the ship has 5 available SAM inventory. The solution obeys the radar/weapon setup times, radar illumination, and all other requirements.



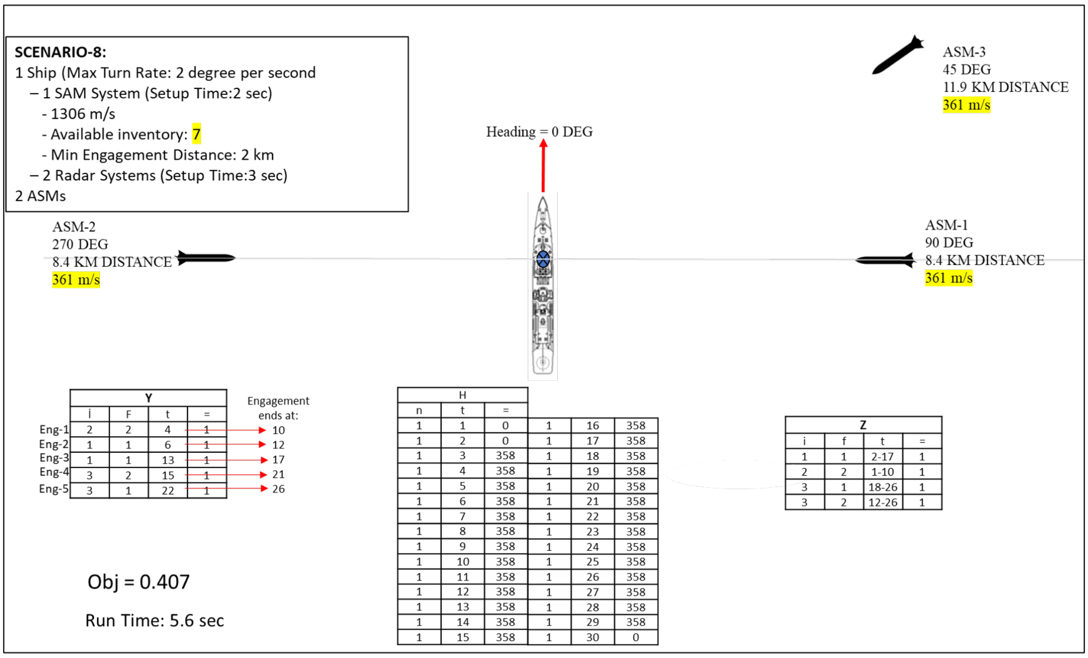
	Y			
	i	F	t	=
Eng-1	1	1	4	1
Eng-2	2	2	6	1
Eng-3	1	1	12	1
Eng-4	2	2	14	1
Eng-5	1	1	18	1

H					
n	t	=			
1	1	0	1	16	358
1	2	0	1	17	358
1	3	358	1	18	358
1	4	358	1	19	358
1	5	358	1	20	358
1	6	358	1	21	358
1	7	358	1	22	358
1	8	358	1	23	358
1	9	358	1	24	358
1	10	358	1	25	358
1	11	358	1	26	358
1	12	358	1	27	358
1	13	358	1	28	358
1	14	358	1	29	358
1	15	358	1	30	0

Z			
i	f	t	=
1	1	1-22	1
2	2	2-21	1

Figure B.7: Scenario-7

In the eighth scenario, there are 3 incoming ASMs and the ship has 7 available SAM inventory. The solution obeys the radar/weapon setup times, radar illumination, and all other requirements.



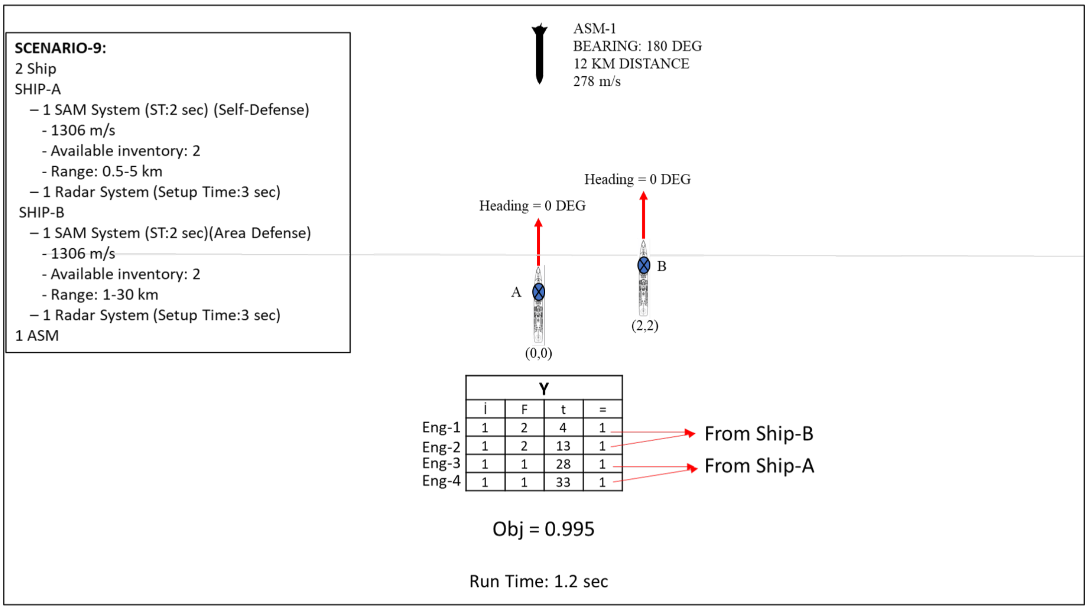
	Y			
	i	F	t	=
Eng-1	2	2	4	1
Eng-2	1	1	6	1
Eng-3	1	1	13	1
Eng-4	3	2	15	1
Eng-5	3	1	22	1

H					
n	t	=			
1	1	0	1	16	358
1	2	0	1	17	358
1	3	358	1	18	358
1	4	358	1	19	358
1	5	358	1	20	358
1	6	358	1	21	358
1	7	358	1	22	358
1	8	358	1	23	358
1	9	358	1	24	358
1	10	358	1	25	358
1	11	358	1	26	358
1	12	358	1	27	358
1	13	358	1	28	358
1	14	358	1	29	358
1	15	358	1	30	0

Z			
i	f	t	=
1	1	2-17	1
2	2	1-10	1
3	1	18-26	1
3	2	12-26	1

Figure B.8: Scenario-8

In the ninth scenario, there are 1 incoming ASM and two ships. Both ships have two available SAM inventory. The solution obeys the shoot-look-shoot policy.



	i	F	t	=
Eng-1	1	2	4	1
Eng-2	1	2	13	1
Eng-3	1	1	28	1
Eng-4	1	1	33	1

Figure B.9: Scenario-9

In the tenth scenario, the ship's RCS data is also included in the parameters. There are 2 incoming ASMs and one ship. The ship has 2 available SAM inventory. In order to increase its survivability, the ship has to show its 40-degree aspect to the ASMs. The ship's maneuvering that provides this can be seen in the solution.

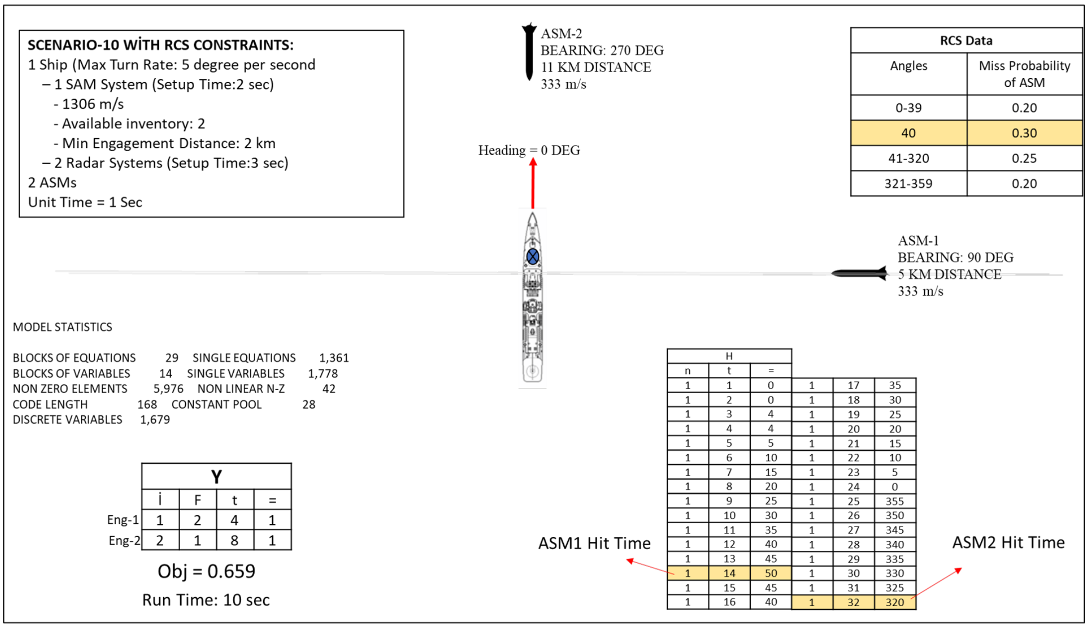


Figure B.10: Scenario-10

C Linearization of the Model

In the following subsections, the formulations of the linearized NADP problem for 3 different objective functions are given.

C.1 Objective Function-1: Maximization of the probability of no-leaker for the entire TG

The only nonlinear part of the model is the objective function given below:

$$max \prod_{i \in A} \left(1 - PK_i \prod_{t \in T, f \in F} (1 - p_{ift})^{Y_{ift}} \right)$$

The nonlinearity in the model can be linearized by using logarithms and piecewise linear functions. Let us define h_i as the probability of no-leaker for ASM i.

$$h_i = \left(1 - PK_i \prod_{t \in T, f \in F} (1 - p_{ift})^{Y_{ift}} \right)$$

The objective function can be restated as:

$$max \prod_{i \in A} h_i$$

And a new set of constraint can be added into the model as follows:

$$\left(1 - PK_i \prod_{t \in T, f \in F} (1 - p_{ift})^{Y_{ift}}\right) \geq h_i, \forall i \in A$$

Taking the logarithms of the objective function and both sides of the constraint does not change the optimal solution since $ln(a) \leq ln(b)$ if and only if $a \leq b$:

$$max \sum_{i \in A} ln(h_i)$$

$$-ln(PK_i) - \sum_{t \in T, f \in F} ln(1 - p_{ift})Y_{ift} \geq -ln(1 - h_i), \forall i \in A$$

$ln(PK_i)$ term creates non-linearity in the above inequality. With a modification in the decision variable PK_i in the constraint set 4.22, this non-linearity can be eliminated. Let us define $lnPK_i$ as new decision variable to replace $ln(PK_i)$. In the right-hand side of constraint set 4.22, $(1 - rcs_{i\alpha})$ part can be written as $ln(1 - rcs_{i\alpha})$ since $BR_{i\alpha}$ decision variable takes the value of 1 for only one angle α. $lnPK_i$ can be calculated as the same value of $ln(PK_i)$ in this way. Modified version of the constraint set 4.22 is given below:

$$lnPK_i = \sum_{\alpha \in [0,359]} ln(1 - rcs_{i\alpha})BR_{i\alpha} \qquad \forall i \in A,$$

And the new constraint set can be restated as:

$$-lnPK_i + \sum_{t \in T, f \in F} a_{ift}Y_{ift} \geq b_i, \forall i \in A$$

where $a_{ift} = -ln(1 - p_{ift})$ and $b_i = -ln(1 - h_i)$.

Further simplification procedure is presented by Karasakal (2004)[46] using piece-wise linearization and the relationship between $ln(h)$ in the objective function and $-ln(1-h_i)$ in the right hand side of the new constraint.

The resulting model is given below:

$$max \sum_{i \in A} (c^1 b_i^1 + c^2 b_i^2 + c^3 b_i^3 + c^4 b_i^4)$$

$$s.t. - lnPK_i + \sum_{t \in T, f \in F} a_{ift} Y_{ift} \geq b_i^1 + b_i^2 + b_i^3 + b_i^4, \qquad \forall i \in A \qquad (8.1)$$

$$0 \leq b_i^1 \leq Z_1, \qquad \forall i \in A \qquad (8.2)$$

$$0 \leq b_i^2 \leq Z_2 - Z_1, \qquad \forall i \in A \qquad (8.3)$$

$$0 \leq b_i^3 \leq Z_3 - Z_2, \qquad \forall i \in A \qquad (8.4)$$

$$0 \leq b_i^4 \leq Z_4 - Z_3, \qquad \forall i \in A \qquad (8.5)$$

and Constraint Sets 1-34. Here $c^1, c^2, c^3, c^4, Z_1, Z_2, Z_3, Z_4$ values [1] are determined according the relationship between $ln(h_i)$ and $-ln(1-h_i)$ seen in the Figure C.1.

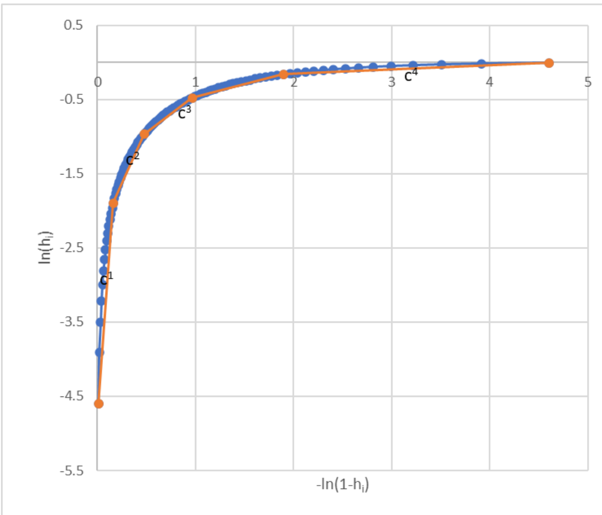


Figure C.1: Relationship between $ln(h_i)$ and $-ln(1-h_i)$

Note that this is not the exact linearization of the nonlinear model. However this

[1] $c^1 = 17.761$ $c^2 = 2.848$ $c^3 = 0.979$ $c^4 = 0.332$ $c^5 = 0.053$ $Z_1 = 0.163$ $Z_2 = 0.478$ $Z_3 = 0.968$ $Z_4 = 1.897$ $Z_5 = 4.6052$

provides an approximation for the nonlinear model and helps to find good enough solutions.

C.2 Objective Function-2: Minimization of the maximum hit probability of the threats

The nonlinear part in the objective function-2 formulation is the maximum hit probability constraint given below:

$$PK_i \prod_{t \in T, f \in F} (1 - p_{ift})^{Y_{ift}} \leq W \qquad \forall i \in A$$

Taking the logarithms of the objective function and both sides of the constraint does not change the optimal solution:

$$min \; ln(W)$$

subject to

$$lnPK_i + \sum_{t \in T, f \in F} ln(1 - p_{ift}) Y_{ift} \leq ln(W) \qquad \forall i \in A$$

However, $ln(W)$ still creates nonlinearity in the above formulation. Let us define lnW as a new decision variable to replace $ln(W)$. This operation does not change the optimal solution because both the objective function and the right hand side of the constraint has $ln(W)$. The resulting model is given below:

$$min \; lnW$$

subject to

$$lnPK_i - \sum_{t \in T, f \in F} a_{ift} Y_{ift} \leq lnW \qquad \forall i \in A \tag{8.6}$$

and Constraint Sets 4.1-4.34.

This formulation provides exact linearization of the objective function-2 nonlinear model.

C.3 Objective Function-3: Minimization of the value-weighted total hit probability of the threats

The nonlinear part of the model is the objective function given below:

$$min \sum_{i \in A} v_i \left(PK_i \prod_{t \in T, f \in F} (1 - p_{ift})^{Y_{ift}} \right)$$

Let us define h_i as the hit probability of ASM i.

$$h_i = PK_i \prod_{t \in T, f \in F} (1 - p_{ift})^{Y_{ift}}$$

The objective function can be restated as:

$$min \sum_{i \in A} v_i h_i$$

And a new set of constraint can be added into the model as follows:

$$PK_i \prod_{t \in T, f \in F} (1 - p_{ift})^{Y_{ift}} \leq h_i, \forall i \in A$$

Taking the logarithms of the both sides of the constraint does not change the optimal solution:

$$-lnPK_i - \sum_{t \in T, f \in F} ln(1 - p_{ift})Y_{ift} \geq -ln(h_i), \forall i \in A$$

And nonlinear part can be linearized using piecewise linear functions and the relationship between h_i in the objective function and $-ln(h_i)$ in the right hand side of the new constraint. The resulting model is given below:

$$min \sum_{i \in A} v_i(c^1 b_i^1 + c^2 b_i^2 + c^3 b_i^3 + +c^4 b_i^4)$$

$$s.t. - lnPK_i + \sum_{t \in T, f \in F} a_{ift} Y_{ift} \geq b_i^1 + b_i^2 + b_i^3 + b_i^4, \quad \forall i \in A \quad (8.7)$$

$$0 \leq b_i^1 \leq Z_1, \quad \forall i \in A \quad (8.8)$$

$$0 \leq b_i^2 \leq Z_2 - Z_1, \quad \forall i \in A \quad (8.9)$$

$$0 \leq b_i^3 \leq Z_3 - Z_2, \quad \forall i \in A \quad (8.10)$$

$$0 \leq b_i^4 \leq Z_4 - Z_3, \quad \forall i \in A \quad (8.11)$$

and Constraint Sets 4.1-4.34. Here $c^1, c^2, c^3, c^4, Z_1, Z_2, Z_3, Z_4$ values [2] are determined according the relationship between h_i and $-ln(h_i)$ seen in the Figure C.2.

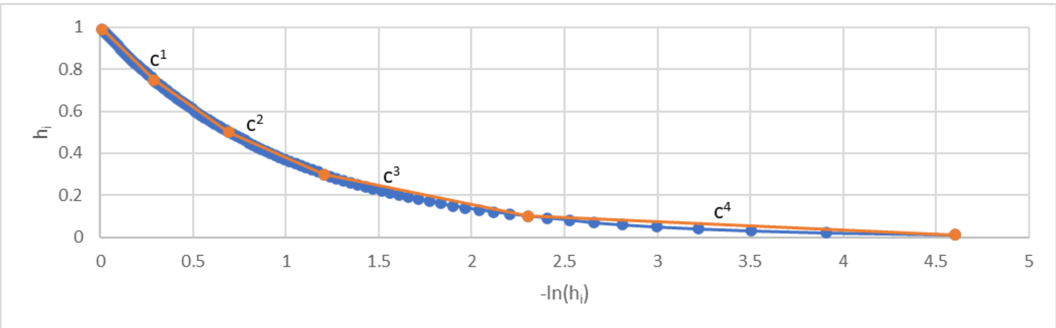


Figure C.2: Relationship between h_i and $-ln(h_i)$

[2] $c^1 = -0.864$ $c^2 = -0.612$ $c^3 = -0.387$ $c^4 = -0.178$ $c^5 = -0.036$
$Z_1 = 0.288$ $Z_2 = 0.693$ $Z_3 = 1.204$ $Z_4 = 2.303$ $Z_5 = 4.6052$

Similar to the linearization of the objective function-1 nonlinear model, this formulation does not provide exact linearization of the objective function-3 model.

D Generation of the Scenarios

The parameter generation structure specified in the Table D.1 is utilized in order to create the scenarios to be used in the computational experiments.

In the computational experiments, these parameters listed in the table are determined randomly within the specified intervals. Then other model parameters (J_{it}, U_{ift}, μ_{if}, p_{ift}, θ_{int}, etc.) are calculated and GAMS input file is prepared by MATLAB. GAMS or the developed algorithm solves the problem using this input file, and then MATLAB reports the result. Figure-D.1 summarizes the structure of how computational experiments are carried out.

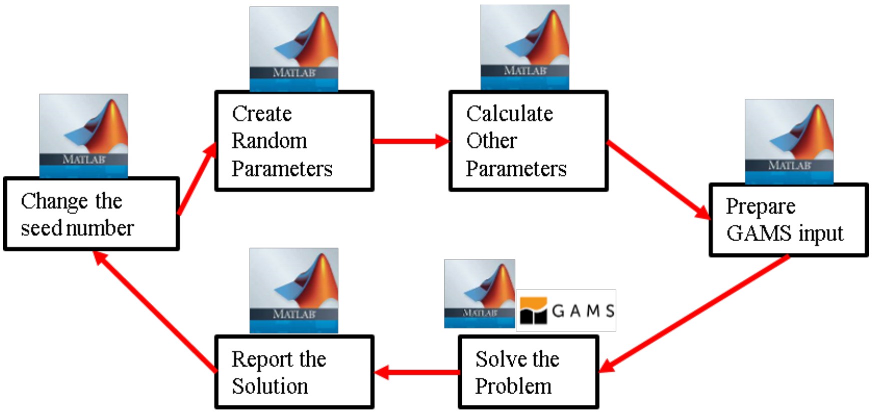


Figure D.1: Using MATLAB and GAMS in the Experiments

It may be useful to explain the approach of determining the hit probabilities of friendly weapon systems used in the scenario creation structure. It is assumed that a threat whose initial position, speed and route is known will not change its route and speed during its operation. In real case, ASMs can make waypoints during their courses toward the target. Therefore, any change in the threat parameters will be considered as a new scenario for the NADP problem.

A meeting point is determined where the weapon system will have the maximum kill probability for a given threat. Then, kill probabilities are determined using the

Table D.1: List of parameters for scenario generation

Parameter	**Value Interval**
Number of Ships	$\{1, 2, ..., 8\}$
Maximum Turn Rate of Ships	$\{2, 3, ..., 9\}$ degree per second
Initial position of the ships	$[-10, 10]$ nautical miles (NM) x-y coord.
Initial heading of the ships	$\{0, 1, ..., 359\}$ degree
Number of Weapon Systems on Each Ship	$\{1, 2\}$
Initial Inventory for Each Weapon System	$\{1, 2, .., 5\}$
Speed of the Weapon Systems	$[0.8, 4]$ Mach per hour
Minimum Distance for Engagements	$[1, 5]$ NM
Maximum Distance for Engagements	$[10, 40]$ NM
Illumination Type of the Weapon Systems (All weapons are type 1 for now)	1 - Semi Active HAW-CWI 2 - Semi Active MCG-CWI 3 - Semi Active MCG-ICWI 4 - Fire and Forget
Launch Delay of the Weapon Systems	$\{1, 2, 3\}$ seconds
Max Efficiency Range of the Weapon Systems	[Min Dist,(Min Dist+Max Dist)/2] NM
PK Value in the Max Efficiency Range	$[0.4, 0.7]$ probability
Kill Probability Change per Km Range	$[0.005, 0.01]$ probability
Setup Time Between Two Consecutive Fires	$\{1, 2, 3, 4\}$ seconds
Weapon System's Defense Capability	1 - Self Defense 2 - Area Defense
Number of Radar Systems on each ship	$\{1, 2\}$
Radar Maximum Distance for Tracking	$[25, 50]$ NM
Radar Setup Time for Tracking the Threat	$\{1, 2, 3\}$ seconds
Radar/Weapon Blind Sectors	For each weapon or radar system; With equal probabilities, no blind sector Or randomly selected 30 degrees
Number of the Threats	{1,2,..., 4 x Number of Ships}
Initial position of the Threats	$[-50, 50]$ excluding $[-10, 10]$ NM x-y coord.
Speed of the Threats	$[0.8, 4]$ Mach per hour
Value of the Ships	$[0, 1]$ (weights are normalized)
Value of the Threats	$[0, 1]$ (target ship's value is taken)
Radar Cross Section Probability Values	$[0.1, 0.3]$

assumption that the kill probability decreases linearly as the distance to this point increases. In real case, threat/engagement assessment is done by much more complex algorithms on the ship's combat management systems. Modeling the kill probability involves considering various factors, such as target distance, attack angle, weapon and threat's technical capabilities. However, detailed modeling of these factors is beyond the scope of this work. We assumed that this structure is sufficient for this study.

E Parameters of the Stage 1.1 Genetic Algorithm

Many experiments have been carried out to develop the best algorithm implementation and to select the most suitable parameters in order to increase the performance of the genetic algorithm. Some tested values for the parameters are presented below. The final versions of the parameters used in the algorithm are marked in bold.

A complete design of experiments for parameter setting optimization was not performed due to the complexity and the potential for significantly increased study time. Optimizing the GA's parameters can often be an exhaustive process, requiring numerous trials and fine-tuning. Therefore, it was decided that a complete design of the experiments was not within the scope of this study. Instead, best-practice values from the trials were adopted to keep the parameters at reasonable levels.

Table E.1: Parameters of the Stage 1.1 Genetic Algorithm

Population Size	20, 30, 40, **50**, 60 Start with a number and increase it by 1 at each iteration max($8(\|S\|+\|R\|), 50$) min ($5(\|A\|\|F\|), 50$) max($10\|S\|+10\|R\|, 50$) min(max($20, 4\|A\|\|N\|), 40$)
Initial Population Size	**max(Population Size, min(**$10\|A\|\|F\|$**,500))** max(Population Size, min($20\|A\|\|F\|$,500)) 50, 60, 70, 80, 90, 100, 200, Population Size
Number of Crossovers in each iteration	15, 20, 25, 30, **35**, 40, 45, 50
Crossover Selection rule	**Random**
Mutation rate	0.15, 0.20, **0.25**, 0.30, 0.35, 0.40, 0.50, 0.60, 0.70
Number of new solutions to be added to the population at each iteration	0, 3, **5**, 8, 10
Convergence Criterion	20, **30**, 40, M iterations without improvement (big M, it means this criterion is not used)
Maximum number of iterations	**100**, 200

F Detailed Comparison Results of the Heuristic Algorithms in MxN NADP Scenarios

Detailed results of Table 5.8 are presented in this appendix.

Table F.1: MxN NADP Scenario Comparison Results (OBJ-1)

	Number of Ships	Number of Threats	OBJ-1 Prob. of No Leaker					
			Greedy Algorithm	Linearized Model	Decomposition Approach	$HA^{GA+EN+DP}$	HA^{GA+DP}	$HA^{GA+EN+DP}$ (1 sec limit)
S-1	1	1	0.6115	0.6115	0.6115	0.6115	0.6115	0.6115
S-2	1	1	0.7334	0.7334	0.7334	0.7334	0.7334	0.7334
S-3	1	2	0.3365	0.3365	0.3365	0.3365	0.3365	0.3365
S-4	1	2	0.4035	0.4045	0.4019	0.4027	0.4027	0.4027
S-5	1	3	0.1875	0.4370	0.4370	0.4370	0.4370	0.4370
S-6	1	3	0.0410	0.0492	0.0492	0.0492	0.0492	0.0492
S-7	1	4	0.1769	0.4255	0.4538	0.4366	0.4359	0.4396
S-8	2	1	0.8806	0.8806	0.8806	0.8806	0.8806	0.8806
S-9	2	2	0.6438	0.7374	0.7374	0.7374	0.7374	0.7374
S-10	2	2	0.2951	0.2951	0.2951	0.2951	0.2951	0.2951
S-11	2	3	0.6273	0.7266	0.7291	0.7313	0.7313	0.7313
S-12	2	3	0.4038	0.4817	0.4817	0.4817	0.4817	0.4631
S-13	2	4	0.4316	0.6622	0.6625	0.6397	0.6397	0.6388
S-14	2	5	0.0424	0.0563	0.0563	0.0563	0.0563	0.0563
S-15	3	1	0.7562	0.7562	0.7562	0.7562	0.7385	0.7562
S-16	3	2	0.8125	0.8187	0.8187	0.8187	0.8108	0.8187
S-17	3	3	0.2873	0.4656	0.4656	0.4656	0.4656	0.4656
S-18	3	4	0.3780	0.5955	0.5955	0.5955	0.5955	0.5874
S-19	3	4	0.1048	0.1226	0.1226	0.1226	0.1226	0.1226
S-20	3	5	0.0679	0.1685	0.1685	0.1685	0.1685	0.1685
S-21	3	6	0.0748	0.1626	0.1657	0.1381	0.1381	0.1431
S-22	4	2	0.5373	0.6175	0.6175	0.6175	0.6175	0.6175
S-23	4	3	0.6453	0.7626	0.7626	0.7591	0.7373	0.7453
S-24	4	3	0.6052	0.7538	0.7538	0.7538	0.7538	0.7537
S-25	4	4	0.3178	0.4408	0.4518	0.4408	0.4408	0.4314
S-26	4	5	0.2090	0.3154	0.3154	0.3142	0.3142	0.3100
S-27	4	6	0.0382	0.0819	0.0819	0.0780	0.0636	0.0780
S-28	4	7	0.1385	0.1502	0.1432	0.1710	0.1710	0.1697
S-29	5	3	0.4493	0.6643	0.6568	0.6558	0.6558	0.6482
S-30	5	4	0.4315	0.4872	0.4880	0.4849	0.4834	0.4550
S-31	5	4	0.4595	0.5063	0.5063	0.5083	0.5077	0.4889
S-32	5	5	0.3906	0.5440	0.5441	0.5440	0.5440	0.5429
S-33	5	6	0.2833	0.3614	0.3617	0.3091	0.3074	0.2995
S-34	5	7	0.0951	0.2760	0.2720	0.2555	0.2555	0.2320
S-35	5	8	0.0221	0.0574	0.0587	0.0550	0.0550	0.0296
S-36	6	4	0.5874	0.7547	0.7456	0.7126	0.7126	0.7488
S-37	6	5	0.2658	0.4556	0.4556	0.4433	0.4433	0.4455
S-38	6	6	0.3395	0.3964	0.4028	0.3059	0.3054	0.3059
S-39	6	7	0.1200	0.2259	0.2259	0.2170	0.2170	0.1866
S-40	6	7	0.1660	0.2219	0.2263	0.2212	0.2202	0.1768
S-41	6	8	0.0534	0.1147	0.1075	0.1105	0.1104	0.0862
S-42	6	9	0.0959	0.1240	0.1287	0.1272	0.1272	0.1208
S-43	7	5	0.4296	0.5714	0.5446	0.5340	0.5340	0.5294
S-44	7	6	0.2096	0.2680	0.2959	0.2473	0.2473	0.2527
S-45	7	7	0.1190	0.2364	0.2364	0.2364	0.2364	0.2022
S-46	7	8	0.2320	0.3891	0.3668	0.3229	0.3066	0.2622
S-47	7	9	0.0148	0.0692	0.0841	0.0633	0.0631	0.0633
S-48	7	10	0.0155	0.0294	0.0294	0.0222	0.0222	0.0145
S-49	8	6	0.1502	0.2449	0.2300	0.1930	0.1930	0.1369
S-50	8	8	0.3284	0.3745	0.3825	0.3322	0.3320	0.2543
Average			**0.3209**	**0.4084**	**0.4087**	**0.3986**	**0.3969**	**0.3892**
Percent Deviation % From the Best Solution (Average)			**28.32**	**1.25**	**0.88**	**4.59**	**5.19**	**9.11**
Number of Best Solutions			**6**	**33**	**38**	**22**	**19**	**16**

Table F.2: MxN NADP Scenario Comparison Results (OBJ-2)

	Number of Ships	Number of Threats	OBJ-2 Min of Max Hit Prob.					
			Greedy Algorithm	Linearized Model	Decomposition Approach	$HA^{GA+EN+DP}$	HA^{GA+DP}	$HA^{GA+EN+DP}$ (1 sec limit)
S-1	1	1	0.3885	0.3885	0.3885	0.3885	0.3885	0.3885
S-2	1	1	0.2666	0.2666	0.2666	0.2666	0.2666	0.2666
S-3	1	2	0.4350	0.4350	0.4350	0.4350	0.4350	0.4350
S-4	1	2	0.4073	0.4043	0.3880	0.3851	0.3851	0.3851
S-5	1	3	0.7034	0.3170	0.3170	0.3170	0.3170	0.3170
S-6	1	3	0.7077	0.7077	0.7077	0.7077	0.7077	0.7077
S-7	1	4	0.7036	0.2971	0.2642	0.2629	0.2629	0.2629
S-8	2	1	0.1194	0.1194	0.1194	0.1194	0.1194	0.1194
S-9	2	2	0.3452	0.2424	0.2424	0.2424	0.2424	0.2424
S-10	2	2	0.5358	0.5358	0.5358	0.5358	0.5358	0.5358
S-11	2	3	0.2252	0.1027	0.1101	0.1147	0.1147	0.1147
S-12	2	3	0.3189	0.2640	0.2640	0.2640	0.2640	0.2561
S-13	2	4	0.2495	0.1391	0.1395	0.1371	0.1371	0.1371
S-14	2	5	0.5498	0.5498	0.5498	0.5498	0.5498	0.5498
S-15	3	1	0.2438	0.2438	0.2438	0.2438	0.2615	0.2438
S-16	3	2	0.1382	0.1366	0.1366	0.1366	0.1366	0.1366
S-17	3	3	0.4914	0.3143	0.3143	0.3143	0.3143	0.3143
S-18	3	4	0.3232	0.1405	0.1405	0.1405	0.1405	0.1461
S-19	3	4	0.7021	0.7021	0.7021	0.7021	0.7021	0.7021
S-20	3	5	0.7008	0.3506	0.3506	0.3506	0.3506	0.3506
S-21	3	6	0.7015	0.4084	0.4386	0.4724	0.4724	0.4475
S-22	4	2	0.3973	0.3079	0.3079	0.3079	0.3079	0.3079
S-23	4	3	0.2113	0.1313	0.1313	0.1330	0.1330	0.1313
S-24	4	3	0.2357	0.1485	0.1485	0.1485	0.1485	0.1487
S-25	4	4	0.3484	0.2715	0.2258	0.2715	0.2715	0.2789
S-26	4	5	0.3351	0.3383	0.3383	0.3383	0.3383	0.3383
S-27	4	6	0.7003	0.5019	0.5019	0.5019	0.5019	0.5019
S-28	4	7	0.4429	0.4429	0.4429	0.4429	0.4429	0.4429
S-29	5	3	0.3438	0.1817	0.2120	0.1935	0.1935	0.1937
S-30	5	4	0.2523	0.2521	0.2521	0.2543	0.2543	0.2599
S-31	5	4	0.2420	0.1880	0.1880	0.1855	0.1855	0.1855
S-32	5	5	0.3503	0.2319	0.2319	0.2319	0.2319	0.2319
S-33	5	6	0.4421	0.3483	0.3436	0.4421	0.4421	0.3483
S-34	5	7	0.5283	0.2547	0.2462	0.2558	0.2558	0.2602
S-35	5	8	0.7006	0.3746	0.3746	0.3746	0.3746	0.5798
S-36	6	4	0.2543	0.0999	0.1426	0.1947	0.1947	0.1010
S-37	6	5	0.3409	0.2823	0.2823	0.2872	0.2872	0.2872
S-38	6	6	0.3109	0.2634	0.2566	0.2620	0.2620	0.2620
S-39	6	7	0.3852	0.3852	0.3852	0.3852	0.3852	0.3852
S-40	6	7	0.4020	0.2853	0.2853	0.2853	0.2853	0.4028
S-41	6	8	0.7303	0.7303	0.7303	0.7303	0.7303	0.7303
S-42	6	9	0.3822	0.3228	0.3228	0.3228	0.3228	0.3228
S-43	7	5	0.3125	0.1579	0.1579	0.1579	0.1579	0.1579
S-44	7	6	0.2981	0.3065	0.2981	0.3358	0.3358	0.2981
S-45	7	7	0.4659	0.3168	0.3168	0.3168	0.3168	0.3760
S-46	7	8	0.3161	0.2190	0.2190	0.2190	0.2190	0.3161
S-47	7	9	0.7153	0.3668	0.3529	0.3668	0.3668	0.3668
S-48	7	10	0.5422	0.5422	0.5422	0.5422	0.5422	0.5422
S-49	8	6	0.4422	0.3156	0.3750	0.3750	0.3750	0.4422
S-50	8	8	0.2482	0.2285	0.2274	0.2482	0.2482	0.3114
Average			**0.4187**	**0.3172**	**0.3179**	**0.3239**	**0.3243**	**0.3314**
Percent Deviation % From the Best Solution (Average)			**43.19**	**1.13**	**1.96**	**4.66**	**4.81**	**6.29**
Number of Best Solutions			**15**	**38**	**39**	**34**	**33**	**29**

Table F.3: MxN NADP Scenario Comparison Results (OBJ-3)

	Number of Ships	Number of Threats	OBJ-3 Average Value Weighted Hit Prob.					
			Greedy Algorithm	Linearized Model	Decomposition Approach	$HA^{GA+EN+DP}$	HA^{GA+DP}	$HA^{GA+EN+DP}$ (1 sec limit)
S-1	1	1	0.3885	0.3885	0.3885	0.3885	0.3885	0.3885
S-2	1	1	0.2666	0.2666	0.2666	0.2666	0.2666	0.2666
S-3	1	2	0.4197	0.4197	0.4197	0.4197	0.4197	0.4197
S-4	1	2	0.3633	0.3626	0.3657	0.3651	0.3651	0.3651
S-5	1	3	0.3669	0.2390	0.2390	0.2390	0.2390	0.2390
S-6	1	3	0.6452	0.6134	0.6134	0.6134	0.6134	0.6134
S-7	1	4	0.2901	0.1868	0.1766	0.1846	0.1849	0.1833
S-8	2	1	0.1194	0.1194	0.1194	0.1194	0.1194	0.1194
S-9	2	2	0.1810	0.1345	0.1345	0.1345	0.1345	0.1345
S-10	2	2	0.4501	0.4501	0.4501	0.4501	0.4501	0.4501
S-11	2	3	0.1579	0.1014	0.1005	0.0992	0.0992	0.0992
S-12	2	3	0.2589	0.2153	0.2153	0.2153	0.2153	0.2259
S-13	2	4	0.1870	0.0973	0.0972	0.1055	0.1055	0.1058
S-14	2	5	0.4765	0.4512	0.4512	0.4512	0.4512	0.4512
S-15	3	1	0.2438	0.2438	0.2438	0.2438	0.2615	0.2438
S-16	3	2	0.0774	0.0730	0.0730	0.0730	0.0799	0.0730
S-17	3	3	0.2831	0.1935	0.1935	0.1935	0.1935	0.1935
S-18	3	4	0.1941	0.1149	0.1149	0.1149	0.1149	0.1196
S-19	3	4	0.3289	0.3124	0.3124	0.3124	0.3124	0.3124
S-20	3	5	0.3993	0.3171	0.3171	0.3171	0.3171	0.3171
S-21	3	6	0.2707	0.2384	0.2328	0.2357	0.2357	0.2551
S-22	4	2	0.2047	0.1745	0.1745	0.1745	0.1745	0.1745
S-23	4	3	0.1562	0.0970	0.0970	0.0982	0.1037	0.1014
S-24	4	3	0.1308	0.0713	0.0713	0.0713	0.0713	0.0714
S-25	4	4	0.2462	0.1820	0.1787	0.1820	0.1820	0.1864
S-26	4	5	0.2724	0.2049	0.2049	0.2053	0.2053	0.2072
S-27	4	6	0.3723	0.3152	0.3152	0.3244	0.3530	0.3244
S-28	4	7	0.2094	0.1890	0.1938	0.1957	0.1957	0.1966
S-29	5	3	0.2294	0.1265	0.1286	0.1299	0.1299	0.1334
S-30	5	4	0.2175	0.1958	0.1957	0.1973	0.1976	0.2071
S-31	5	4	0.1763	0.1581	0.1581	0.1575	0.1578	0.1653
S-32	5	5	0.1712	0.1261	0.1261	0.1261	0.1261	0.1265
S-33	5	6	0.1454	0.1234	0.1233	0.1286	0.1294	0.1552
S-34	5	7	0.2633	0.1659	0.1672	0.1757	0.1757	0.1865
S-35	5	8	0.3529	0.2845	0.2824	0.2891	0.2891	0.3032
S-36	6	4	0.0982	0.0524	0.0495	0.0531	0.0531	0.0580
S-37	6	5	0.2019	0.1330	0.1330	0.1357	0.1357	0.1366
S-38	6	6	0.1458	0.1301	0.1254	0.1584	0.1586	0.1584
S-39	6	7	0.2634	0.1971	0.1971	0.2021	0.2021	0.2191
S-40	6	7	0.2182	0.1916	0.1898	0.1924	0.1929	0.2103
S-41	6	8	0.1845	0.1439	0.1428	0.1483	0.1484	0.1662
S-42	6	9	0.1956	0.1675	0.1703	0.1676	0.1676	0.1876
S-43	7	5	0.1201	0.1129	0.1052	0.1067	0.1067	0.1090
S-44	7	6	0.2436	0.2193	0.2100	0.2358	0.2358	0.2256
S-45	7	7	0.2643	0.1907	0.1907	0.1907	0.1907	0.2046
S-46	7	8	0.1603	0.1120	0.1115	0.1270	0.1312	0.1611
S-47	7	9	0.2972	0.2390	0.2314	0.2514	0.2518	0.2514
S-48	7	10	0.3358	0.2975	0.2975	0.3163	0.3164	0.3468
S-49	8	6	0.2211	0.1775	0.1761	0.2049	0.2049	0.2307
S-50	8	8	0.1327	0.1046	0.1020	0.1224	0.1224	0.1416
Average			**0.2520**	**0.2084**	**0.2075**	**0.2122**	**0.2135**	**0.2184**
Percent Deviation % From the Best Solution (Average)			**29.55**	**0.88**	**0.18**	**3.26**	**4.01**	**7.31**
Number of Best Solutions			**6**	**31**	**41**	**22**	**19**	**16**

Table F.4: MxN NADP Scenario Comparison Results (OBJ-4)

	Number of Ships	Number of Threats	OBJ-4 Value Weighted Ship Surv. Prob.					
			Greedy Algorithm	Linearized Model	Decomposition Approach	$HA^{GA+EN+DP}$	HA^{GA+DP}	$HA^{GA+EN+DP}$ (1 sec limit)
S-1	1	1	0.6115	0.6115	0.6115	0.6115	0.6115	0.6115
S-2	1	1	0.7334	0.7334	0.7334	0.7334	0.7334	0.7334
S-3	1	2	0.3365	0.3365	0.3365	0.3365	0.3365	0.3365
S-4	1	2	0.4035	0.4045	0.4019	0.4027	0.4027	0.4027
S-5	1	3	0.1875	0.4370	0.4370	0.4370	0.4370	0.4370
S-6	1	3	0.0410	0.0492	0.0492	0.0492	0.0492	0.0492
S-7	1	4	0.1769	0.4255	0.4538	0.4366	0.4359	0.4396
S-8	2	1	0.9523	0.9523	0.9523	0.9523	0.9523	0.9523
S-9	2	2	0.7150	0.7899	0.7899	0.7899	0.7899	0.7899
S-10	2	2	0.6475	0.6475	0.6475	0.6475	0.6475	0.6475
S-11	2	3	0.7431	0.8303	0.8318	0.8336	0.8336	0.8336
S-12	2	3	0.5231	0.5854	0.5854	0.5854	0.5854	0.5705
S-13	2	4	0.6645	0.8154	0.8156	0.8001	0.8001	0.7995
S-14	2	5	0.3315	0.3418	0.3418	0.3418	0.3418	0.3418
S-15	3	1	0.9512	0.9512	0.9512	0.9512	0.9477	0.9512
S-16	3	2	0.9381	0.9416	0.9416	0.9416	0.9361	0.9416
S-17	3	3	0.7718	0.8318	0.8318	0.8318	0.8318	0.8318
S-18	3	4	0.7896	0.8732	0.8732	0.8732	0.8732	0.8679
S-19	3	4	0.5838	0.5920	0.5920	0.5920	0.5920	0.5920
S-20	3	5	0.5171	0.5845	0.5845	0.5845	0.5845	0.5845
S-21	3	6	0.6979	0.7201	0.7279	0.7309	0.7309	0.7065
S-22	4	2	0.8635	0.8836	0.8836	0.8836	0.8836	0.8836
S-23	4	3	0.9293	0.9541	0.9541	0.9535	0.9507	0.9519
S-24	4	3	0.9390	0.9661	0.9661	0.9661	0.9661	0.9661
S-25	4	4	0.8165	0.8642	0.8669	0.8642	0.8642	0.8609
S-26	4	5	0.6772	0.7439	0.7439	0.7434	0.7434	0.7408
S-27	4	6	0.4956	0.5887	0.5887	0.5807	0.5511	0.5807
S-28	4	7	0.6626	0.6600	0.6581	0.6808	0.6808	0.6799
S-29	5	3	0.8427	0.9041	0.9019	0.9017	0.9017	0.8995
S-30	5	4	0.8271	0.8434	0.8436	0.8425	0.8422	0.8346
S-31	5	4	0.8709	0.8827	0.8827	0.8826	0.8825	0.8783
S-32	5	5	0.8520	0.8912	0.8912	0.8912	0.8912	0.8908
S-33	5	6	0.8036	0.8363	0.8366	0.8244	0.8233	0.7951
S-34	5	7	0.7946	0.8692	0.8690	0.8562	0.8562	0.8435
S-35	5	8	0.4966	0.6031	0.6063	0.5960	0.5960	0.5945
S-36	6	4	0.9387	0.9671	0.9687	0.9661	0.9661	0.9634
S-37	6	5	0.8233	0.8821	0.8821	0.8797	0.8797	0.8789
S-38	6	6	0.9300	0.9363	0.9389	0.9251	0.9251	0.9251
S-39	6	7	0.6525	0.7004	0.7004	0.6954	0.6954	0.6823
S-40	6	7	0.7545	0.7867	0.7886	0.7863	0.7858	0.7612
S-41	6	8	0.8435	0.8631	0.8709	0.8589	0.8589	0.8599
S-42	6	9	0.7549	0.7908	0.7869	0.7905	0.7905	0.7682
S-43	7	5	0.9270	0.9292	0.9347	0.9339	0.9339	0.9323
S-44	7	6	0.8201	0.8362	0.8436	0.8263	0.8263	0.8343
S-45	7	7	0.7369	0.8057	0.8057	0.8057	0.8057	0.7907
S-46	7	8	0.8473	0.8821	0.8807	0.8633	0.8594	0.8497
S-47	7	9	0.6347	0.7044	0.7147	0.6880	0.6877	0.6880
S-48	7	10	0.5444	0.5910	0.5910	0.5718	0.5717	0.5447
S-49	8	6	0.8071	0.8588	0.8597	0.8452	0.8452	0.8021
S-50	8	8	0.8828	0.9041	0.9059	0.8925	0.8925	0.8745
Average			**0.7017**	**0.7477**	**0.7491**	**0.7452**	**0.7442**	**0.7395**
Percent Deviation % From the Best Solution (Average)			**7.63**	**0.34**	**0.13**	**0.64**	**0.78**	**1.36**
Number of Best Solutions			**6**	**31**	**39**	**22**	**20**	**16**

Table F.5: MxN NADP Scenario Comparison Results (Run Times)

	Number of Ships	Number of Threats	Run Time (seconds)					
			Greedy Algorithm	Linearized Model	Decomposition Approach	$HA^{GA+EN+DP}$	HA^{GA+DP}	$HA^{GA+EN+DP}$ (1 sec limit)
S-1	1	1	0.1026	0.0920	0.4300	0.1176	0.5465	0.1503
S-2	1	1	0.0710	0.2250	0.9340	0.0802	0.4294	0.0804
S-3	1	2	0.1597	0.1120	0.8470	0.6290	0.5225	0.8632
S-4	1	2	0.1483	0.3170	1.1570	0.8889	0.8429	1.0728
S-5	1	3	0.1970	2.5800	3.5530	0.9220	0.7682	0.9905
S-6	1	3	0.1770	0.2330	0.8110	0.5799	0.5800	0.6193
S-7	1	4	0.4613	500.4360	13.4540	1.9932	1.9278	1.5240
S-8	2	1	0.2162	0.1190	0.7930	0.2035	0.6948	0.2345
S-9	2	2	0.8341	2.1440	2.1030	4.5271	2.6736	2.5838
S-10	2	2	0.2982	0.2140	1.0410	0.8048	0.8338	0.8674
S-11	2	3	0.2303	4.1970	2.5120	1.5997	1.6211	1.2465
S-12	2	3	0.2821	15.2240	13.2290	2.4178	2.3580	1.2853
S-13	2	4	0.4212	500.4440	31.4410	2.7327	2.9774	1.5026
S-14	2	5	0.3640	57.9770	8.2920	1.8808	1.7950	1.3710
S-15	3	1	0.2201	0.1220	1.0160	0.2121	0.6001	0.2244
S-16	3	2	0.3094	0.2810	2.1800	1.0025	1.0195	1.0724
S-17	3	3	0.4871	2.9600	5.6850	1.8425	1.8110	1.5386
S-18	3	4	0.8255	2.6470	5.1000	6.5027	6.5912	1.9705
S-19	3	4	0.7669	0.8120	2.1040	1.5954	1.6325	1.5814
S-20	3	5	0.9068	1.1770	3.3490	1.9662	1.9916	1.9530
S-21	3	6	0.8339	500.7490	507.4240	5.9150	5.9578	1.9490
S-22	4	2	0.9016	0.5480	1.6230	1.7853	1.7837	1.8822
S-23	4	3	1.2376	5.9300	7.6900	8.0239	8.0191	2.3549
S-24	4	3	1.2855	7.5080	3.7710	6.8564	6.9398	2.3348
S-25	4	4	0.7376	0.8150	2.1240	4.5605	4.5696	1.8693
S-26	4	5	0.5575	10.8040	4.5240	4.0723	3.9923	1.6178
S-27	4	6	0.9228	3.6630	5.5610	2.7329	2.7626	2.0443
S-28	4	7	1.9260	503.0420	29.3360	28.3305	24.8214	4.5568
S-29	5	3	0.6708	500.4510	7.6410	4.6267	4.6075	1.7778
S-30	5	4	0.5969	500.5020	6.7060	5.1657	5.2693	1.6400
S-31	5	4	0.6548	4.5970	2.8330	4.2057	4.3245	1.7210
S-32	5	5	1.6196	1.4280	14.2910	5.7656	6.1262	2.7340
S-33	5	6	1.0798	501.2940	12.6390	10.2289	10.5603	2.5096
S-34	5	7	1.3451	579.8650	276.7050	13.0861	13.3507	3.1298
S-35	5	8	1.1507	502.0490	64.7680	9.3588	9.5697	2.3838
S-36	6	4	1.7107	9.6360	16.0680	10.5886	10.6135	2.9638
S-37	6	5	1.0580	3.1160	6.9800	7.4202	7.3403	2.1740
S-38	6	6	3.5930	505.5630	60.5300	11.2621	10.9628	4.8943
S-39	6	7	2.4810	15.0820	15.0910	6.3008	6.2595	2.3441
S-40	6	7	1.3336	48.9160	23.9210	12.8163	12.9801	3.0223
S-41	6	8	1.7622	435.5990	89.9990	14.2960	14.1165	3.7028
S-42	6	9	1.5332	56.1670	78.1260	13.1627	13.2923	3.3551
S-43	7	5	1.8052	501.4360	21.4190	12.4705	12.6100	3.6540
S-44	7	6	2.3142	502.9380	44.5470	18.7697	18.4843	5.0155
S-45	7	7	1.8950	5.1030	18.2580	11.9162	11.8906	3.7043
S-46	7	8	2.6052	506.6130	1451.6690	36.5189	35.5453	7.6803
S-47	7	9	1.4899	502.4740	37.8250	6.3870	6.4607	3.4859
S-48	7	10	2.2000	385.2400	31.9460	17.2931	17.3335	4.7636
S-49	8	6	1.2197	9.1720	6.7580	6.4035	7.8416	2.4941
S-50	8	8	3.5174	507.8930	141.8570	36.8625	36.1582	5.8150
Average			**1.0703**	**174.2101**	**61.8532**	**7.3936**	**7.3352**	**2.3261**

CURRICULUM VITAE

Surname, Name : Arslan, Caner

EDUCATION

M.S. in Industrial and Operations Research, University of Michigan, Ann Arbor, MI, USA, 2015

B.S. in Operations Analysis, Naval Academy, İstanbul, Turkey, 2010

WORK EXPERIENCE

Operations Research Analist , Operations Research Division, Turkish Navy HQs, Ankara, Turkiye, August 2015 – Present.

Electronics Officer, TCG KALKAN (PGFG), Kocaeli, Turkiye, August 2011 – August 2013.

Weapon Electronics Officer, TCG GÖKOVA (FFGH), Kocaeli, Turkiye, September 2010 – August 2011.

FOREIGN LANGUAGES

English

RESEARCH INTERESTS

Mathematical modelling, optimization and its applications,
Decision Support Systems,
Military Operations Research,
Heuristic search.

HOBBIES

Tennis, Scuba Diving, Computer Technologies, Books

PENDING PURSUITS

Engage in agriculture,
Learning the art of fishing,
Learning to play a musical instrument (initial attempt will focus on piano),
Lifting an amateur-level tennis trophy...