Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 9228993, 13 pages
https://doi.org/10.1155/2022/9228993

Research Article

Weapon Target Assignment Model for Small Unit Ground Combat Using Mixed Integer Nonlinear Program and Lagrangian Relaxation

Yeonseung Kim,[1] **Jaeyoung Cho**,[2] **Sangwoo Han**,[3] **and Namsuk Cho**[4]

[1]*Service Support Group, Korea-U.S. Combined Forces Command, Seoul, PO BOX 18, Republic of Korea*
[2]*College of Business, Prairie View A & M University, Prairie View, TX 77446, USA*
[3]*Agency for Defense Development, Daejeon, PO Box 35, Republic of Korea*
[4]*Operations Research, Korea National Defense University, Nonsan, Republic of Korea*

Correspondence should be addressed to Namsuk Cho; ncho64@gmail.com

Received 17 July 2022; Revised 20 November 2022; Accepted 22 November 2022; Published 24 December 2022

Academic Editor: Zhenbo Wang

The importance of the weapon target assignment problem is increasing as weapon systems are expected to become intelligent and unmanned on the future battlefield. This study discusses the weapon-target assignment in small-scale ground force combat, which has not been addressed much in other studies. In ground combat, target assignment means the start of an engagement, so assignment should be made taking into account not only the probability of killing an enemy but also the probability of surviving friendly forces. We introduce the WTAG (WTA ground) model that reflects these characteristics and describes methodologies for solving the problem including mixed integer nonlinear problems and Lagrangian relaxation approaches. Our finding is that the algorithm presented can efficiently provide the bounds of the problem in a practical size.

1. Introduction

In the future battlefield, the use of unmanned weapon systems will be further expanded. UAV (unmanned aerial vehicle) and drones are currently being used on the battlefield, and the use of unmanned weapons on the ground, called UGV (unmanned ground vehicle), will play an important role in the near future. Unmanned weapon system has to make various decisions autonomously, in the chaotic battle field, so the high level of situational judgement is required. Accordingly, the movement formation and optimal movement path to achieve the reconnaissance goal of the UAV cluster, stealth effect from detection, relay communication, tracking, and hitting targets are being studied recently. In particular, the target assignment for UAV has been in the spotlight in academic and military communities.

On the other hand, UGV is being developed to assist infantry small combat, such as reconnaissance of dangerous areas, ammunition delivery, casualty transport, and close combat with the enemy. Here, in order to efficiently perform the close combat next to the maneuvering infantry, it is important which targets are allocated to UGV in a very short time. In addition, most recently, studies on counter-UAV, protecting high-value assets from attacks by unmanned weapon system, have also been conducted naturally with the development of UAV research. That is, if targets are assigned to attack UAV, a defense system has to be established in consideration of the assignments, from the defense point of view.

Considering these points, the most basic but important decision is target assignment. Target assignment is simply a process of matching weapons with targets. This problem may be solved with a few simple rules such as "nearest target first" or "follow predefined priority of targets." However, in an effort to find an exact answer to this problem, an optimization problem, called the WTA (weapon target

assignment), has been studied since the 1950s in the military operations research field. WTA is finding the best weapon-target assignment solution for minimizing the total expected survival value of the targets.

In this research, we introduce the WTA model for small-scale ground combat. The important characteristics that distinguish ground combat from other battles are as follows.

First, a common but not necessary assumption to the general WTA problem is "fire and forget" which means that we are concentrating on the assignment before shooting but do not pay much attention to what happens after that. For this reason, weapons were mainly thought of as missiles and artillery such that they had no direct contact with the target. However, if the infantry is assumed, the assignment of weapons to targets will inevitably entail engagement. Most ground combat takes place at close range, and the fact that friendly forces can see and attack the enemy means that the enemy can also do the same. Second, in ground combat, target assignment can be made differently depending on the operational stage. For example, in security area operations, it is more important to protect one's combat power, so the commander will minimize unnecessary engagement. On the other hand, in the target area operations, the commander will pour our available combat power for a decisive victory. Therefore, in different operational stages, it is necessary to construct the model for a different purpose. Lastly, in small unit ground combat, target assignment decision must be made faster. In the case of missile or artillery fire assignment, there is time to carefully plan for allocation. However, in ground battle, commanders have to judge the situation quickly and respond when they encounter unexpected environment.

In this study, we propose the WTAG (weapon target assignment optimization model for small unit ground battle) that reflects the above considerations. The remainder of the paper is organized as follows. In Section 2, we introduce previous WTA studies and explain the uniqueness of our research. In Section 3, we describe the WTAG problem and suggest solution methods. In Section 4, we provide the computational results. In the last section, the conclusion of the study and future directions are explained.

2. Literature Review

The origins of the WTA problem date back to the 1950s. Manne [1], Braford [2], and Day [3] developed and standardized the problem in the 1950s and 1960s. Since then, it has developed with the following several characteristics. First, it is NP-complete. Lloyd and Witsenhausen established the NP-completeness of the problem in 1986 [4]. So, as the instance size increases, the time of the algorithm for finding the optimal solution increases exponentially, and moreover, there is no optimal algorithm so far. Second, the WTA problem is a MINLP (mixed integer nonlinear programming). LP (linear programming) has an exact solution method like the simplex method, but MINLP does not. Thus, the relaxed method is usually used when solving the MINLP. Many researchers have suggested different formulations and devised exact algorithms to solve the problem. Relatively recently, there have been attempts by applying heuristic algorithms. Among them, an exact method had been researched for some of the following special cases.

First of all, Manne [1] explained that it was possible to devise a LP formulation of the WTA problem under the assumption that all weapons are identical. And, denBroeder et al. [5] came up with an algorithm that provides an optimal solution to Manne's problem. Day [3] researched the dimensionality and complexity of the problem. Day [3] considerably reduced the dimensions of the problem by decomposing the assignment problem into smaller targeting problems, and it provides information for solving larger targeting problems.

Bracken and McCormick [6] handled two kinds of weapon allocation problems and suggested the LP approximation method. The first problem is to maximize target damage, and the second problem is to minimize cost with constraints that at least a specified target damage has to be needed. These traditional WTA problems, which is called static WTA problem, with the objective function of minimizing the threat or survival probability of the enemy were also suggested by Kolitz [7], Lemus and David [8], and Zu [9]. Chang et al. [10] suggested a nearly optimal algorithm for the large-scale WTA problem. Assuming that at least one weapon can be assigned to one target, a nearly optimal solution can be obtained. More recently, Ni et al. [11] devised the Lagrangian relaxation method and decomposed the Lagrangian function into smaller subproblems. And each problem can be solved more easily than the original problem. Andersen et al. [12] devised a specialized exact algorithm that is based on the piecewise linear approximation of the objective function which is computationally solvable.

Meanwhile, dynamic WTA is a multistage problem, and it considers the weapons and targets to be attacked for subsequent stages. Soland et al. first suggested the dynamic version of the WTA problem with the assumption that there is a one kind of asset, and the assets are identical in each stage. The dynamic problem has been studied by Wacholder et al. [13], Christ et al. [14], and Hosein and Athans [15]. Especially, Hosein [16] devised the dynamic versions of the asset-based and target-based WTA. Hosein used the maximum marginal return algorithm and suboptimal algorithm, and it is efficient to obtain the bounds of the optimal solution. Silav et al. [17] developed biobjective model for dynamic WTA, and a linearization approach is applied to the objective function. And the result showed that the approach can be used for real-time constrained WTA problem and can assist decision-making in a relatively short time.

Also, some of the heuristic method researches were suggested to solve the problem. Xin et al. [18] devised the Tabu search for finding a solution, Madni and Andrecut [19] used simulated annealing and threshold-accepting methods, and Xin et al. [20] suggested an efficient rule-based constructive heuristic method. Hong et al. [21] compared three heuristic algorithms, ACO (ant colony optimization), PSO (particle swarm optimization), and GA (genetic algorithm), and found out that ACO was outperforming in overall scenarios. And many researchers have tried to improve the

original algorithms for better optimality and solving time. Hong et al. [21] proposed a population initialization algorithm to improve the ability of GA for solving WTA. When initializing the algorithm, the characteristics of the WTA are reflected, the dominant gene is inherited, and the search space is set wide so that a high-quality solution can be found efficiently. Li et al. [22] modified Pareto ACO, and in order to avoid defects in the original ACO, Li et al. tried to use an improved movement probability rule, a dynamic evaporation rate strategy, and a global updating rule of pheromone. Chang et al. [23] used ABC (artificial bee colony) algorithm for improving the timeliness of the dynamic WTA. Chang et al. progressed the original ABC algorithm, and the initial solution could be obtained more effectively. These heuristic methods may not be able to guarantee the optimal solution, but it is effective to find a local solution to the MINLP problem in a short time.

So far, we reviewed representative WTA studies, and our research differs from other studies as follows. First, we propose a new mathematical model that has not been discussed before, reflecting the characteristics of ground combat described in Section 1. More specifically, our model also considers the survivability of weapons, which will become an important characteristic as battles between autonomous unmanned systems become more frequent in the future. Second, in terms of methodology for solving the problem, we have to admit that we borrowed the core idea of Ni et al. [11] that uses Lagrangian formulation and divides objective function into a few easily solvable subproblems. However, since the similar idea was applied to an entirely different problem, a distinct approach was needed in various aspects, such as the construction of Lagrangian functions and obtaining of solution for subproblems. Moreover, we described algorithms for finding Lagrangian multipliers with a comparative experiment, which was not covered in detail by Ni et al.

3. Methodology

3.1. The WTAG Problem. In this section, we define the WTAG problem and explain solving strategy. The notations and formulation of the problem are as shown in Table 1.

Then, our WTAG problem can be defined as follows:

$$\min \sum_{j\in\mathscr{T}} V_j \prod_{i\in\mathscr{W}} \left(1 - p_{\mathrm{ij}}\right)^{x_{\mathrm{ij}}}, \tag{1}$$

$$s.t. \quad \sum_{j\in\mathscr{T}} x_{ij} \le 1 \forall i \in \mathscr{W}, \tag{2}$$

$$\sum_{i\in\mathscr{W}} V_i \prod_{j\in\mathscr{T}} \left(1 - \frac{s_{ij}}{\sum_{i'\in\mathscr{W}} x_{i'j}}\right)^{x_{ij}} \ge \beta, \tag{3}$$

$$x_{ij} \in \{0,1\} \forall i \in \mathscr{W}, \forall j \in \mathscr{T}. \tag{4}$$

The objective function is to minimize the expected target value, and inequality (2) means that all weapons can be assigned only once at a time. That is, weapons only can attack no more than one target. The inequality (3) is a single new constraint added to the original WTA problem. The left-hand side estimates the friendly assets' remaining value after an engagement, and it is bounded by constant β in right-hand side. The constant $\beta \in \mathbb{R}$ is useful for implementing operation phases in military. For example, commanders in the security area may want to keep their combat power in reserve for the decisive operation. In this case, we may save the combat power of friendly forces by using large β. In contrast, in decisive operations, commanders will try to achieve their objective even if they consume almost all of their combat power. In this case, we may make an engagement actively using small β.

There are two important assumptions in making the inequality (3). First, we determine that the enemy's counter-attack is an immediate reaction. Figure 1(a) shows such a case. In this case, if $W1$ is assigned to $T1$, $T1$ also counter-attack $W1$. In contrast, Figure 1(b) shows nonimmediate reaction case. In this case, $T1$ can attack either $W1$ or $W2$, depending on the intention of the enemy. In this research, the enemy's response is assumed to be (a).

Another important assumption is that each agent only can be assigned to no more than one enemy since an agent is considered as a weapon (e.g., unmanned ground vehicle, machine gunner) not a unit (e.g., squad, platoon). So, in Figure 2, the assignments of friendly assets are valid, but those of enemy targets are not. Since we assume that each enemy agent is also a weapon, not a unit, it can counterattack against only one friendly asset.

Based on two assumptions, it is reasonable to divide parameter s_{ij} by the number of assignments to target j in the sense that it is a kind of expected value. For instance, in Figure 2, the expected probability of kill asset $W1$ by target $T1$ can be estimated by $s_{w_1,t_1}/2$ since $T1$ can either counter-attack to $W1$ or $W2$. In a similar way, probability of kill asset $W2$ by target $T1$ is $s_{w_2,t_1}/2$. In this case, we assume that the probability that targets counter-attack to friendly assets follows a uniform distribution. However, because inequality (3) has the denominator that can have zero (i.e., none of the weapons are assigned to a certain target), it is not properly defined. We, hence, introduce a reformed formulation, WTAG_E (WTAG_Extended) which gives the same solution to WTAG but deals with the zero-divide issue using a surrogate modeling technique.

$$\begin{aligned} &\min \sum_{j\in\mathscr{T}} V_j \prod_{i\in\mathscr{W}} \left(1 - p_{ij}\right)^{x_{ij}} \\ &s.t. \quad \sum_{j\in\mathscr{T}} x_{ij} \le 1 \forall i \in \mathscr{W}, \end{aligned} \tag{5}$$

$$\sum_{i\in\mathscr{W}} x_{ij} \le M * w_j, \forall j \in \mathscr{T}, \tag{6}$$

$$w_j \le \sum_{i\in\mathscr{W}} x_{ij}, \forall j \in \mathscr{T}, \tag{7}$$

$$t_{ij} \le s_{ij} + M * w_j, \forall i \in \mathscr{W}, \forall j \in \mathscr{T}, \tag{8}$$

Table 1: Definitions and notations.

Sets		
$\mathscr{W} = \{0, 1, 2, \ldots, m\}$	Set of friendly assets/units/weapons	
$\mathscr{T} = \{0, 1, 2, \ldots, n\}$	Set of enemy units/targets	
Parameters		
p_{ij}	Probability of kill target j by asset i	$\forall i \in \mathscr{W}, \forall j \in \mathscr{T}$
s_{ij}	Probability of kill asset i by target j	$\forall i \in \mathscr{W}, \forall j \in \mathscr{T}$
V_i	Value of asset i	$\forall i \in \mathscr{W}$
V_j	Value of target j	$\forall j \in \mathscr{T}$
Decision variable		
$x_{\text{ij}} =$	$\begin{cases} 1, & \text{asset } i \text{ is assigned to target } j, \\ 0, & o/w. \end{cases}$	$\forall i \in \mathscr{W}, \forall j \in \mathscr{T}$

$$t_{ij} \geq s_{ij} - M * w_j, \forall i \in \mathscr{W}, \forall j \in \mathscr{T}, \tag{9}$$

$$t_{ij} \leq \frac{s_{ij}}{\sum_{i' \in \mathscr{W}} x_{i'j} + \epsilon} + M * \left(1 - w_j\right), \forall i \in \mathscr{W}, \forall j \in \mathscr{T}, \tag{10}$$

$$t_{ij} \geq \frac{s_{ij}}{\sum_{i' \in \mathscr{W}} x_{i'j} + \epsilon} - M * \left(1 - w_j\right), \forall i \in \mathscr{W}, \forall j \in \mathscr{T}, \tag{11}$$

$$\begin{aligned} &\sum_{i \in \mathscr{W}} V_i \prod_{j \in \mathscr{T}} \left(1 - t_{ij}\right)^{x_{ij}} \geq \beta, \\ &x_{ij} \in \{0, 1\}, \forall i \in \mathscr{W}, \forall j \in \mathscr{T}, \\ &w_j \in \{0, 1\}, \forall j \in \mathscr{T}. \end{aligned} \tag{12}$$

In this formulation, M is a big number, and ϵ is a small number. The decision variable $t_{\text{ij}} \in \mathbb{R}^+$ and $w_j \in \{0, 1\}$ is added in the formulation. The t_{ij} is a positive value that replaces the term $s_{\text{ij}}/\sum_{i' \in \mathscr{W}} x_{i'j}$, and w_j is a binary variable required to define t_{ij}. If $\sum_{i' \in \mathscr{W}} x_{i'j}$ is zero, w_j has to be zero by the inequality (8), and t_{ij} has to be s_{ij} by the inequality (9) and (10). At this moment, the inequality (11) and (12) does not force anything to the bound for t_{ij}. And if $\sum_{i' \in \mathscr{W}} x_{i'j} \geq 1$, w_j has to be 1 by the inequality (7), and t_{ij} has to be $s_{\text{ij}}/\sum_{i' \in \mathscr{W}} x_{i'j}$ by the inequality (11) and (12). Similarly, the inequality (9) and (10) does not make meaningful bound for t_{ij} at this time.

Unfortunately, both the WTAG and WTAG_E problem are NP-complete. Hence, it is almost not possible to solve the formulation efficiently. In the remainder of the section, we discuss an efficient algorithm for solving WTAG. As we mentioned before, we mainly refer to Ni et al. [11] for developing the idea.

3.2. Efficient Algorithm. The key idea of the algorithm is to reform the WTAG problem by using the Lagrangian relaxation method. And the relaxed objective function is divided into three subproblems.

3.2.1. Lagrangian Relaxation. Before we construct the Lagrangian relaxation of the WTAG, we first transpose a term that has power using a natural log.

$$\begin{aligned} y_j &= \prod_{i \in \mathscr{W}} \left(1 - p_{ij}\right)^{x_{ij}} \forall j \in \mathscr{T}, \\ &\rightarrow \ln y_j = \sum_{i \in \mathscr{W}} \ln\left(1 - p_{ij}\right) \cdot x_{ij} \forall j \in \mathscr{T}. \end{aligned} \tag{13}$$

As the natural log is a nondecreasing monotonic function, it does not spoil the solution. Here, p_{ij} is the probability of kill target j by asset i, so y_j means the survival probability of the target j. In this way, the survival probability of the friendly assets, z_i, can be transposed and rearranged as follows:

$$z_i = \prod_{j \in \mathscr{T}} \left(1 - \frac{s_{ij}}{\sum_{i' \in \mathscr{W}} x_{i'j}}\right)^{x_{ij}}, \forall i \in \mathscr{W}, \tag{14}$$

$$\rightarrow \ln z_i = \sum_{j \in \mathscr{T}} \ln\left(1 - \frac{s_{\text{ij}}}{\sum_{i' \in \mathscr{W}} x_{i'j}}\right) \cdot x_{ij}, \forall i \in \mathscr{W}, \tag{15}$$

$$\Leftrightarrow \ln z_i = \sum_{j \in \mathscr{T}} \ln\left(1 - \frac{s_{ij}}{u}\right) \cdot x_{ij}, \forall i \in \mathscr{W}, 1 \leq u \leq |\mathscr{W}|, \tag{16}$$

We substitute the denominator term in (19) using surrogate constant u where $1 \leq u \leq |\mathscr{W}|$. Denominator term is a number of assignments that allocated to target j. As we discussed, such number can be zero. However, in this equation, as a term x_{ij} is multiplied, if there is no assignment, such term automatically has zero value. Hence, the bound of u is proper. The equations related to the transpositions are added to the WTAG formulation.

$$\begin{aligned} &\min \sum_{j \in \mathscr{T}} V_j \cdot y_j, \\ &\text{s.t.} \quad \sum_{j \in \mathscr{T}} x_{ij} \leq 1 \forall i \in \mathscr{W}, \end{aligned} \tag{17}$$

$$\sum_{i \in \mathscr{W}} V_i \cdot z_i \geq \beta, \tag{18}$$

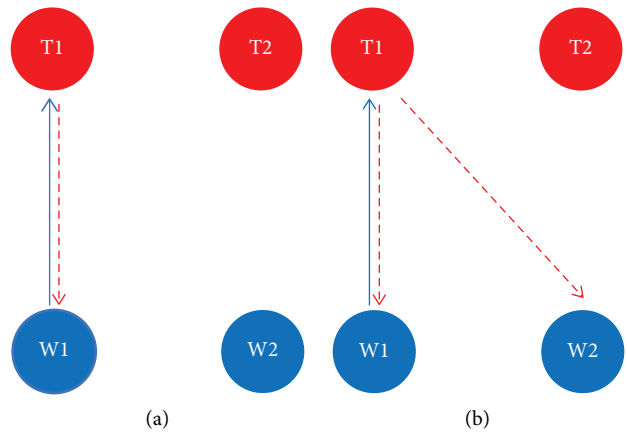


Figure 1: Enemy's counter-attack cases. (a) Immediate reaction and (b) nonimmediate reaction.

$$\ln y_j = \sum_{i\in\mathscr{W}} \ln\left(1 - p_{ij}\right)\cdot x_{ij}, \forall j \in \mathscr{T}, \tag{19}$$

$$\ln z_i = \sum_{j\in\mathscr{T}} \ln\left(1 - \frac{s_{ij}}{u}\right)\cdot x_{ij} \forall i \in \mathscr{W}, 1 \le u \le |\mathscr{W}|, \tag{20}$$

$$x_{ij} \in \{0,1\}, \forall i \in \mathscr{W}, \forall j \in \mathscr{T}. \tag{21}$$

In this formulation, the y_j and z_i are also decision variables, as well as x_{ij}. Variables y_j and z_i are nonnegative as it represents a probability. Now, the WTAG Lagrangian function is as follows:

$$\begin{aligned}\min \mathscr{L}(x,y,z,\lambda) = &\sum_{j\in\mathscr{T}} V_j\cdot y_j - \lambda^{\beta}\left(\sum_{i\in\mathscr{W}} V_i\cdot z_i - \beta\right)\\ &+ \sum_{j\in\mathscr{T}} \lambda_j^y\left(\sum_{i\in\mathscr{W}} \ln\left(1-p_{ij}\right)\cdot x_{ij} - \ln y_j\right)\\ &+ \sum_{i\in\mathscr{W}} \lambda_i^z\left(\sum_{j\in\mathscr{T}} \ln\left(1-\frac{s_{ij}}{u}\right)\cdot x_{ij} - \ln z_i\right),\end{aligned} \tag{22}$$

$$\text{s.t.} \sum_{j\in\mathscr{T}} x_{ij} \le 1 \forall i \in \mathscr{W}, \tag{23}$$

$$\lambda^{\beta} \ge 0, \lambda_i^z, \lambda_j^y - \text{free} \forall i \in \mathscr{W}, \forall j \in \mathscr{T}. \tag{24}$$

The equations (23)–(25) are moved to the objective function with corresponding multipliers. Here, λ^{β} is one dimensional vector, and λ_j^y, λ_i^z is $|\mathscr{T}|$, $|\mathscr{W}|$ dimensional vector, respectively. So, integrated multiplier vector λ is a $1 + |\mathscr{T}| + |\mathscr{W}|$ dimensional vector. The important point here, theoretically, is that if this problem is unbounded, it means that there are implicit restrictions on the dual variables, λ vector. It has to be applied when solving dual problem using subgradient method. Even we are given multiplier vector, it is still a mixed integer nonlinear problem. Accordingly, in the next section, we propose a solving method by decomposing the objective function into three subproblems.

3.2.2. Subproblems. The Lagrangian function can be separated according to the decision variables, x_{ij}, y_j, z_i, and we define each as $L_1(x)$, $L_2(y)$, $L_3(z)$.

$$\begin{aligned}L_1(x) = \min &\sum_{j\in\mathscr{T}} \lambda_j^y\left(\sum_{i\in\mathscr{W}} \ln\left(1-p_{ij}\right)\cdot x_{ij}\right) + \sum_{i\in\mathscr{W}} \lambda_i^z\left(\sum_{j\in\mathscr{T}} \ln\left(1-\frac{s_{ij}}{u}\right)\cdot x_{ij}\right) + \lambda^{\beta}\cdot\beta,\\ \Leftrightarrow &\sum_{j\in\mathscr{T}}\sum_{i\in\mathscr{W}}\left(\lambda_j^y\cdot\ln\left(1-p_{ij}\right) + \lambda_j^z\cdot\ln\left(1-\frac{s_{ij}}{u}\right)\right)\cdot x_{ij} + \lambda^{\beta}\cdot\beta,\end{aligned} \tag{25}$$

$$L_2(y) = \min \sum_{j\in\mathscr{T}}\left(V_j\cdot y_j - \lambda_j^y\cdot\ln y_j\right), \tag{26}$$

$$L_3(z) = \min -\sum_{i\in\mathscr{W}}\left(\lambda^{\beta}\cdot V_i\cdot z_i + \lambda_i^z\cdot\ln z_i\right). \tag{27}$$

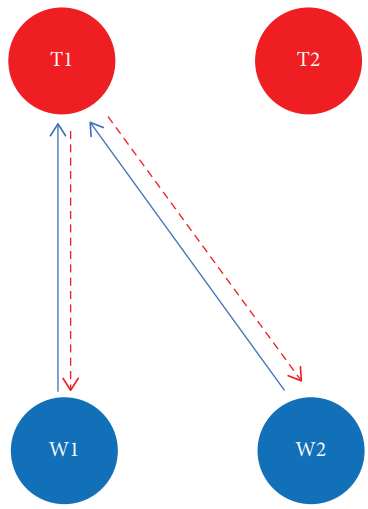


Figure 2: Assignments of the enemy target.

We now describe how to solve each subproblem.

3.2.3. Solving Subproblem $L_1(x)$. As $L_1(x)$ term includes variable x, we consider constraint of x_{ij}.

$$L_1(x) = \min \sum_{j\in\mathcal{T}} \sum_{i\in\mathcal{W}} \left(\lambda_j^y \ln\left(1 - p_{ij}\right) + \lambda_i^z \ln\left(1 - \frac{s_{ij}}{u}\right)\right) x_{ij} + \lambda^\beta \beta,$$
$$\text{s.t,} \quad \sum_{j\in\mathcal{T}} x_{\text{ij}} \le 1 \forall i \in \mathcal{W}. \tag{28}$$

Given multiplier, the objective function is a linear function of x_{ij}, and the constraint matrix is a totally unimodular (TU). Therefore, we solve $L_1(x)$ as linear programming.

3.2.4. Solving Subproblem $L_2(y)$. $L_2(y)$ formulation, (28), can be decomposed into $|\mathcal{T}|$ one-dimension discrete problems. And we find a solution for each subproblem.

$$SL_{2j}\left(y_j\right) = \min V_j \cdot y_j - \lambda_j^y \cdot \ln y_j, \forall j \in \mathcal{T}. \tag{29}$$

As discussed, y_j is the survival probability of the target j. If the asset i that has the largest p_{ij} attacks target j, the survival probability of the target j will have smallest value, γ_j. In contrast, if the asset i that has the smallest p_{ij} attacks target j, the survival probability of the target j will have largest value, α_j. We define bound of the y_j as follows:

$$\alpha_j = 1, \forall j \in \mathcal{T} \tag{30}$$

$$\gamma_j = \left(1 - \left(\max_{1\le i\le \mathcal{W}} p_{\text{ij}}\right)\right)^{|\mathcal{W}|}, \forall j \in \mathcal{T} \tag{31}$$

$$\gamma_j \le y_j \le \alpha_j, \forall j \in \mathcal{T} \tag{32}$$

Since there may not be any assets that attack target j, the maximum value of the survival probability of the target j is 1. The solution y_j can be obtained depending on the shape of the objective function graph within the bound, $\gamma_j \le y_j \le \alpha_j$. The shape of the graph can be different from the value of λ_j^y because the V_j and y_j are positive. If $\lambda_j^y < 0$, the $SL_{2j}(y_j)$ graph is monotonic increasing as y_j increases, as seen in Figure 3(a). In this case, the minimum value within the range of y is obtained when $y_j = \gamma_j$. If $\lambda_j^y = 0$, minimum value is obtained when $y_j = \gamma_j$. Lastly, if $\lambda_j^y > 0$, $\text{SL}_{2j}(y_j)$ is convex, and the minimum value is obtained at the critical point where the gradient is 0, as seen in Figure 3(b). And it is important to check whether the critical point is in the range of y_j. Accordingly, the solution can be obtained by the conditions as follows:

$$y_j^* = \begin{cases} \alpha_j, & \alpha_j \le \dfrac{\lambda_j^y}{V_j}, \\ \dfrac{\lambda_j}{V_j}, & \gamma_j \le \dfrac{\lambda_j^y}{V_j} \le \alpha_j, \\ \gamma_j, & \dfrac{\lambda_j^y}{V_j} \le \gamma_j. \end{cases} \tag{33}$$

3.2.5. Solving Subproblem $L_3(z)$. The $L_3(z)$, (28), can be decomposed into $|\mathcal{W}|$ one-dimension discrete problems.

$$\text{SL}_{3i}(z_i) = \min\ -\lambda^\beta \cdot V_i \cdot z_i - \lambda_i^z \cdot \ln z_i, \forall i \in \mathcal{W}. \tag{34}$$

As z_i represents the survival probability of asset i, we define the bound of the z_i as follows:

$$\alpha_i = 1, \forall i \in \mathcal{W}, \tag{35}$$

$$\gamma_i = \left(1 - \left(\max_{1\le j\le \mathcal{T}} \frac{s_{\text{ij}}}{u}\right)\right)^{|\mathcal{T}|}, 1 \le u \le |\mathcal{W}|, \forall i \in \mathcal{W}, \tag{36}$$

$$\gamma_i \le z_i \le \alpha_i, \forall i \in \mathcal{W}. \tag{37}$$

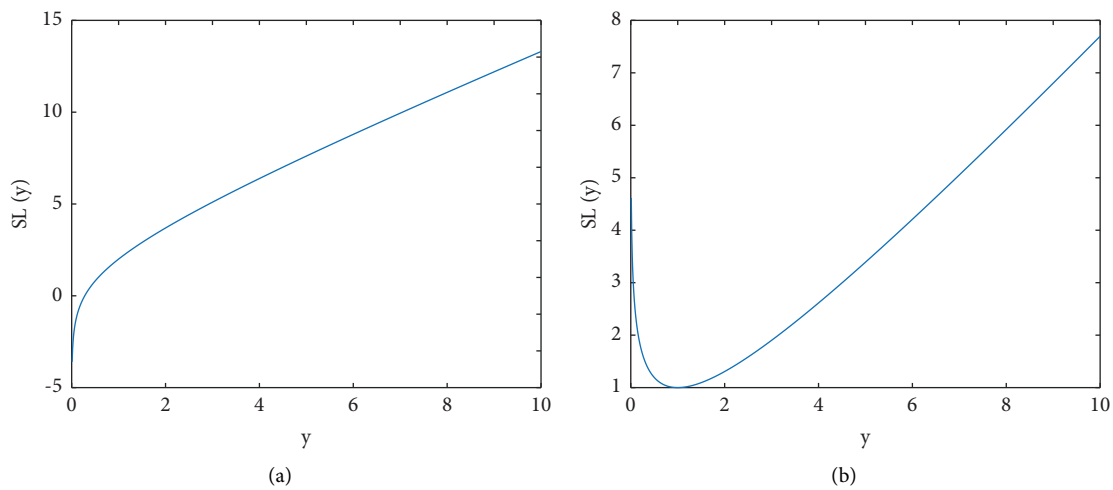


FIGURE 3: Dynamics of $\mathrm{SL}_{2j}(y_j)$ with different λ values. (a) $\lambda_j^y < 0$, (b) $\lambda_j^y > 0$.

The solution to subproblem (33) can be obtained in a similar way of $L_2(y)$ by observing the dynamics of $L_3(z)$ with respect to z.

As shown in Figure 4(a), when $\lambda_i^z < 0$, we have a concave function. However, different to the case of subproblem $L_2(y)$, we solve minimization problem, and the critical point is no more optimal. And if $\lambda_i^z \geq 0$, it shows monotonic nonincreasing, as seen in Figure 4(b). Therefore, we define the analytical solution of both cases as follows:

$$z_i^* \in \underset{z}{\operatorname{argmin}}\left(L_3(\alpha_i), L_3(\gamma_i)\right). \tag{38}$$

However, γ_i in (35) has a variable u which is indeed $u = \sum_{i' \in \mathscr{W}} x_{i'j}$. And this means that subproblem (33) is not separated from a variable x. Our heuristic remedy is to use $u = |\mathscr{W}|$. Although this setting obviously will change the solution, it at least provides a feasible solution according to the following logic. Looking back on the previous discussion, z_i is survival probability of weapon i. If $u = |\mathscr{W}|$, z_i has the greatest value. Let's define such z_i to be $z_{\max}$. And we substitute $z_{\max}$ into (21), then following is satisfied.

$$\sum_{i \in \mathscr{W}} V_i z_{u_{\max}} \geq \sum_{i \in \mathscr{W}} V_i z_{\mathrm{opt}} \geq \beta, \tag{39}$$

where z_{opt} is the solution where we set $u = \sum_{i' \in \mathscr{W}} x_{i'j}$

3.2.6. Line Search Method. Finding exact multipliers in Lagrangian relaxation is another optimization problem, and we introduce two algorithms for solving the Lagrangian dual problem. The first approach is to find a local λ solution using iterative first-order line search, as described in Algorithm 1.

In algorithm 1, $\lambda_k = [\lambda_j^y, \lambda_i^z, \lambda^\beta], \forall i \in \mathscr{W}, \forall j \in \mathscr{T}$. Through numerical experiments, we found that the choice of step length α_k had the greatest effect on the performance of the algorithm. We empirically decided to adopt the ideas of Fan et al. [24] and Feng and Liao [25].

$$\alpha_k = \gamma\left(\frac{1}{k+1}\right)\left(\frac{\|\lambda_k\|^{\max}}{\|\xi_k\|^{\max}}\right), 0 \leq \gamma \leq 1, \tag{40}$$

where $\|\lambda_k\|^{\max}$ is the maximum of multiplier for coupling constraint, and $\|\xi_k\|^{\max}$ is a maximum of subgradient for coupling constraint. In this algorithm, as described in line 8, it terminates the search if the gradient norm (i.e., search direction) is less than epsilon, 0.2 in this study. Moreover, as a result of experiments, we empirically suggest using one vector for the initial choice of λ_k.

3.2.7. Subgradient Method. Another method for solving the Lagrangian dual problem is the subgradient method, and the pseudocode of the method for our problem is shown in Algorithm 2.

In algorithm 2, $\lambda_k = [\lambda_j^y, \lambda_i^z, \lambda^\beta], \forall i \in \mathscr{W}, \forall j \in \mathscr{T}$ at iteration step k. We define two initial lambda vectors nearly symmetric about the origin for the purpose of preventing the LP problem in line 5 is unbounded. The specific numbers in the initial lambda are determined based on parameters of instance which is described in Table 2. As a result of numerous experiments, we suggest to use $\min\{V_j\} \times \max(1 - p_{\mathrm{ij}})$ as the initial choice of lambda and $-(\min\{V_j\} \times \max(1 - p_{\mathrm{ij}}))$ for choice of second iterations. As explained in Section 3.2.1, since $\lambda_\beta \geq 0$ has to be satisfied, we set the last element of the initial λ_2 to zero [26]. As described in line 6, the algorithm stops if one of the two conditions is satisfied. One is to check if the value of the gradient norm (i.e., $\|s_k\|$) gets closer to 0, and the second is to see if the $\mathscr{L}_0$ value doesn't get any better for a certain period of iterations (i.e., $\|\mathscr{L}_0^{k+1} - \mathscr{L}_0^*\| / \|\mathscr{L}_0^k - \mathscr{L}_0^*\| < \epsilon$ where $\mathscr{L}_0^k$ is solution of k th iteration of LP problem in line 5 and $\mathscr{L}_0^*$ is best solution so far. In this study, $\epsilon = 0.1$) [27]. Our conjecture for both WTAG_LS and WTAG_SG is that first-order information (i.e., $\nabla\lambda_k$) does not provide good information since we solve 3 subproblems in discrete manner.

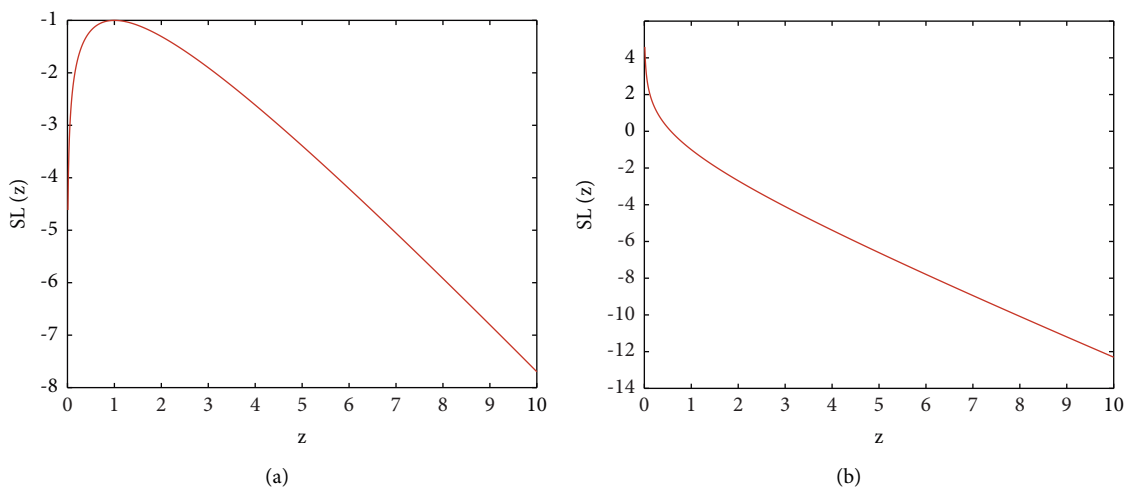


FIGURE 4: Dynamics of $SL_{3i}(z_i)$ with different λ values. (a) $\lambda_i^z < 0$, (b) $\lambda_i^z > 0$.

(1) Set iteration number $k \leftarrow 1$, initial $\lambda_1 \leftarrow [1, 1, \ldots, 1, 1]$
(2) Solve 3 subproblems (solution: $\hat{x}, \hat{y}, \hat{z}$)
(3) Compute p_k (step direction) $\triangleright p_k = \nabla \lambda_k$
(4) Compute α_k (step size) $\triangleright \alpha_k = \gamma(1/k+1)\|(\lambda_k\|^{\max}/\|p_k\|^{\max})$, $0 \le \gamma \le 1$
(5) **if** $\|p_k\| > \epsilon$ **then** $\triangleright \epsilon =$ tolerance
(6) $\lambda_{k+1} \leftarrow \lambda_k + \alpha_k p_k$
(7) Go line 2 and $k \leftarrow k+1$
(8) **else if** $\|p_k\| \le \epsilon$ **then**
(9) STOP
(10) **end if**

ALGORITHM 1: Line search method: WTAG_LS

4. Computational Results

In order to verify the effectiveness of the algorithm, we conduct various experiments. Again, so far, we proposed three different variations of WTAG. All these models have in common that they are for solving WTAG, but WTAG_E is MINLP formulation, and both WTAG_LS and WTAG_SG are Lagrangian-based algorithms. We use BARON (version 20.4.14) with default setting (optimality gap 99% and CPU time limit 4000 sec) under GAMS (version 32.2) environment for solving WTAG_E. Two Lagrangian algorithms (i.e., WTAG_LS and WTAG_SG) are implemented using MATLAB (R2020b 9.9.0). All experiments are conducted on a CPU (Intel i5-10210U) with RAM (16 GB).

4.1. Instances. This section describes the instance used in the experiments. First, we create small-size random instances where parameters are uniformly distributed on a given bound described in Table 2. We choose uniform distribution because military data should not be disclosed to the public, and uniform sampling is preferable for validating the robustness of the algorithm by avoiding being biased induced by data selection. The bound in Table 2 is also an arbitrary number, but we set a lower bound of p_{ij}, s_{ij} to 0.4 since the situation where probability of kill is very small is unrealistic. Examples of instances created by these rules are presented in Table 3.

4.2. Result of Small Instances. Computational results of 2 weapons and 2 targets instances are described in Table 4. Table 4 shows the objective function value, solution, and CPU elapsed time of three different models. Because WTAG_E is a nonlinear formulation, it does not guarantee a global optimality. However, in small-size instances, as described in Table 4, it may provide an optimal solution. All WTAG_E solutions in Table 4 are turned out to be optimal (i.e., confirmed by enumerating all possible solutions). Besides, WTAG_E was the fastest in time. On the other hand, the two heuristic methods do not outperform in terms of optimality and time. The optimality of WTAG_LS and WTAG_SG was distributed between approximately 43% and 99.9%. In experiment of small size instances, we find that Lagrangian algorithms provide solid lower bound. However, it did not help much in terms of time or optimality.

4.3. Results of Large Instances. Results of large-size instances are described in Table 5. Up to 6 weapons and 6 targets, the WTAG_E provided a local solution in the shortest CPU

(1) Set iteration number $k \leftarrow 1$
(2) initial $\lambda_1 \leftarrow [20, 20, \ldots, 20, 20]$, $\lambda_2 \leftarrow [-20, -20, \ldots, -20, 0]$
(3) Solve 3 sub-problems (solution: $\hat{x}, \hat{y}, \hat{z}$)
(4) Compute s_k (subgradient) $\triangleright s_k = \nabla \lambda_k$
(5) Solve LP problem of λ (solution: $\hat{\lambda}$)
$$\max, \quad \mathscr{L}_0, \quad \mathscr{L}_0 \leq L(\lambda_k) + s_k^T(\lambda_{k+1} - \lambda_k),$$
(6) **if** $\|s_k\| \leq \epsilon$ or $\widehat{\mathscr{L}_0}$ has no improvement **then** $\triangleright \epsilon =$ tolerance
(7) STOP
(8) **else**
(9) $\lambda_{k+1} \leftarrow \hat{\lambda}$
(10) Go line 3,$k \leftarrow k+1$, Add inequality to LP problem in line 5.
(11) **end if**

Algorithm 2: Subgradient Method: WTAG_SG

Table 2: Instance bound of experiments.

p_{ij}, s_{ij}	V_i, V_j	β
$0.4 \leq p_{ij}, s_{ij} \leq 1$	$60 \leq V_i, V_j \leq 100$	$(1/5)\sum_{j\in\mathscr{T}} V_j \leq \beta \leq (3/5)\sum_{j\in\mathscr{T}} V_j$

Table 3: Instances in 2 weapons and 2 targets.

Instances	$p_{w_1t_1}$	$p_{w_2t_1}$	$p_{w_1t_2}$	$p_{w_2t_2}$	$s_{w_1t_1}$	$s_{w_2t_1}$	$s_{w_1t_2}$	$s_{w_2t_2}$	V_{w_1}	V_{w_2}	V_{t_1}	V_{t_2}	β
Par-2 : 2-1	0.7	0.4	0.6	0.7	0.7	0.4	0.6	0.8	80	90	80	70	90
Par-2 : 2-2	0.9	0.6	0.7	0.5	0.5	0.5	0.5	0.4	100	100	100	100	100
Par-2 : 2-3	0.4	0.5	0.9	0.8	0.8	0.8	0.7	0.9	60	60	90	70	60
Par-2 : 2-4	0.7	0.7	0.5	0.5	0.5	0.7	0.7	0.6	70	90	90	90	60
Par-2 : 2-5	0.4	0.8	0.8	0.5	0.5	0.5	0.5	0.5	80	80	80	80	80
Par-2 : 2-6	0.6	0.8	0.5	0.7	0.7	0.7	0.8	0.5	80	95	85	90	90
Par-2 : 2-7	0.8	0.9	0.7	0.6	0.7	0.5	0.5	0.4	100	100	100	100	100
Par-2 : 2-8	0.4	0.4	0.4	0.5	0.5	0.4	0.8	0.4	100	60	70	70	80
Par-2 : 2-9	0.8	0.7	0.4	0.9	0.4	0.6	0.5	0.5	70	80	90	80	80
Par-2 : 2-10	0.4	0.7	0.4	0.8	0.6	0.5	0.8	0.5	100	100	100	100	100

Table 4: Results of 2 weapons and 2 targets.

Instances	WTAG_E			WTAG_LS				WTAG_SG			
	Obj	x	Time (sec)	Obj	x	Time (sec)	Opt (%)	Obj	x	Time (sec)	Opt (%)
Par-2 : 2-1	84.4	1, 1, 0, 0	0.16	37.64	1, 0, 0, 1	3.02	44.6	37.97	1, 0, 0, 1	6.29	44.9
Par-2 : 2-2	20.0	1, 0, 0, 1	0.10	19.72	1, 0, 0, 1	3.53	98.6	19.97	1, 0, 0, 1	7.82	99.9
Par-2 : 2-3	94.2	0, 0, 1, 1	0.15	50.72	1, 1, 0, 0	2.92	53.8	51.29	1, 1, 0, 0	5.78	54.4
Par-2 : 2-4	98.1	1, 1, 0, 0	0.21	44.44	1, 0, 0, 1	3.17	45.3	44.74	1, 1, 0, 0	6.25	45.6
Par-2 : 2-5	32.0	0, 1, 1, 0	0.13	31.83	0, 1, 1, 0	3.18	99.5	31.99	0, 1, 1, 0	7.64	99.9
Par-2 : 2-6	89.5	0,0,1,1	0.12	38.11	1, 1, 0, 0	2.56	42.6	38.62	1, 0, 0, 1	4.78	43.2
Par-2 : 2-7	40.0	0, 1, 1, 0	0.10	38.21	0, 1, 1, 0	3.58	95.5	39.84	0, 1, 1, 0	8.20	99.6
Par-2 : 2-8	77.0	1, 0, 0, 1	0.09	74.24	1, 0, 0, 1	2.80	96.4	76.99	1,0,0,1	7.34	99.9
Par-2 : 2-9	22.0	1, 0, 0, 1	0.14	21.76	1, 0, 0, 1	2.58	98.9	21.98	1, 0, 0, 1	6.72	99.9
Par-2 : 2-10	112	0, 0, 1, 1	0.10	65.61	1, 1, 0, 0	2.81	58.5	71.15	1, 0, 0, 1	5.92	66.2

time. In this table, the objective value in the WTAG_E column means an integer feasible solution value. For WTAG_LS and WTAG_SG, it is a lower bound for true optimal value. As instance size increases, WTAG_E is not able to obtain a local solution whereas two heuristic methods provide a bound in a relatively shorter time.

In Table 6, we describe the result of WTAG_LS and WTAG_SG for larger instances, and those are not solvable using WTAG_E. In this table, the objective value is the Lagrangian function value, and $\sum_{j\in\mathscr{T}} V_j y_j$ in third column is objective value without the Lagrangian function which is actually the original objective function. The

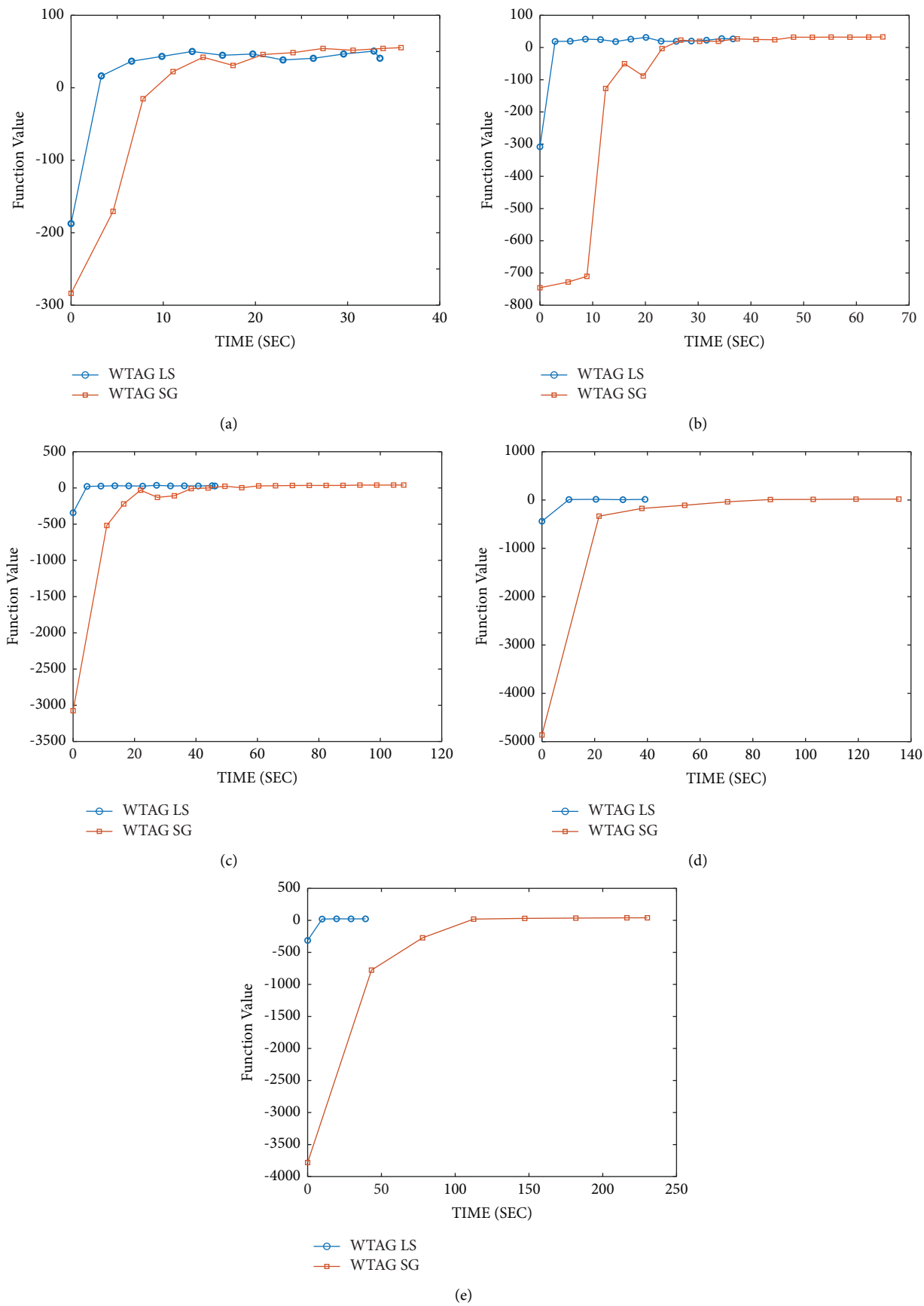


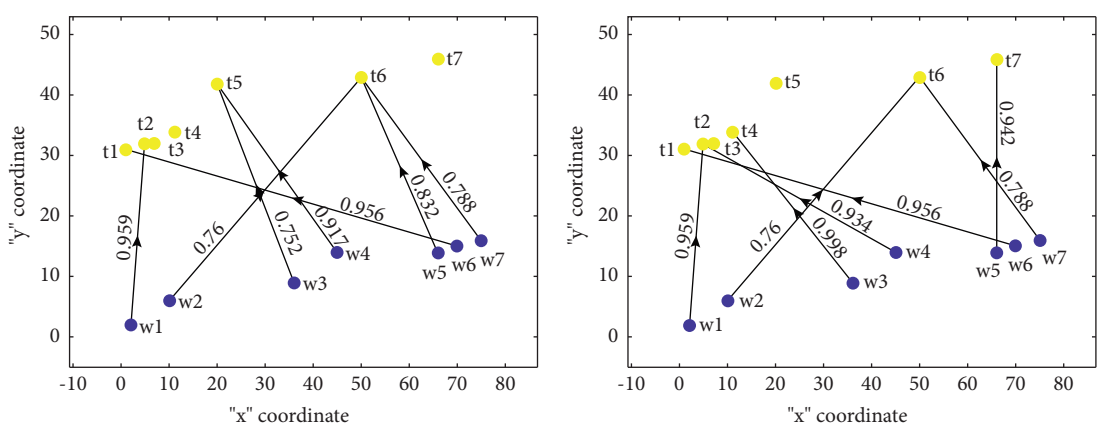
Figure 5: Dynamics of function value with respect to time of two algorithms with large instances. (a) par-5.5, (b) par-6.6, (c) par-7:7, (d) par-8:8, (e) par-9:9.

Table 5: Results of large-size instances 1.

Instances	Size	WTAG_E		WTAG_LS		WTAG_SG	
		Obj	Time (sec)	Obj	Time (sec)	Obj	Time (sec)
Par-3 : 3	3 : 3	32.02	0.50	26.03	21.17	25.89	9.05
Par-4 : 4	4 : 4	80.86	2.16	20.70	24.02	35.26	23.25
Par-5 : 5	5 : 5	111.81	8.92	40.74	33.51	55.35	35.80
Par-6 : 6	6 : 6	90.56	15.53	26.78	36.61	32.85	65.00
Par-7 : 7	7 : 7	104.64	86.81	26.70	46.19	40.00	106.22
Par-8 : 8	8 : 8	—	≥ 4000	13.64	35.19	19.47	135.39
Par-9 : 9	9 : 9	—	≥ 4000	23.22	45.45	39.66	230.23

Table 6: Results of large-size 2.

Instances	WTAG_LS				WTAG_SG			
	Obj	Time (sec)	$\sum_{j\in\mathcal{T}} V_j y_j$	Opt (%)	Obj	Time (sec)	$\sum_{j\in\mathcal{T}} V_j y_j$	Opt (%)
Par-8 : 7	18.10	31.91	22.17	81.6	25.87	105.34	26.51	97.2
Par-8 : 8	13.64	35.19	17.17	79.4	19.47	135.39	19.77	98.4
Par-8 : 10	59.24	42.11	60.50	97.9	76.09	199.89	76.40	99.6
Par-9 : 8	20.66	42.48	22.14	93.3	23.87	127.06	24.63	96.9
Par-9 : 9	23.22	45.45	33.88	68.5	39.66	230.23	39.70	99.4
Par-9 : 11	38.24	43.94	59.97	63.8	69.85	237.50	73.90	94.5
Par-10 : 9	16.74	46.47	18.97	88.2	22.20	199.30	25.05	88.7
Par-10 : 10	34.67	50.73	48.85	71.0	47.35	241.25	48.56	97.5
Par-10 : 11	28.62	58.55	32.96	86.8	39.87	267.71	43.38	91.9
Par-11 : 11	23.09	63.39	29.08	79.4	31.73	409.83	32.00	99.1
Par-12 : 12	31.12	81.06	36.50	85.3	34.00	515.25	36.03	94.4



(a) (b)

Figure 6: Weapon target assignment example. (a) par-7:7 (solved using WTAG LS), (b) par-7:7 (solved using WTAG SG).

optimality gap in the fourth column calculates the gap between these two values. The reason why the value of $\sum_{j\in\mathcal{T}} V_j y_j$ is shown in this table is that, if inequality (23) and (25) is satisfied, such value becomes the feasible bound of this problem. For example, with instance par-9: 11-12, a solution from WTAG_LS satisfies inequality (23), and we can use 59.97 as a feasible bound for the original problem.

Figure 5 shows the dynamics of function value according to the CPU time of WTAG_LS and WTAG_SG. The WTAG_LS obtains solution in a relatively short time compared to WTAG_SG in all cases. Comparing the two algorithms, our conclusion is WTAG_LS which is superior in terms of time.

Figure 6 shows examples of assignments in 2-dimensional spaces. The coordinates of the weapons and targets in the figure are given arbitrarily. And through visualization, we find some considerations that occur in real-world ground combat and that we need to reflect on in the model. For instance, since every weapon has range, assigning a target at a long distance (e.g., t2 and w6 in Figures 6(a) and 6(b)) is not allowed. And we can deal with these issues by adding or modifying the equations in the math model. For example, the rule of not making

assignments that are too far apart can be implemented using a set of inequalities $x_{ij} \leq d_{ij}, \forall i \in W, j \in T$ where d_{ij} is predefined binary parameters that takes 1 if the distance between weapon i and target j is not too far. From this point of view, optimization model for solving weapon-target assignment is useful despite the difficulties it contains. In the practical aspect of the result, since a small ground unit consists of about 10 different types of weapons, the methods are usable referring to the WTAG_LS algorithm which can find the solution in about 1 minute or less.

5. Conclusion

The WTA problem has been studied for a long time in the optimization field, but there have been few attempts to apply it to the ground battle. The purpose of this study is that (1) the new WTAG, WTA for Ground, the problem is formulated by considering the characteristics of ground combat, and (2) the most efficient way to solve this problem was presented.

First, the WTAG problem reflected the important characteristic that engagement should be considered when assigning a target, and this was reflected through a parameter called β that the commander can select according to the operational situations. Second, the Lagrangian relaxation is used to solve the problem efficiently, and then, we divided the problem into three subproblems. To obtain the Lagrangian multipliers, the line search method and the subgradient method are implicated. There are three computational methods proposed to solve this problem. WTAG_E is MINLP, and as the size of the problem increases, it is difficult to obtain a solution. WTAG_LS and WTAG_SG can both provide a dual bound of the problem, even a primal bound under specific conditions. However, not all methods provide global solutions.

Future research directions are as follows. As discussed a bit in Section 4, there are many characteristics of small unit ground combat that need to be further considered in order for the allocation derived from this problem to be put to practical use (e.g., range of the weapon, line of sight, closeness, cross-shoot). Lastly, the Lagrangian-based method can find the solutions that provide bound, but it is not fast enough to find the solution in real-time meaning that the performance of the algorithm in terms of time needs to be improved.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors acknowledge financial support from the project UE202075ID funded by the Agency for Defense Development.

References

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

[2] J. C. Braford, *Determination of optimal assignment of a weapon system to several targets*, AEREITM-9, Vought Aeronautics, Dallas, Texas, 1961.

[3] R. H. Day, "Allocating weapons to target complexes by means of nonlinear programming," *Operations Research*, vol. 14, no. 6, pp. 992–1013, 1966.

[4] S. P. Lloyd and H. S. Witsenhausen, "Weapons allocation is np-complete," in *Proceedings of the 1986 Summer Computer Simulation Conference*, pp. 1054–1058, Reno, Nevada, July 1986.

[5] G. G. denBroeder, R. E. Ellison, and L. Emerling, "On optimum target assignments," *Operations Research*, vol. 7, no. 3, pp. 322–326, 1959.

[6] J. Bracken and G. P. McCormick, "Selected Applications of Nonlinear Programming," Technical report, Research Analysis Corp Mclean va, New York, NY, USA, Technical report, 1968.

[7] S. E. Kolitz, "Analysis of a maximum marginal return assignment algorithm," in *Proceedings of the 27th IEEE Conference on Decision and Control*, pp. 2431–2436, IEEE, Austin, TX, USA, December 1988.

[8] F. Lemus and K. H. David, "An optimum allocation of different weapons to a target complex," *Operations Research*, vol. 11, no. 5, pp. 787–794, 1963.

[9] F. Zu, "Allocation of different weapons to a target complex," in *Proc Int Conf Syst Sci Eng*vol. 11, no. 5, pp. 25–28, ICSSE, 1988.

[10] S. Chang, R. M. James, and J. J. Shaw, "Assignment algorithm for kinetic energy weapons in boost phase defence," in *Proceedings of the 26th IEEE Conference on Decision and Control*, pp. 1678–1683, IEEE, Los Angeles, CA, USA, December 1987.

[11] M. Ni, Z. Yu, F. Ma, and X. Wu, "A Lagrange relaxation method for solving weapon-target assignment problem," *Mathematical Problems in Engineering*, vol. 2011, Article ID 873292, 2110 pages, 2011.

[12] A. C. Andersen, K. Pavlikov, and T. A. Toffolo, "Weapon-target assignment problem: exact and approximate solution algorithms," *Annals of Operations Research*, vol. 312, 2019.

[13] E. Wacholder, "A neural network-based optimization algorithm for the static weapon-target assignment problem," *ORSA Journal on Computing*, vol. 1, no. 4, pp. 232–246, 1989.

[14] J. F. Christ, E. W. Page, and G. A. Tagliarini, "Performance evaluation of a neural network for weapon-to-target assignment," in *Applications of Artificial Neural Networks IV*, vol. 1965, pp. 676–681, International Society for Optics and Photonics, 1993.

[15] P. A. Hosein and M. Athans, "An asymptotic result for the multi-stage weapon-target allocation problem," in *Proceedings of the 29th IEEE Conference on Decision and Control*, pp. 240–245, IEEE, Honolulu, HI, USA, December 1990.

[16] P. A. Hosein, "A Class of Dynamic Nonlinear Resource Allocation Problems," Technical report, Massachusetts Inst Of Tech Cambridge Lab For Information And Decision Systems, Cambridge, United Kingdom, 1989.

[17] A. Silav, E. Karasakal, and O. Karasakal, "Bi-objective dynamic weapon-target assignment problem with stability measure," *Annals of Operations Research*, vol. 311, no. 2, pp. 1229–1247, 2021.

[18] 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.

[19] 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.

[20] 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, 2011.

[21] S.-S. Hong, J.-H. Yoo, S.-J. Park, C.-M. Mun, and M.-M. Han, "A suite of heuristic algorithms for weapon-target allocation in air defense," *Journal of Korean Society for Internet Information*, vol. 13, no. 1, pp. 195-196, 2012.

[22] Y. Li, Y. Kou, Z. Li, An Xu, and Y. Chang, "A modified pareto ant colony optimization approach to solve biobjective weapon-target assignment problem," *International Journal of Aerospace Engineering*, vol. 2017, Article ID 1746124, 14 pages, 2017.

[23] T. Chang, D. Kong, N. Hao, K. Xu, and G. Yang, "Solving the dynamic weapon target assignment problem by an improved artificial bee colony algorithm with heuristic factor initialization," *Applied Soft Computing*, vol. 70, pp. 845–863, 2018.

[24] W. Fan, Y. Liao, J. B. Lee, and Y.-K. Kim, "Evaluation of two Lagrangian dual optimization algorithms for large-scale unit commitment problems," *Journal of Electrical Engineering and Technology*, vol. 7, no. 1, pp. 17–22, 2012.

[25] X. Feng and Y. Liao, "A new Lagrangian multiplier update approach for Lagrangian relaxation based unit commitment," *Electric Power Components and Systems*, vol. 34, no. 8, pp. 857–866, 2006.

[26] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, United Kingdom, 2004.

[27] B. Guta, "Subgradient Optimization Methods in Integer Programming with an Application to a Radiation Therapy Problem," 2003.