WTA problem

Authors: Cyprien Fourcroy, Paulina Kulczyk, Jan Pogłód

with support of Professor Radosław Pytlak, DSc, Eng. and Robert Przybycień, PhD student

Warsaw University of Technology
Department of Mathematics and Information Science

ABSTRACT

In this study, we examine weapons and target allocation (WTA) models specifically tailored to maritime task force air defense scenarios, with an emphasis on realistic anti-missile defense (ASM) conditions. By adapting Karasakal's well-known multi-stage algorithm to a single-stage configuration, we facilitate a direct comparison with the exact algorithm proposed by Chen & Lu. Through computational experiments and comparative analysis, we evaluate the time efficiency and defensive effectiveness of both algorithms by considering different scenarios with diverse datasets. In addition, we discuss the potential applicability and extension of WTA models to other strategic sectors, suggesting broader impacts beyond naval air defense.

Keywords: *WTA, optimization problem, military algorithms*

1 Introduction

The Weapon-Target Allocation (WTA) problem, central to military operations research since its formulation by Manne in 1958 [8], addresses the optimal assignment of defensive resources against incoming threats to minimize potential damage. The rising threat from advanced **anti-ship missiles (ASMs)** underscores the importance of effective WTA models, particularly within naval **task group (TG)** air defense scenarios.

WTA algorithms, ranging from heuristic to exact methods, significantly impact defense capabilities by optimizing resource allocation under diverse operational constraints such as ammunition limitations and engagement tactics. In this study, we comparatively examine two recognized approaches: a well-cited multi-stage allocation algorithm proposed by Karasakal (2008) [4] and a precise, exact solution method developed by Yiping Lu and Danny Z. Chen [7]. To facilitate meaningful comparisons, we adapt Karasakal's model into a single-stage configuration, aligning it structurally with Chen & Lu's methodology.

Our modifications simplify the original multi-stage assumptions, enabling direct benchmarking of algorithmic effectiveness, computational speed, and operational viability across varied scenarios. Detailed explanations of these adjustments and their implications will be addressed in the methodology section (Sec.: 2).

Finally, we discuss the potential applicability and extension of WTA models into not only broader defense and strategic operational contexts, opening avenues for future research and application, but also in other sectors domain (see Sec.: 4).

1.1 Literature Revieve

The Weapon-Target Allocation (WTA) problem has been extensively explored since the seminal work of Manne (1958), who introduced foundational static allocation models aimed at optimally assigning defensive resources. Subsequently, the literature has expanded significantly, encompassing diverse algorithmic strategies tailored to increasingly complex operational scenarios.

Early WTA models primarily utilized static allocation techniques focusing on predetermined threat environments. Progressing from these foundational approaches, researchers developed dynamic and multi-stage variants to better represent real-world complexities, as exemplified by recent contributions such as those from Kline, Ahner, and Hill (2018) [5]. These dynamic models offer improved accuracy by reflecting evolving threats and resource constraints over multiple stages of engagement.

Algorithmic solutions to the WTA problem broadly encompass heuristic approaches—such as genetic algorithms and ant colony optimization—that prioritize computational efficiency and adaptability, alongside exact computational methods like branch-and-bound and integer programming, which emphasize solution optimality. Karasakal's (2008) algorithm exemplifies a practical multi-stage approach tailored specifically to naval task group scenarios, incorporating assumptions reflective of real operational conditions, including ammunition constraints and coordinated engagement policies.

In contrast, Chen & Lu's exact algorithm prioritizes computational rigor and solution precision, making it particularly suitable for benchmark evaluations. Given the structural differences between multi-stage and exact approaches, our study strategically adapts Karasakal's method to a single-stage format to facilitate clear and direct comparisons.

Recent advances further enrich the WTA literature, incorporating sensor-integrated systems, multi-objective optimization, and game-theoretic frameworks. These methodologies present promising avenues for extending and generalizing our findings to diverse defense contexts and beyond, highlighting ongoing opportunities for innovation and practical application.

2 Methodological Framework

In this section, we present the methodological framework adopted to conduct a comparative analysis of the selected weapon-target allocation (WTA) algorithms. We first describe the algorithmic approaches, detailing the modifications we applied to ensure a fair and meaningful benchmark. Subsequently, we outline the dataset and scenarios used, emphasizing how they reflect realistic operational conditions pertinent to naval task group air defense scenarios.

2.1 Algorithmic Approaches and Modifications

In our study, we comparatively analyze two weapon-target allocation (WTA) algorithms: one proposed by Karasakal in 2008 and another by Lu & Chen in 2019. A crucial difference between these two lies in their structural assumptions. Karasakal's algorithm is designed as a multi-stage model based on a Shoot-Look-Shoot (SLS) tactic. In this approach, engagements are executed in successive phases: missiles are launched, the outcomes are observed, and additional responses are then decided. By contrast, the model proposed by Lu and Chen assumes a single-stage setting in which all missiles are allocated simultaneously in one decision cycle, without reconsideration of shooting again (this approach can be seen in Figure 1).

This fundamental divergence prevents direct algorithmic comparison. To facilitate a meaningful benchmark, we adapt Karasakal's multi-stage model into a one-stage version. This modification places both algorithms on equal footing regarding their engagement structure. Minor differences in input parameters were also reconciled, and details of these technical adjustments are presented later in this section.

Before we present the mathematical core of the Lu & Chen algorithm, we briefly introduce **the basic notation used throughout our models:**

- i: index of incoming anti-ship missiles (ASMs),
- j: : index of surface-to-air missile (SAM) systems onboard the naval task group (TG),
- a_i: : importance or threat level of ASM , representing its danger to the task group,
- $p_{i,j}$: single-shot kill probability of SAM system against ASM , where $0 < p_{i,j} < 1$,
- $x_{i,j}$: if SAM system j to be launched against ASM i ($x_{i,j} = 1$, if launched; $x_{i,j} = 0$ otherwise.)

We begin with an overview of the Lu & Chen algorithm, which remains unchanged in our comparative analysis.

Lu & Chen (2019) Lu & Chen model the WTA problem using a binary integer programming formulation enhanced with a column generation strategy. Their core objective is to minimize the total expected survival value of all targets after weapon engagements. The mathematical model consists of a master problem and subproblem structure, where candidate engagement patterns (or "columns") are iteratively generated and added to improve the current solution.

The simplified version of their model can be described as follows:

- **Each pattern represents a unique way to assign available weapons to a particular target.**

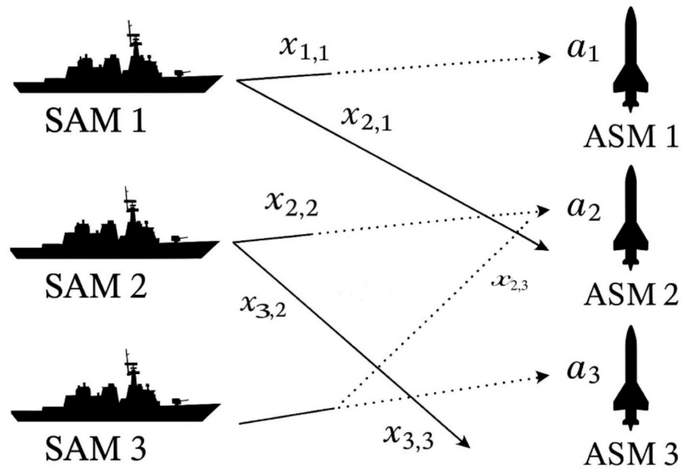

Fig. 1: Diagram of the WTA model. Symbols represent: SAM - surface-to-air missile, ASM - anti-ship missile a_i - danger to the SAM, $x_{i,j}$ - if SAM system j is launched against ASM i.

- **The value of a pattern is determined by computing the expected survival of the target if that engagement is executed.**
- **The master problem selects an optimal subset of patterns such that each target is covered exactly once, and weapon usage constraints are satisfied.**

In practice, the algorithm starts with a restricted set of patterns and solves the master problem using CPLEX. Dual variables from the solution guide the column generation procedure, identifying new patterns (columns) with negative reduced costs. These are added iteratively until no such pattern remains, at which point the model is solved as a full integer program.

In the following paragraph, we describe Karasakal's model and the adaptations made to it to align it structurally with this single-stage formulation.

Karasakal (2008) Karasakal originally proposed a nonlinear integer programming model (Problem P) to allocate SAM systems to incoming ASMs in a naval task group. The objective was to maximize the overall probability of neutralizing threats. The problem (P) was structured as follows:

$$\textbf{[P]: } 1 - \prod_{j \in M | (i,j) \in v(i,j)} (1 - p_{i,j})^{x_{i,j}} \geq h_i \forall i \in M \quad (1)$$

$$\text{S.t.:} \sum_{\{i \in N | (i,j) \in v(i,j)\}} x_{ij} \leq d_j, \quad \forall j \in M \quad (2)$$

$$\sum_{\{j \in M | (i,j) \in v(i,j)\}} x_{ij} \leq s_i, \quad \forall i \in N \quad (3)$$

$$0 \leq x_{ij} \leq u_{ij}, \quad \forall i, j \in v(i,j) \quad (4)$$

where $v(i,j)$: the ability of the SAM j system to attack the ASM i.

To make the model tractable, the nonlinear structure was approximated using logarithmic transformations (problem P'). Also to guarantee reaching a feasible solution and and to map its reality, two linear integer programming variants were proposed:

Model P1 – Feasibility via Fake Weapons
Model P1 introduces a set of artificial weapons (with high kill probabilities) to guarantee feasibility. The objective is to **minimize their usage**, encouraging solutions that rely on real weapons.

$$\textbf{[P']: } \sum_{i=1}^{M} x_{ij} \cdot \log(1 - p_{ij}) \leq \log(1 - h_j) \quad \forall j \in N \quad (5)$$

$$\min \sum_{i=M_{\text{real}}+1}^{M} \sum_{j=1}^{N} x_{ij} \quad (6)$$

(2)

S. t.: (2), (3) and (4).

Model P2 – Constraint Relaxation with Slack Variable
Model P2 avoids fake weapons by introducing a slack variable $e \geq 0$ that softly relaxes the survival constraint. The model minimizes the violation:

$$\sum_{i=1}^{M} x_{ij} \cdot \log(1 - p_{ij}) \geq \log(1 - h_j) - e \quad \forall j = 1, \ldots, N \tag{5}$$

$$\sum_{j=1}^{N} x_{ij} \leq 1 \quad \forall i = 1, \ldots, M \tag{6}$$

$$\min \quad e \tag{7}$$

(3)

S. t.: (2), (3) and (4).

Our Modifications to Karasakal's Model To align Karasakal's multi-stage model with Lu & Chen's single-stage framework, we introduced the following simplifications:

- Removed constraints related to ammunition limits d_j and engagement scheduling s_i.
- Recast $x_{i,j}$ as binary variables: $x_{i,j} \in \{0, 1\}$, indicating whether j SAM is assigned to ASM i.
- Substituted the required kill probability h_i with a function of ASM importance: $h(i) = \frac{a_i}{max(a)}$.
- Replaced (as author proposed) product terms with log-sum expressions to preserve convexity while enabling linear formulation.

These changes result in a simplified one-stage linear model suitable for benchmarking against Lu & Chen's algorithm while preserving core features of target prioritization and allocation feasibility.

2.2 Dataset

Each problem instance is defined by the following parameters:

- M: the number of weapons;
- N: the number of targets;
- $\mathbf{P} = [p_{ij}]$: a matrix of kill probabilities, where p_{ij} is the probability that weapon i successfully destroys target j;
- $\mathbf{a} = [a_j]$: the value or importance of each target j.

Solving the Weapon-Target Assignment (WTA) problem at full scale can result in an excessive number of binary decision variables. In our models, the number of such variables is approximately $2MN$ (due to binary variables and auxiliary variables introduced for modeling constraints). To ensure that all models remain solvable in reasonable time with CPLEX, we imposed a constraint on the instance size: we only process instances where $2MN \leq 1000$.

Shrinking Strategy. When an instance exceeds this threshold, we reduce its size while preserving key statistical properties. Let $\bar{a}$ and $\bar{p}$ denote the mean target value and mean kill probability, respectively. We also define the ratio $\rho = N/M$.

To preserve the general structure of the original instance, we:

- select a subset of $M_{\text{new}} < M$ weapons whose average probabilities are closest to the global mean $\bar{p}$;
- select a subset of $N_{\text{new}} < N$ targets whose values are closest to the mean $\bar{a}$;
- ensure that the target-to-weapon ratio ρ is approximately preserved: $\frac{N_{\text{new}}}{M_{\text{new}}} \approx \frac{N}{M}$;
- ensure that $\bar{a}_{\text{new}} \approx \bar{a}$ and $\bar{p}_{\text{new}} \approx \bar{p}$.

This approach maintains the statistical characteristics of the original problem, thereby allowing us to compare models' performances on scaled-down, yet representative, instances.

3 Experiments and results

To test the performance and behavior of the algorithms proposed by Karasakal and Lu & Chen, we designed two types of experiments based on 24 benchmark instances. For the P1 problem we created fake weopons as shown in the both Figures 2 3. We created exactly as many fake weapons as real ones

- **Experiment 1** assumes a high number of targets with a limited number of available real (and fake) weapons. This setting tests the models' ability to allocate scarce resources efficiently under pressure. The corresponding distribution is shown in Figure 2.
- **Experiment 2** assumes a reversed situation: a high number of available weapons relative to a smaller number of targets. This configuration assesses how models behave when defense resources are abundant. See Figure 3 for the distribution.

For model P1, each instance was extended with the same number of fake weapons as real ones to ensure feasibility in both experiments.

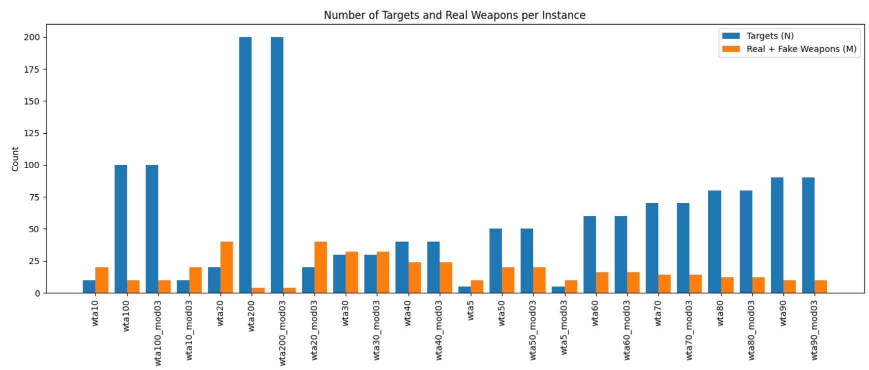

Fig. 2: Number of targets and total weapons (real + fake) per instance (experiment 1)

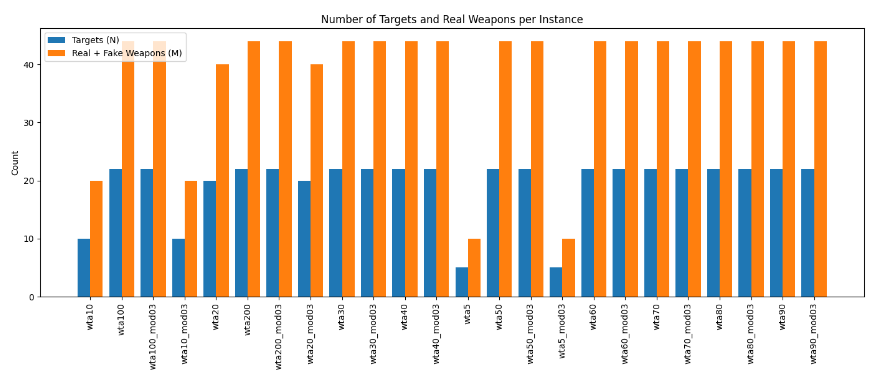

Fig. 3: Number of targets and total weapons (real + fake) per instance (experiment 2)

Expected Survival Analysis

To assess the effectiveness of the P1 model in both scenarios, we computed the *expected survival* of the targets using only the real weapons (fake ones were excluded from this metric). The expected survival for each instance is calculated as:

$$\text{ExpectedSurvival} = \sum_{j=1}^{N} a_j \cdot \prod_{i=1}^{M_{\text{real}}} (1 - p_{ij})^{x_{ij}} \tag{4}$$

where a_j is the value of target j, p_{ij} is the kill probability of weapon i against target j, and $x_{ij} \in \{0, 1\}$ indicates whether weapon i is assigned to target j.

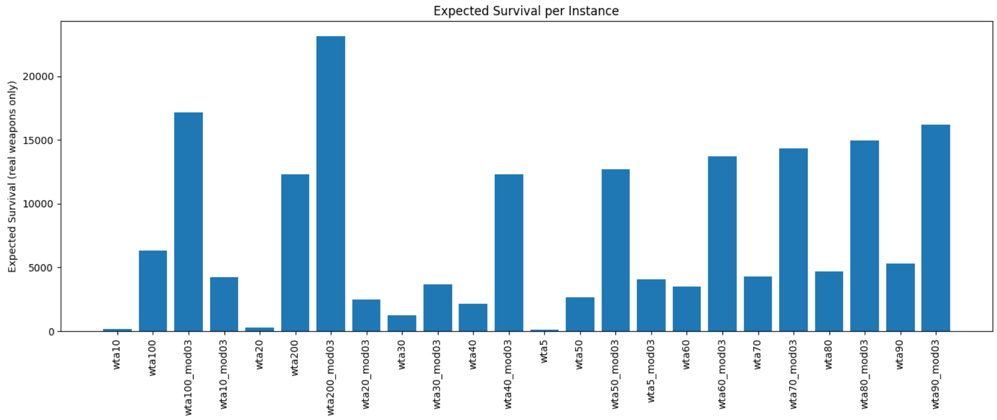

Fig. 4: Expected survival using real weapons only (experiment 1)

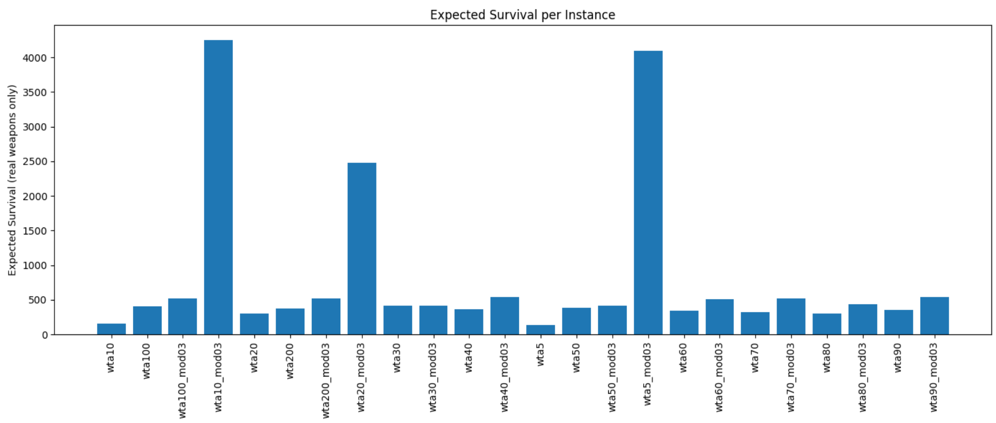

Fig. 5: Expected survival using real weapons only (experiment 2)

Figures 4 and 5 present the computed expected survival for all 24 instances in both experimental settings. We observe a notable contrast between the two experiments:

- **Experiment 1** involves a larger number of targets compared to available weapons. As a result, the total value of the protected targets is much higher, and the expected survival values reach **up to 20,000**.
- **Experiment 2**, on the other hand, includes fewer targets and more available weapons. Because of more firepower is available per target, the total expected survival is lower (**below 4,000**) due to the lower total value of all targets in these instances.

These findings confirm that the absolute level of expected survival strongly depends on the total target value in the instance. While better weapon-to-target ratios improve survivability per target, the overall survival value is lower when there are too much missieles on targets.

Comparison of P1 and P2: Expected Survival

To evaluate the practical impact of the two modeling approaches, we compared the *expected survival* values obtained using weapons under models P1 and P2.

As shown in Figure 6, model P2 consistently produces equal or higher expected survival values across nearly all instances.

In many cases, P2 significantly outperforms P1 in terms of survival, especially when weapon resources are limited. This is expected, since P2 relaxes the survival constraint slightly using a slack variable e, which allows it to achieve better overall allocation and reduce overcommitment to difficult targets. P1, by contrast, strictly enforces the hard constraints using fake weapons if needed, which may lead to suboptimal use of real weapons in some scenarios.

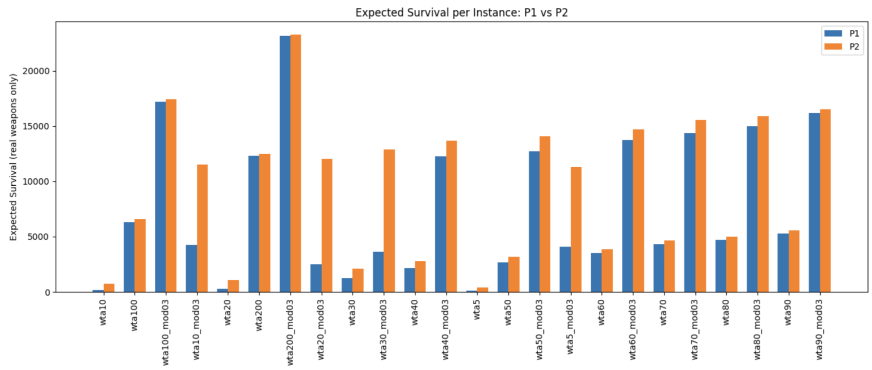

Fig. 6: Expected survival value P1 vs P2

Model P2 provides a more flexible and often more effective strategy in terms of maximizing target survival using real weapons.

Per-Target Attack Effectiveness Analysis

To evaluate how effectively the model allocates firepower across targets, we analyze the *log-sum attack effectiveness* for each target in a given instance. This visualization allows us to interpret how much destructive pressure is applied to each target relative to its required survivability.

For each target $j \in \{1, \ldots, N\}$, we compute:

- **LHS (Log-Sum Effectiveness)** – cumulative destructive impact from assigned weapons:

$$\mathrm{LHS}_j = \sum_{i=1}^{M} x_{ij} \cdot \log(1 - p_{ij})$$

  This value represents the logarithmic survival probability resulting from the weapons assigned to target j. A more negative LHS indicates stronger or more numerous attacks.

- **RHS (Survival Threshold)** – minimum required survival level for the target:

$$\mathrm{RHS}_j = \log(1 - h_j)$$

  Here, $h_j \in (0, 1)$ reflects the required minimum survival probability of the target. Targets with higher value a_j are typically assigned higher h_j, hence stricter thresholds.

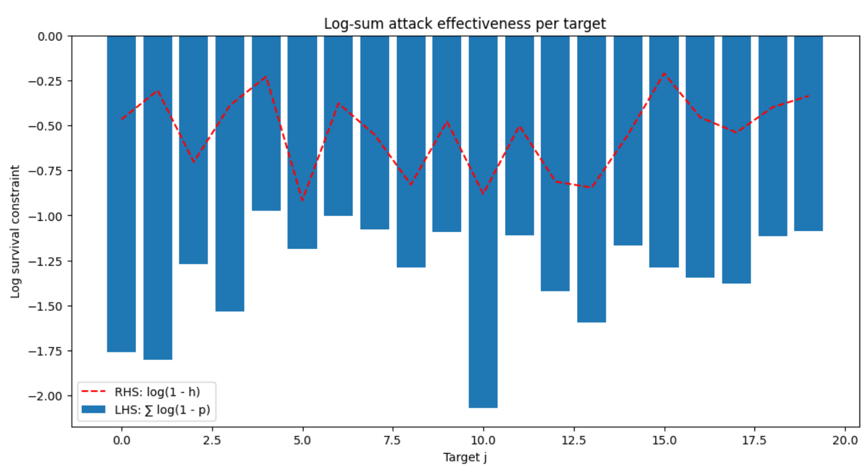

Fig. 7: Expected survival using real weapons only (experiment 1)

Figure 7 illustrates the attack effectiveness (LHS) for each target as a blue bar, while the red dashed line represents the survival threshold (RHS) the model must not exceed.

This analysis highlights:

- Targets where the LHS is significantly below the RHS are subject to strong suppression — more destructive power was assigned than strictly necessary.
- Targets with LHS values close to RHS represent minimal but sufficient allocation. The model achieves efficient coverage by applying just enough force.
- Uniformity or disparity across targets reveals how attack resources are distributed. Sharp dips in LHS imply prioritization or tactical concentration.

This visualization helps us understand how the model balances survivability constraints and resource efficiency, and how different targets are prioritized in terms of attack pressure.

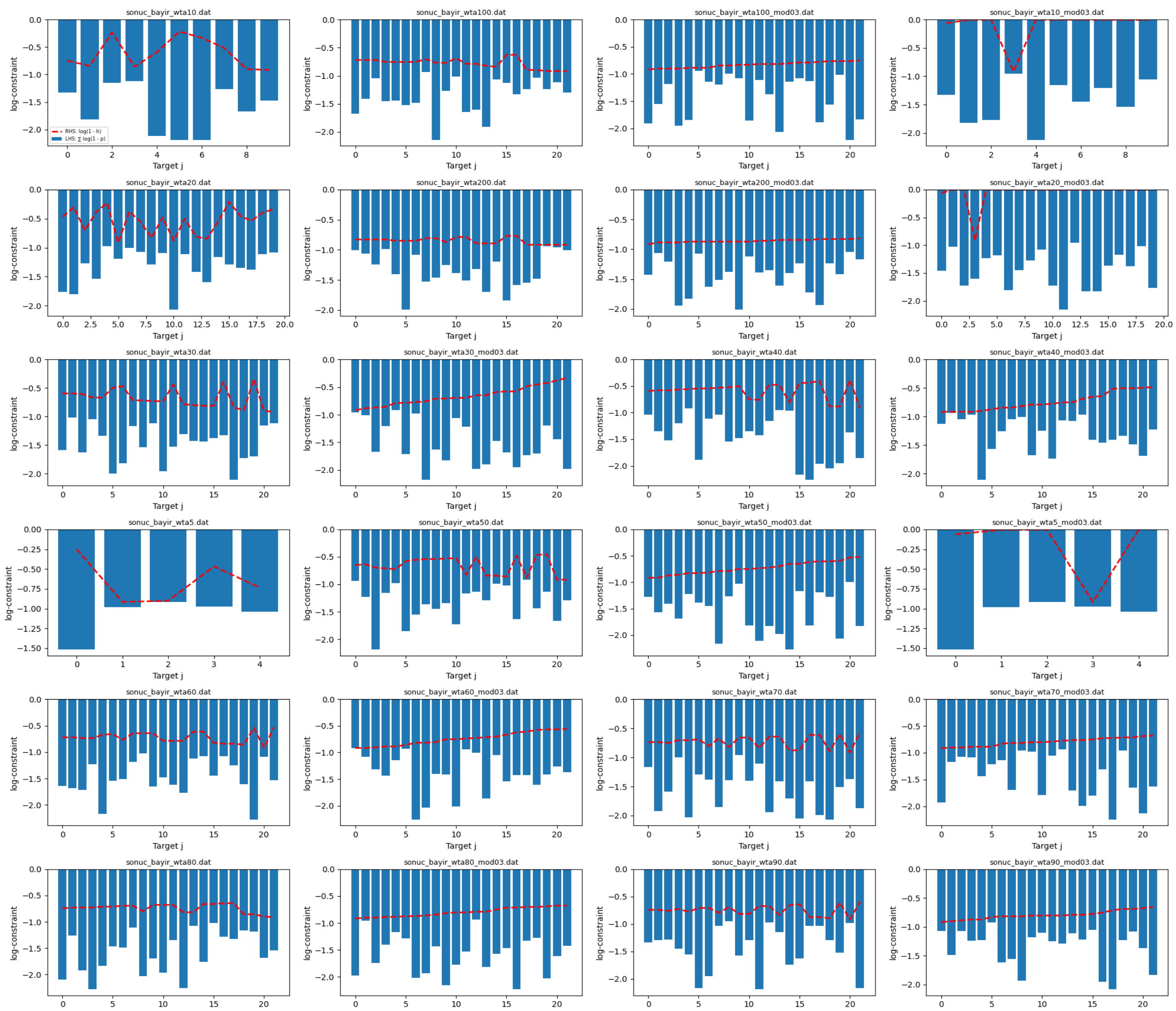

Fig. 8: Attack effectiveness on all instnces (experiment 2)

3.1 Comparison of Karasakal's algorithm with Chen & Lu's implementation

Using Expected Survival as a metric, we compare the P1 and P2 formulations of Karasakal's model. It is expected that these models will perform worse than the Chen & Lu implementation (P3), as the latter directly optimizes this metric as its objective function, whereas in P1 and P2, it must be computed a posteriori.

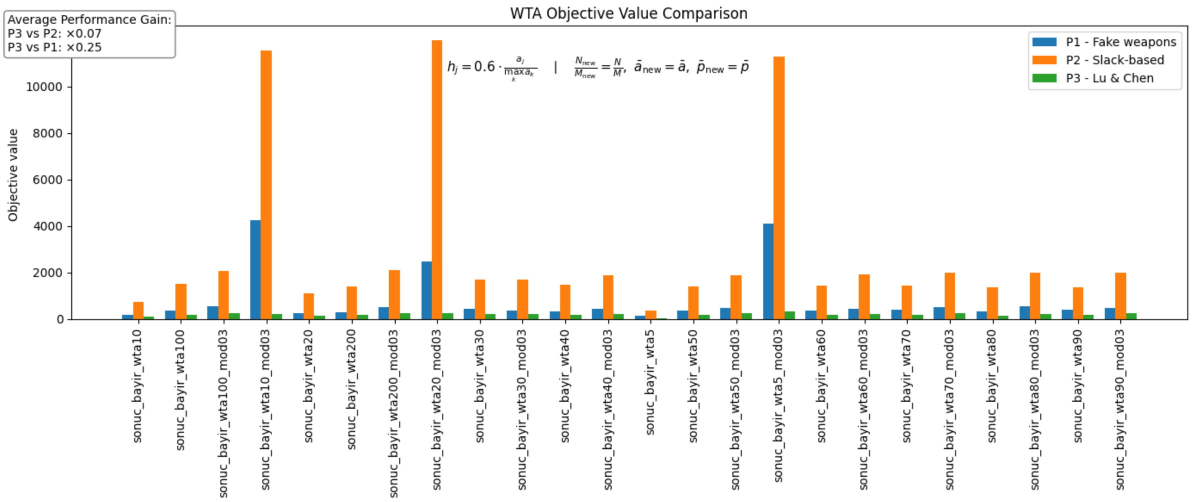

Fig. 9: Expected Survival across models P1, P2, and Chen & Lu (P3) — Experiment 1.

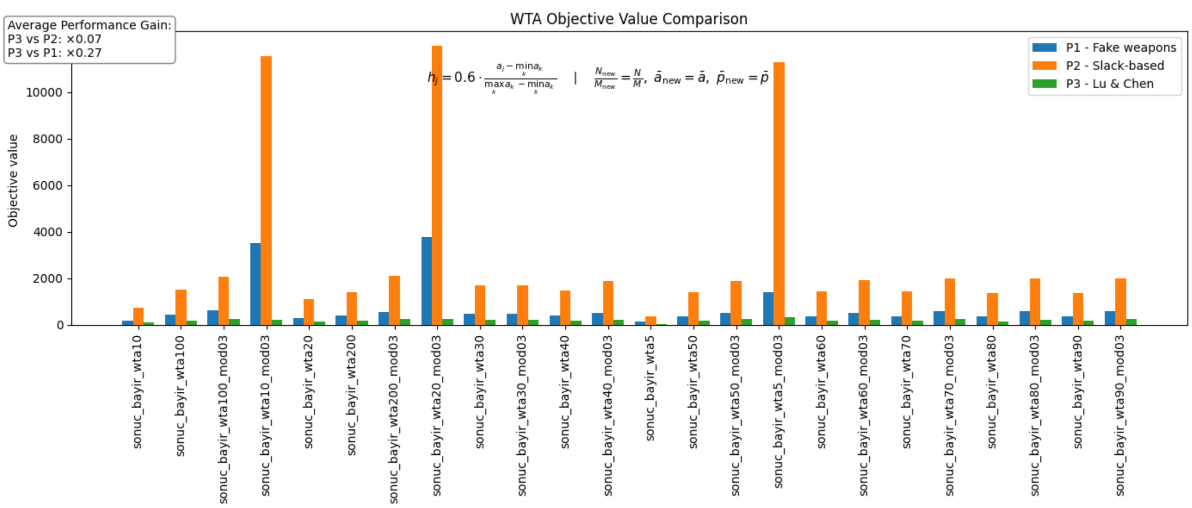

Fig. 10: Expected Survival across models P1, P2, and Chen & Lu (P3) — Experiment 2.

As observed in Figures 9 and 10, the expected survival values follow a consistent hierarchy across both experiments:

$$\text{ExpectedSurvival(P3)} < \text{ExpectedSurvival(P1)} < \text{ExpectedSurvival(P2)}$$

This ranking is coherent with the nature of each formulation: P3 (Chen & Lu) aims to **minimize survival** as its direct objective function, while P1 and P2 are alternative formulations where survival is only computed afterward. Since our objective is to minimize the expected survival of enemy targets, the solution proposed by P1 is more suitable in an engagement context. However, as we will see, this strategy requires longer computation time, which is sometimes not feasible in real-time combat situations

3.2 Time comparison results

To evaluate the computational efficiency of each model, we compare solving times across the three implementations. As shown in Figure 11, Model P1 is on average **10.13 times faster** than the exact method proposed by Chen & Lu (P3), and Model P2 is approximately **4.30 times faster** than P1.

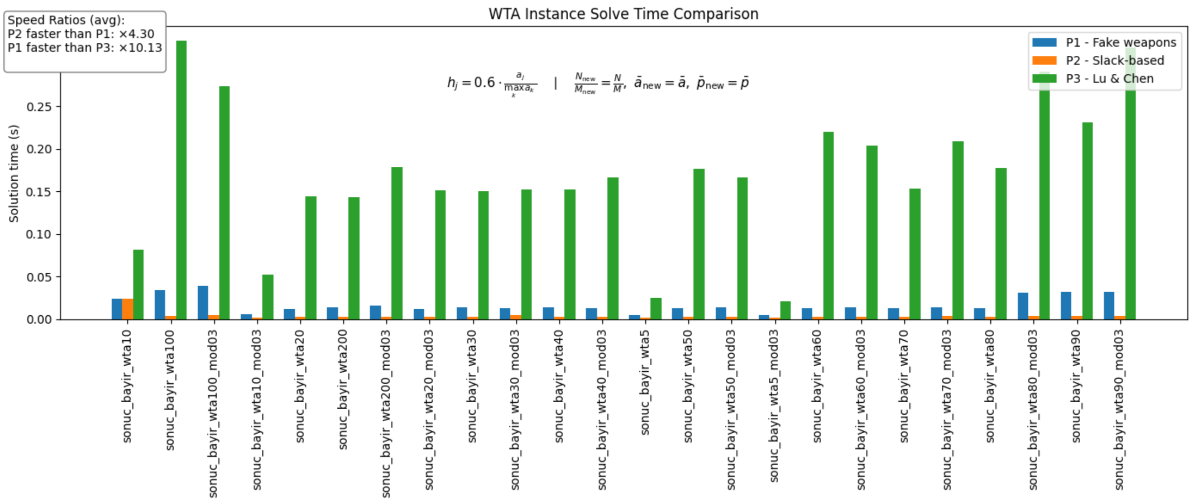

Fig. 11: Solving times for P1, P2, and Chen & Lu (P3) — base configuration.

These results confirm that while P3 achieves high precision, it incurs significant computational costs. On the other hand, P1 and P2 provide valuable trade-offs between accuracy and efficiency, making them more suitable for real-time or resource-constrained scenarios.

We then assess whether the specific functional form used to account for target importance significantly affects computation time. The results in Figure 12 show that the choice of target importance function has negligible impact on solving time.

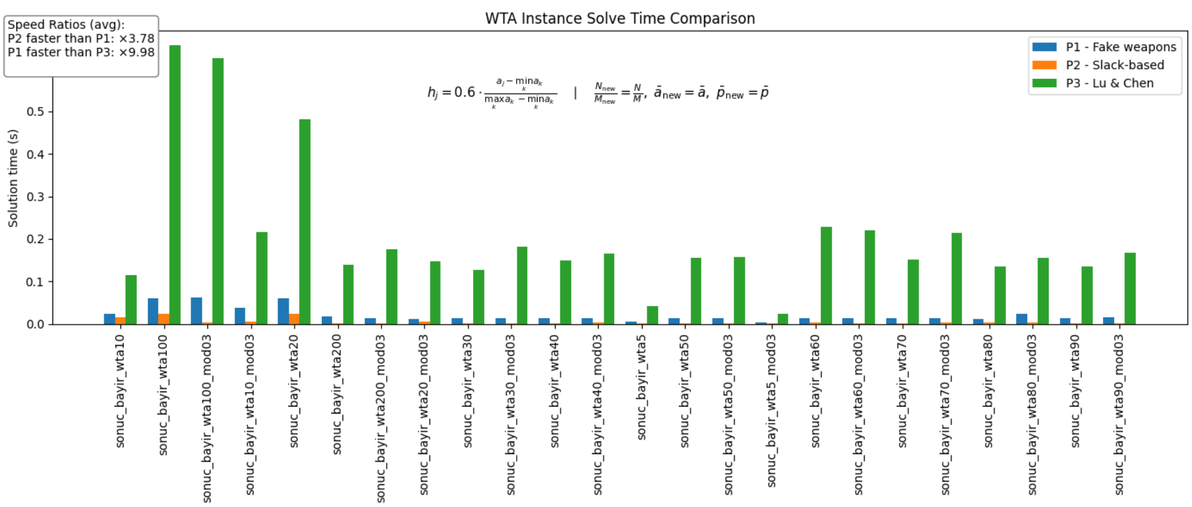

Fig. 12: Solving times with alternative target-value scaling functions.

In summary, we consistently observe the following hierarchy in solving time:

$$\text{runtime(P2)} < \text{runtime(P1)} < \text{runtime(P3: Chen \& Lu)}$$

4 Applicability and extensions

While Weapon-Target Allocation (WTA) models are deeply rooted in naval air defense, their structure and solution techniques offer significant value in other strategic domains, particularly where limited resources must be optimally matched to dynamic, high-stakes demands.

Homeland Security & Weapon Logistics A location–allocation WTA variant, developed for the Turkish Land Forces, optimizes where to preposition various weapon types across multiple depots. It minimizes setup and transport costs while managing the risk of re-allocating munitions during crises [2]. By extending this to encompass mobile or civil defense units, WTA frameworks can enhance responsiveness in domestic emergencies and terrorist threats.

UAV and Autonomous Systems Recent research explores WTA frameworks tailored to UAV (Unmanned Aerial Vehicles) swarms using game-theoretic multi-strategy approaches [1, 3]. Any system of autonomous agents—like drone fleets or robotic search-and-rescue units—benefits from these allocation heuristics.

Vehicle Routing and Logistics (overall) In vehicle routing problems (VRPs), a fleet of vehicles must service a set of demand points under constraints such as capacity, time windows, or risk exposure. Much like the WTA's assignment of weapons to high-value targets, VRPs optimize the allocation of limited transport assets to priority destinations. Recent work leverages similar heuristics—e.g., large neighborhood search or decomposition-based methods—as in WTA solution strategies to manage dynamic routing under uncertainty [6].

Emergency Evacuation Planning During mass evacuations, emergency managers face complex allocation problems: matching evacuees to exits, shelters, or transportation modes. The structure mirrors WTA: maximizing the number of people safely evacuated (analogous to neutralized targets) under time, direction of danger and resource constraints. Multi-objective WTA approaches—such as Pareto-optimal frontiers balancing time, risk, and throughput—are increasingly adapted in evacuation modeling.

5 Future work

Several promising directions exist for extending this research:

- Investigate more deeply the impact of alternative functions $f(a_i) = h_i$ on the performance of the modified Karasakal model.
- Expand the benchmark by incorporating additional WTA models, including heuristic approaches.
- Explore a multi-stage adaptation of Lu and Chen's algorithm to evaluate its potential under dynamic engagement scenarios. Assess the applicability of WTA frameworks in civilian or non-military domains such as emergency response coordination or dynamic traffic flow management [9, 10].

References

[1] Wenhao Bi, Zhaoxi Wang, Yang Xu, and An Zhang. Weapon-target assignment for unmanned aerial vehicles: A multi-strategy threshold public goods game approach. *Defence Technology*, 2025.

[2] Cihan Çetinkaya and Samer Haffar. A risk-based location-allocation approach for weapon logistics. *Logistics*, 2(2):9, 2018.

[3] Yifan Gao, Lei Zhang, Chuanyue Wang, Xiaoyuan Zheng, and Qianling Wang. An evolutionary game-theoretic approach to unmanned aerial vehicle network target assignment in three-dimensional scenarios. *Mathematics*, 11(19):4196, 2023.

[4] Orhan Karasakal. Air defense missile-target allocation models for a naval task group. *Computers & Operations Research*, 35(6):1759–1770, 2008.

[5] Alexander Kline, Darryl Ahner, and Raymond Hill. The weapon-target assignment problem. *Computers & Operations Research*, 105:226–236, 2019.

[6] Çağrı Koç, Tolga Bektaş, Ola Jabali, and Gilbert Laporte. A hybrid evolutionary algorithm for heterogeneous fleet vehicle routing problems with time windows. *Computers & Operations Research*, 64:11–27, 2015.

[7] Yiping Lu and Danny Z Chen. A new exact algorithm for the weapon-target assignment problem. *Omega*, 98:102138, 2019.

[8] Alan S Manne. A target-assignment problem. *Operations research*, 6(3):346–351, 1958.

[9] Olive Niyomubyeyi, Petter Pilesjö, and Ali Mansourian. Evacuation planning optimization based on a multi-objective artificial bee colony algorithm. *ISPRS International Journal of Geo-Information*, 8(3):110, 2019.

[10] Huaixi Xing and Qinghua Xing. An air defense weapon target assignment method based on multi-objective artificial bee colony algorithm. *Computers, Materials & Continua*, 76(3), 2023.