applied sciences

MDPI

Article

Robust Optimization Approach Using Scenario Concepts for Artillery Firing Scheduling Under Uncertainty

Yong Baek Choi [1], Ho Yeong Yun [1], Jang yeop Kim [2], Suk Ho Jin [3] and Kyung Sup Kim [1,*]

[1] Department of Industrial Engineering, Yonsei University 50 Yonsei-ro, Seodaemun-gu, Seoul 03722, Korea
[2] Institute of Defense Acquisition Program, Kwangwoon University, Seoul 01897, Korea
[3] Division of Business Administration, Cheongju University, 298, Daeseong-ro, Cheongwon-gu, Cheongju-si, Chungcheongbuk-do 28503, Korea
* Correspondence: kyungkim@yonsei.ac.kr; Tel.: +82-2-2123-4012

Received: 05 June 2019; Accepted: 12 July 2019; Published: 14 July 2019

Abstract: Real wars involve a considerable number of uncertainties when determining firing scheduling. This study proposes a robust optimization model that considers uncertainties in wars. In this model, parameters that are affected by enemy's behavior and will, i.e., threats from enemy targets and threat time from enemy targets, are assumed as uncertain parameters. The robust optimization model considering these parameters is an intractable model with semi-infinite constraints. Thus, this study proposes an approach to obtain a solution by reformulating this model into a tractable problem; the approach involves developing a robust optimization model using the scenario concept and finding a solution in that model. Here, the combinations that express uncertain parameters are assumed by scenarios. This approach divides problems into master and subproblems to find a robust solution. A genetic algorithm is utilized in the master problem to overcome the complexity of global searches, thereby obtaining a solution within a reasonable time. In the subproblem, the worst scenarios for any solution are searched to find the robust solution even in cases where all scenarios have been expressed. Numerical experiments are conducted to compare robust and nominal solutions for various uncertainty levels to verify the superiority of the robust solution.

Keywords: robust optimization; artillery firing scheduling; threat time; uncertainty

1. Introduction

Artillery refers to a type of heavy military ranged weapon that hits remote targets by operating the weapon system as a unit. It differs from small arms or tanks in armored forces that are used by infantry to engage in short-range gunfights against enemies. The process of operating artillery is as follows: First, target information such as a target properties, size, and location are acquired; second, decision-making on firing method is followed to ensure efficient and effective hits against the target; and third, artillery units hit the target. The number of casualties owing to artillery bombardment has been the largest in World War I, World War II, and other wars until the development of modern warfare. Thus, many countries have recognized the power and importance of artillery and have made considerable effort to develop weapon system technologies, artillery operation tactics, and firing methods. Thus, with the development of state-of-the-art technologies, target information has become more accurate owing to information collection assets, and improved and intelligent ammunition performance has increased the destructive power of artillery and the accuracy of hit ratio. In addition, studies on artillery operation tactics such as shoot-and-scoot tactics [1] and firing methods have been increasingly conducted.

This study investigates different firing methods to determine artillery firing scheduling. Studies on firing methods are largely divided into weapon-target assignment problems (WTAPs) and fire-scheduling problems (FSPs). Existing studies on WTAPs are as follows: Reference [2] introduced a WTAP for the first time and proposed that this problem could be approximated to a linear model even though it was probabilistic and nonlinear. Reference [3] proved that the WTAP is NP-complete and that the computation time increases exponentially as the problem size increases. References [4,5] solved the WTAP using a modified evolution algorithm, and Reference [6] solved the WTAP using a hybrid algorithm. Reference [7] modified the WTAP into integer programming and network flow problems and developed an exact algorithm and a heuristic algorithm. The FSP was first introduced by Reference [8]; they proposed a greedy heuristic to minimize the makespan. Reference [9] assumed that the probability of target destruction decreased with time and proposed a scheduling heuristic to minimize the total threat from surviving targets. Reference [10] developed a robust optimization (RO) model that minimized the total threat exposed to friendly forces by considering enemy threat parameters as uncertain parameters since the intangible combat power of enemy targets could not be predicted or estimated accurately. Reference [11] proposed a two-phase method that could evaluate the total threat of an enemy that is inflicted on friendly forces when the enemy threat level is uncertain. Existing studies on WTAPs and FSPs obtain solutions by assuming that the threat times of all targets that should be hit by artillery units are the same. However, the threat times of targets would likely differ in actual wars. Thus, this study proposes an artillery firing scheduling methodology that minimizes the total enemy threat that is inflicted on friendly forces by considering that the targets' threat times are different.

The artillery firing scheduling problem considered in this study is the single machine earliness/tardiness problem (SMETP). An artillery unit in this study is regarded as a single machine, and the target that should be hit by the artillery unit is considered a job. Here, the time from attacking the target by an artillery unit to the target's destruction is considered the processing time and the threat of a target inflicted on friendly forces before its destruction is defined as a penalty. If the threat time of each target is considered as the release time, then if an artillery unit does not hit their target as soon as the target threat occurs, a tardiness penalty is inflicted. Reference [12] first analyzed the SMETP, and Reference [13] proved the SMETP is NP-hard. Since then, studies on SMETPs have focused on meta-heuristic approaches to solve the complexity of local searches. Reference [14] proposed a hybrid meta-heuristic that combined tabu search concepts and genetic algorithms (GAs) to solve the SMETP. Reference [15] proposed a hybrid heuristic that combined local search heuristics with an evolutionary algorithm. Reference [16] proposed an evolutionary approach based on an imperialist competitive algorithm, and Reference [17] proposed an improved bee algorithm that uses GA operators in the global search stage. The present study also uses GA to overcome the complexity of firing sequence combinations in the global search stage. However, while the aforementioned studies assumed that all parameters were certain in the SMETP, the current study proposes an approach to find a robust solution considering parameter uncertainty.

Parameter uncertainty analyses can be divided into stochastic programming and robust optimization approaches. Stochastic programming optimizes expected performances using prior knowledge on the probability distribution of uncertain data. Accordingly, accurate information about the probability distribution of uncertain data should be known in stochastic programming, and data values should not change until decision-making has been executed [18]. However, it is difficult to acquire accurate information about uncertain data's probability distribution. RO searches for a solution that is immune to uncertain parameters. A robust solution presents the best worst-case solution; this solution is more conservative than a stochastic programming solution. However, RO approaches do not require information about uncertain data's probability distribution. Uncertain parameters are expressed in two sets: scenario and interval datasets using expert opinions or historical data [19]. The RO approach was first introduced by Reference [20] and was extended by References [21,22]. It is now applied in various domains. However, studies on RO approaches have been extremely limited in military operation fields. Only two studies by References [10] and [11] discussed the RO approach. Reference [11] considers target threat level as an uncertain parameter by assuming a problem situation where the enemy unit size cannot be accurately determined. In Reference [11], a deterministic optimization model and an RO model that minimize enemy's firearm threat exposed to the friendly force were developed by distinguishing between when the enemy threat parameter is certain and when it is uncertain. Furthermore, a two-phase approach was developed based on the concept of "cardinality-constrained uncertainty." This approach evaluates the enemy's firearm threat to friendly forces by adjusting the degree of uncertainty when the enemy threat is uncertain. Reference [10] considered target threat level as an uncertain parameter by assuming a problem situation where the quantitative true value of the enemy combat power is unknown. In Reference [10], the combat power of the enemy targets was assumed as the target threat parameter and the model was developed by considering this as an uncertain parameter because it is difficult to determine the quantitative true value of the parameter. References [10] and [11] only consider the single factor (target threat level) out of various factors as an uncertain factor because problem complexity and the robustness issue of solutions are negatively affected as the number of considered uncertain factors increases. However, this study considered two factors at the same time as uncertain parameters, the target threat level and target threat time, but proved that only the uncertain target threat time influenced the solution's robustness.

In this paper, an RO approach using the scenario concept was developed. If the enemy threat and enemy threat time values occurred in a predefined section, the random parameter expression combination was defined as a virtual scenario and a method of finding the solution with the RO concept was developed. This approach is divided into master and subproblems. For the master problem, the optimal robust solution is found and the genetic algorithm is used to solve the complexity of global search. In the subproblem, the worst scenario in certain fire scheduling is found and the objective value is allocated by the fitness function score of the genetic algorithm.

The main contributions of this paper are as follows:

1. This study proposed an artillery firing scheduling model to minimize the total threat of an enemy that is inflicted on friendly forces by considering uncertain parameters such as the enemy's threat level and threat time.
2. The combination that realizes uncertain enemy's threat level and threat time parameters were expressed by discrete scenarios, and the model was developed by which a robust solution could be searched using the RO concept.
3. This study reduced the problem complexity by proving that the worst scenario occurs when the uncertain threat level is at maximum value and the uncertain threat time is at minimum value or at maximum value. That is, only uncertain threat times influenced the solution's robustness, thereby reducing the problem's complexity.
4. Simulation experiments proved that a robust solution was better than a nominal solution.

The rest of this paper is organized as follows: Section 2 presents a deterministic firing scheduling model and an RO model that minimizes the total threat of an enemy and proposes an RO approach using the scenario concept. Section 3 evaluates the performance of the proposed RO approach through numerical experiments. Section 4 analyzes the results of the study and highlights future research directions.

2. Models and Methods

Artillery units perform missions to destroy remote enemy targets by firing ammunition. Enemy targets that should be hit by artillery units are assigned from a higher level of command as a mission or requested from friendly maneuvering forces. These enemy targets have different properties and sizes, and their threat times to friendly forces differ. Furthermore, when artillery units hit enemy targets, the time taken to attack each target until it is destroyed differs. The overall process of artillery units to hit all enemy targets sequentially is defined as a fire operation, and the process of hitting a single target is defined as a fire task. Then, artillery units determine the priority of fire tasks as artillery units cannot perform two or more fire tasks together in a fire operation. The criteria to determine the priority of fire tasks in a fire operation are minimization of the completion time of fire operations, minimization of the average fire task time, and minimization of the enemy's total threat inflicted on friendly forces.

This study models the problem of determining the priority of fire tasks, with the aim of minimizing the total threat of enemies inflicted on friendly forces during a fire operation.

The following assumptions are made in this study:

(1) Artillery units can only hit one target at a time and can only hit the next target after completing the current fire operation.
(2) The time taken from attacking an energy target with artillery units to destroying the target is determined by the properties and size of the enemy's target.
(3) The threat level of the target per unit time is constant until artillery units fire the target.
(4) Once the artillery units start hitting a target, the threat level of the target per unit time decreases linearly; the target is destroyed after passing the required time to destroy the target and, then, the threat level (per unit time) becomes zero.
(5) Enemy targets that are not under attack by artillery units may deal damage to friendly forces as much as the threat level per unit time.
(6) Artillery units can start firing from the time when the threat of an enemy target arises.

The notations used in this paper are defined as follows:

Notation	Definition
Decision variable	
x_j	Firing start time of target j
$y_{jj'}$	If target j takes precedence over target j' 1, otherwise 0
x_{jk}	If target j is fired for kth 1, otherwise 0.
t_k	The kth firing time of artillery units
Parameter	
j	Target j, $\mathrm{j} \in \mathrm{N} = \{1, \dots, \mathrm{n}\}$
f_j	Nominal threat level of target j
$\tilde{f}_j$	Uncertain threat level of target j
$\overline{f_j}$	Maximum threat level of target j
$\underline{f_j}$	Minimum threat level of target j
f_k	Threat level of the kth firing target.
a_j	Threat time of target j
$\tilde{a}_j$	Uncertain threat time of target j
$\overline{a_j}$	Max threat time of target j
$\underline{a_j}$	Min threat time of target j
a_k	Threat time of the kth firing target.
p_j	Amount of time required to destroy target j
r_j	Threat removal rate of target j per unit time $r_j = f_j/p_j$
M	Big positive value

2.1. Deterministic Optimization Modeling

Suppose that there are $\mathrm{j} \in \mathrm{N} = \{1, \dots, \mathrm{n}\}$ targets to be fired in a fire operation. Although each target has a different threat level per unit time f_j and a different threat time a_j, information about these parameters is certain. Moreover, the time p_j that artillery units have to hit and destroy target j is predefined and these values are also certain. In such a case, the deterministic modeling of the artillery firing scheduling (DFS) problem that minimizes the total threat of an enemy inflicted on friendly forces is as follows:

$$minimize \sum_{j=1}^{n} \left\{ f_j(x_j - a_j) + \frac{{f_j}^2}{2r_j} \right\} \qquad \text{(DFS)} \qquad (1)$$

subject to

$$x_j + \frac{f_j}{r_j} \le x_{j'} + M\left(1 - y_{jj'}\right) \qquad j, j' = 1, 2, \ldots, n;\ \ j' \ne j \tag{2}$$

$$x_{j'} + \frac{f_{j'}}{r_{j'}} \le x_j + My_{jj'} \qquad j, j' = 1, 2, \ldots, n;\ \ j' \ne j \tag{3}$$

$$y_{jj'} = 0\ or\ 1 \qquad j, j' = 1, 2, \ldots, n;\ \ j' \ne j \tag{4}$$

$$x_j \ge a_j \qquad \forall\, j \tag{5}$$

Equation (1) is an objective function that minimizes the total threat an enemy inflicts on friendly forces. $f_j(x_j - a_j)$ is a threat observed from time a_j, when a threat from target j is observed, to time x_j, when firing starts. Further, $\frac{{f_j}^2}{2r_j}$ is a threat observed from when firing starts to when target j is destroyed. Based on these assumptions, once the artillery units start firing at target j, the threat level of target j per time f_j decreases linearly and the threat from target j to friendly forces for p_j is $\frac{f_j p_j}{2}$, which is equivalent to $\frac{{f_j}^2}{2r_j}$. Equations (2)–(4) are constraints that determine the target firing sequence, which limits the current target firing that is allowed until the time required to destruct the target has elapsed from the previous target firing time. For example, when target j is hit before target *j′*, the condition $y_{jj'} = 1$ is satisfied and time $x_{j\prime}$ to hit target *j′* should be set to after the time required to destruct the target $\frac{f_j}{r_j}(= p_j)$ has elapsed from target *j* firing time x_j. Equation (5) limits the firing by artillery units allowed from time a_j when target *j*′s threat occurs.

2.2. Robust Optimization Modeling

Information regarding enemies during a real war cannot be accurately predicted and accessed. Enemies will strive to deceive opponent forces and to minimize exposure of their own activities. This is because a war can be won if the opponents are at a disadvantage. Thus, it is difficult to accurately predict and evaluate factors such as enemy's threat level and threat time during a war. To address this, this study develops RO modeling to determine firing scheduling and assumes that the enemy's threat level and threat time are uncertain discrete parameters. Further, it is assumed that $\tilde{f}$ belongs to an uncertainty set UF and that $\tilde{a}$ belongs to an uncertainty set UA. Here, an enemy's threat level and threat time are independent values and their probability distribution cannot be determined. That is, f_j and a_j in Equations (1)–(3),(5) are uncertain values.

Generally, it is assumed that data uncertainty in the following nominal linear optimization problem lies only in the elements of matrix A:

$$\begin{gathered}\text{minimize } \ \mathrm{cx}\\ \text{subject to } \ \mathrm{Ax} \le \mathrm{b}\\ \mathrm{l} \le \mathrm{x} \le \mathrm{u}.\end{gathered}$$

However, even if parameter c is uncertain in the objective function, the problem can be alleviated by minimizing y in the objective equation by using a new variable y and including constraint $\sum_{j=1}^{n}\left\{\tilde{f}_j(x_j - \tilde{a}_j) + \frac{\tilde{f}_j^{\,2}}{2r_j}\right\} \le y$ to $\mathrm{Ax} \le \mathrm{b}$ [22].

The robust fire scheduling (RFS) modeling that determines a firing sequence to minimize total threat inflicted on friendly forces in uncertain situations is then produced as follows:

$$minimize\ \ y \qquad \text{(RFS 1)} \tag{6}$$

subject to

$$\sum_{j=1}^{n}\left\{\tilde{f}_j(x_j - \tilde{a}_j) + \frac{\tilde{f}_j^{\,2}}{2r_j}\right\} \le y \qquad \tilde{f}_j \in UF_j, \qquad \tilde{a}_j \in UA_j \tag{7}$$

$$x_j + \frac{\tilde{f}_j}{r_j} \le x_{j'} + M\left(1 - y_{jj'}\right) \qquad j, j' = 1, 2, \ldots, n;\ \ j' \ne j,\ \tilde{f}_j \in UF_j, \tag{8}$$

$$x_{j'} + \frac{\tilde{f}_{j'}}{r_{j'}} \le x_j + My_{jj'} \qquad j, j' = 1, 2, \ldots, n;\ \ j' \ne j,\ \tilde{f}_j \in UF_j, \tag{9}$$

$$y_{jj'} = 0 \; or \; 1 \qquad j, j' = 1, 2, \ldots, n; \;\; j' \neq j \tag{10}$$

$$x_j \geq \tilde{a}_j \qquad \forall \text{ j } \; \tilde{a}_j \in UA_j \tag{11}$$

2.3. RO Approach Using Scenario Concept

The above RO model is an intractable problem with semi-infinite constraints. Thus, an approach using scenario concept is developed to reformulate the problem into a tractable problem and to identify a robust solution. We assume that the enemy's threat level and threat time are discrete parameters, occurring in sections $[\underline{f_j}, \overline{f_j}]$ and $[\underline{a_j}, \overline{a_j}]$, respectively. Let S be the semi-infinite set of possible realizations of timings of threat and threat levels. A possible scenario $s \in S$ represents a unique set of timings of threat and threat levels. The possible scenario s can then be realized with an unknown probability. Let X be the set of possible target firing schedules. A possible schedule $x \in X$ represents a unique schedule of X. In addition, $function\, f(X, S)$ is defined as a function that evaluates a performance of solution X regarding scenario S. The deterministic optimization model can then be expressed as presented in Equation (12) because it searches for the optimal solution regarding a specific single scenario s.

$$\min_{x \in X} f(x, s) \tag{12}$$

Equation (12) searches for a firing sequence x that minimizes $function\, f(X, S)$ in scenario s in which the enemy's threat level and threat time parameters are unique. The RO model is a method that searches for a robust solution that is immune to all scenarios, S, in which uncertain parameters are realized in a predefined section. In general, there are three measures of robustness: absolute robustness, absolute robust deviation, and relative robustness [19]. Absolute robustness helps minimize the maximum absolute cost in the possible outcome set. Absolute robust deviation helps minimize the maximum regret. Here, regret refers to an absolute difference between the realized result and the optimal solution that corresponds to the realized result. Relative robustness helps minimize the maximum relative deviation of the realized outcome from the corresponding optimal solution. In this firing sequence determination model, absolute robustness is selected as the metric for robustness. This is because uncertain parameters considered in this problem are determined by the opponent's will and enemies strive to ensure that the opponents are at a disadvantage as much as possible in a special situation, such as war. Here, if absolute robustness is selected as the measure of robust, the robust optimal solution searches for the worst scenario from all scenarios sets where uncertain parameters are realized for any solution, and it makes the objective equation optimal (best) among those solutions.

Finding the worst scenario in any solution (scheduling) x can be expressed as follows:

$$f^{WORST}(x) = \max_{s \in S} f(x, s) \tag{13}$$

When all scenarios are realized, the best solution is searched for from all feasible solutions, which can be expressed as follows:

$$\min_{x \in X} f^{WORST}(x) = \min_{x \in X} \max_{s \in S} f(x, s) \tag{14}$$

We search for the best worst-case solutions for all scenarios.

RFS 1 is reformulated to RFS 2 to search for the robust solution with the idea of RO:

$$minimize \;\; y \qquad \text{(RFS 2)} \tag{15}$$

subject to

$$\sum_{j=1}^{n} \sum_{k=1}^{n} \left\{ \tilde{f}_j x_{jk} (t_k - \tilde{a}_j) + \frac{\tilde{f}_j^{\,2}}{2r_j} x_{jk} \right\} \leq y \qquad \tilde{f}_j \in UF_j, \qquad \tilde{a}_j \in UA_j \tag{16}$$

$$t_1 = \sum_{i}^{n} \tilde{a}_j x_{j1} \qquad \tilde{a}_j \in UA_j \tag{17}$$

$$t_k = \max\left[\left(t_{k-1} + \sum_{j=1}^{n} x_{jk-1}\frac{\tilde{f}_j}{r_j}\right), \sum_{j}^{n} \tilde{a}_j x_{jk}\right] \qquad k \geq 2,\ \tilde{f}_j \in UF_j,\ \tilde{a}_j \in UA_j \tag{18}$$

$$\sum_{j=1}^{n} x_{jk} = 1 \qquad \forall\ \mathrm{k} \tag{19}$$

$$\sum_{k=1}^{n} x_{jk} = 1 \qquad \forall\ \mathrm{j} \tag{20}$$

$$x_{jk} = 0\ or\ 1 \qquad \forall\ \mathrm{j,k} \tag{21}$$

Equation (17) refers to the first fire task time of artillery units when a firing sequence is provided, which is the time at which the first firing target emerges. Equation (18) is the kth fire task time of artillery units when a firing sequence is given. The possible kth fire task time starts when the (k − 1)th fire task is complete and the kth firing target emerges.

In RFS 2, solution x refers to a firing sequence, which can be expressed as follows:

$$\text{matrix}\ x = \{x_{jk} = 1\ or\ 0, j, k = 1, 2, \ldots, n \mid If\ target\ j\ has\ the\ kth\ firing, x_{jk} = 1, otherwise\ x_{jk} = 0\}$$

A set of feasible solutions, X, can be expressed as follows according to Equations (19)–(21):

$$\mathrm{X} = \left\{x \mid \sum_{j=1}^{n} x_{jk} = 1, \forall\ \mathrm{k};\ \sum_{k=1}^{n} x_{jk} = 1, \forall\ \mathrm{j};\ x_{jk} = 0\ or\ 1, \forall\ \mathrm{j,k}\right\}$$

In RFS 2, the objective equation and constraint of $f^{WORST}(x)$, which searches for the worst scenario (WS) for a solution x, are as follows:

$$f^{WORST}(x) = \max_{s \in S} f(x, s) = maximize \sum_{k=1}^{n}\left\{\tilde{f}_k(t_k - \tilde{a}_k) + \frac{\tilde{f}_k^{\,2}}{2r_k}\right\} \qquad \text{WS 1} \tag{22}$$

subject to

$$t_1 = \tilde{a}_1 \tag{23}$$

$$t_k = \max\left[\left(t_{k-1} + \frac{\tilde{f}_{k-1}}{r_{k-1}}\right), \tilde{a}_k\right] \qquad k \geq 2,\ \tilde{f}_k \in UF_k,\ \tilde{a}_k \in UA_k \tag{24}$$

WS 1 has uncertain parameters in the objective equation and a constraint, and it is still an intractable problem.

Theorem 1. *For any solution $x \in X$, the worst scenario occurs only when $\tilde{f}_j$ is at maximum value in* ∀j *and when $\tilde{a}_j$ is at minimum value or maximum value.*

WS 2, which is equivalent to the WS 1 model, can be reformulated as follows based on Theorem 1.

$$maximize \sum_{\mathrm{k}=1}^{n}\left\{\overline{f_k}\,(t_k - a_k) + \frac{\overline{f_k}^{\,2}}{2r_k}\right\} \qquad \text{WS 2} \tag{25}$$

subject to

$$t_1 = \overline{a_1} \tag{26}$$

$$t_k = \max\left[\left(t_{k-1} + \frac{\overline{f_{k-1}}}{r_{k-1}}\right), a_k\right] \qquad k \geq 2 \tag{27}$$

$$a_k \in [\underline{a_k}, \overline{a_k}\,] \qquad k \geq 2 \tag{28}$$

WS 2 has finite constraints. Now, problem $\min_{x \in X}\max_{s \in S} f(x, s)$ is divided into master and subproblems, as shown in Figure 1, to find the solution.

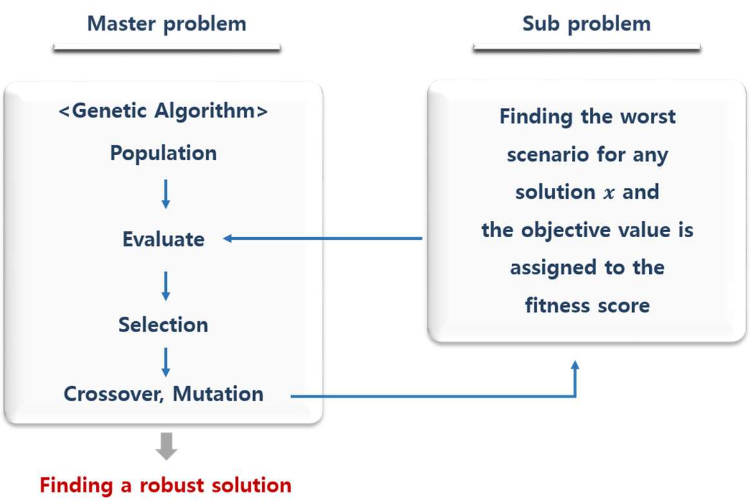

Figure 1. Robust optimization (RO) approach using scenario concept.

"Master problem" refers to finding the robust optimal solution using GA, and "subproblem" refers to finding the worst scenario for any solution x. Model WS 2 helps search for the worst scenario, and the objective value is assigned to the individual fitness function score in the master problem. GA was first invented by Reference [23]. GA is a probabilistic and population-based algorithm, which was inspired by the field of genetics. Generally, GA is applied to problems in which the search space is large for discrete optimization problems [24,25]. This study employed GA to overcome the complexity of solution combination and realistic situations to determine artillery firing schedules within a brief period of time during wars, as suggested by Reference [17]. In this study, the representation method was encoded to a vector (permutation σ) that contained the target information. For example, permutation (3, 1, 4, 2) means that targets 3, 1, 4, and 2 are determined as the first, second, third, and fourth fire tasks, respectively. The initial solution setup in this model is created by the determination criterion that r_j is decreased [11]. In this selection, a stochastic universal selection (SUS) technique was used by referring to a previous study [26], in which comparative experiments were conducted with several selection techniques. Partially matched crossover (PMX), which can avoid chromosome duplication, was applied for the crossover operator, as suggested by Reference [27]. The 2-Opt method, which swaps two different genes selected randomly with probability P_m, was applied for mutation. The termination condition is determined at the maximum CPU time by considering the realistic situation that artillery firing scheduling should be completed within this shortest possible time, and the available time of operation preparation is uncertain.

3. Numerical Experiments and Results

3.1. Experimental Design

In this section, we report numerical experiments to verify the effectiveness of the proposed algorithm. The experiment method is as follows: Robust solution (RS) and nominal solution (NS) are determined, and uncertain threat and threat time parameters of the enemy are created randomly to compare the performance of the two solutions. The Monte Carlo simulation method was used to compare the two solutions, and the objective function values of the two solutions are compared by sampling uncertain data in the predefined section [28]. The experiment employed IBM ILOG CPLEX Optimization software package and C#, and the computer used in the experiment was equipped with an Intel(R) Core i5-6600 processor and 3.3 GHz CPU with 8 GB of RAM.

The parameters in the experiment were generated randomly, as was done in References [9,11]. Target's threat level, f_j, and time to destroy a target, p_j, are generated randomly in sections (3,10) and (1,5), respectively. Target's threat time was generated randomly in section (0, $\sum_{j=1}^{n} p_j$) when considering real wars. Experiments were conducted with four cases with different number of targets:

n = 5, 10, 15, and 20. Uncertainty level θ was selected from set {0.2, 0.4, 0.6, 0.8}. Each experiment had 15 cases with different random seeds. The numerical experiments were conducted with 240 $(4 \times 4 \times 15)$ examples.

3.2. Preliminary Experiment and Results

The method proposed in this paper is an approach based on GA. GA is a probability-based algorithm in which probability parameter values used in the operator influence the algorithm performance. Thus, population size and probability of crossover (P_C) and probability of mutation (P_m), which were parameter factors of the operator used in the algorithm, were determined in the preliminary experiment [29].

Table 1. Experimental factors and their levels.

Factors	Levels	Values
Population Size	3	$n,\ 2n,\ 4n$
Probability of Crossover (P_C)	3	0.1, 0.5, 0.9
Probability of Mutation (P_m)	3	0.01, 0.05, 0.1

The preliminary experiment was performed with the deterministic optimization model, and parameter combinations that exhibited the best performance were determined after configuring 27 (3 × 3 × 3) parameter combinations in Table 1. From target number $n = 15$, three examples were randomly generated and iterative experiments for each were conducted 30 times. If the solution was not updated for more than 1000 times, the experiment was set to terminate.

Table 2. Parameter verification test results of genetic algorithm.

Case	Best	Worst	ST.DEV	Computation Times (Sec)
1	873	873	0	00:02.202
2	757.5	757.5	0	00:02.052
3	585	585	0	00:01.967

The parameter combinations that showed the best result in the preliminary experiment were $population\ size = 2n$, $P_c = 0.9$, and $P_m = 0.1$. The iterative experiments using the parameter combinations were conducted, and the results showed that all combinations reached the same solution within several seconds as summarized in Table 2.

3.3. Simulation Experiment and Results

The NS is determined from the DFS model, assuming that all parameters are certain, and the robust solution (RS) is determined from the RO model, assuming that enemy's threat level and threat time parameters are uncertain; NS and RS are compared in the experiment. Experiments were conducted considering four different number of targets: n = 5, 10, 15, and 20. Here, the uncertain threat, $\tilde{f}_j$, of enemies belongs to the uncertain set $UF_j = \{\tilde{f}_j : f_j - f_j\theta_j \le \tilde{f}_j \le f_j + f_j\theta_j\}$, in which θ_j refers to a uncertainty level selected from {0.2, 0.4, 0.6, 0.8}. The uncertain threat time, $\tilde{a}_j$, of enemies belongs to the uncertain set $UA_j = \{\tilde{a}_j : a_j - a_j\theta_j \le \tilde{a}_j \le a_j + a_j\theta_j\}$, which is selected from {0.2, 0.4, 0.6, 0.8}. The Monte Carlo simulation was conducted to compare the NS and RS; 15 scenarios were created, and 10,000 sets of uncertain data were sampled for each scenario to compare the objective value. The uncertain threat $\tilde{f}_j$ of enemies is randomly selected from section $(f_j - f_j\theta_j, f_j + f_j\theta_j)$, and the uncertain threat occurrence time $\tilde{a}_j$ of enemies is randomly selected from $(a_j - a_j\theta_j, a_j + a_j\theta_j)$.

Table 3 presents the computation time required to search for the optimal robust solution in the CPLEX software. As presented in Table 3, as the number of targets (n) increases, the corresponding computation time increases exponentially. In particular, when the number of targets (n) is eight or larger, the solution cannot be found within three hours by using the CPLEX software. However, firing scheduling should be determined within three hours in a real-world operation. Thus, the optimization method could not be used when the number of targets (n) was eight or larger, and it was verified that the methodology proposed in this study had to be used.

Table 3. CPU times of RO approach using scenario concept with IBM ILOG CPLEX Optimization software package.

Size (n)	5	6	7	8
Computation time (s)	3.25	16.63	317.92	18330.58

Table 4 presents the results of 10,000 simulations for each target number (n) and an uncertainty level (θ) in 15 cases. In Table 4, N refers to the number of better objective values between RS and NS among 150,000 simulation results. For example, when n = 5 and θ = 0.2, the simulation execution result for 150,000 iterations exhibited that 78,556 iterations yielded better results using RS and that 71,444 iterations yielded better results using NS. As shown in Figure 2, RS had a higher number of better results than that of NS for all target sizes.

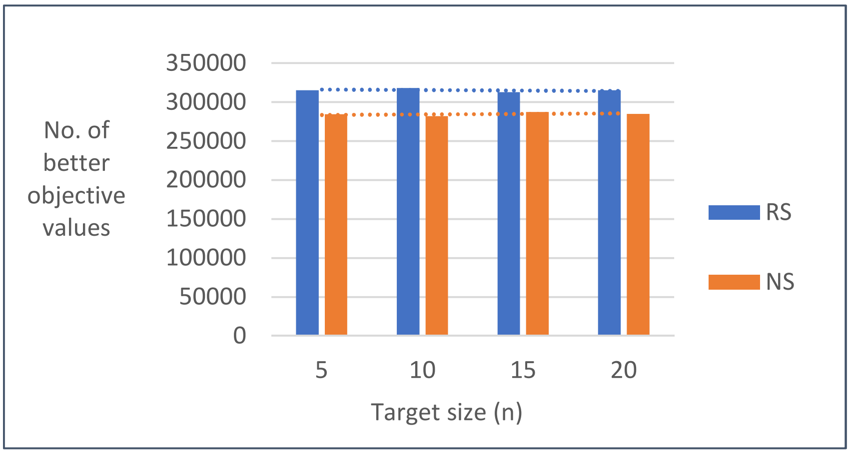

Figure 2. Number of better objective values between robust solution (RS) and nominal solution (NS) among 600,000 simulation results.

Also, the average values of the objective values obtained using RS were smaller than those of NS. The minimum value of the objective value obtained using RS was smaller than that of NS overall, although NS had smaller minimum values than RS in some cases. For example, when n = 10 and θ = 0.4, the minimum value of the objective value using RS was 129.8, whereas that using NS was 116.2.

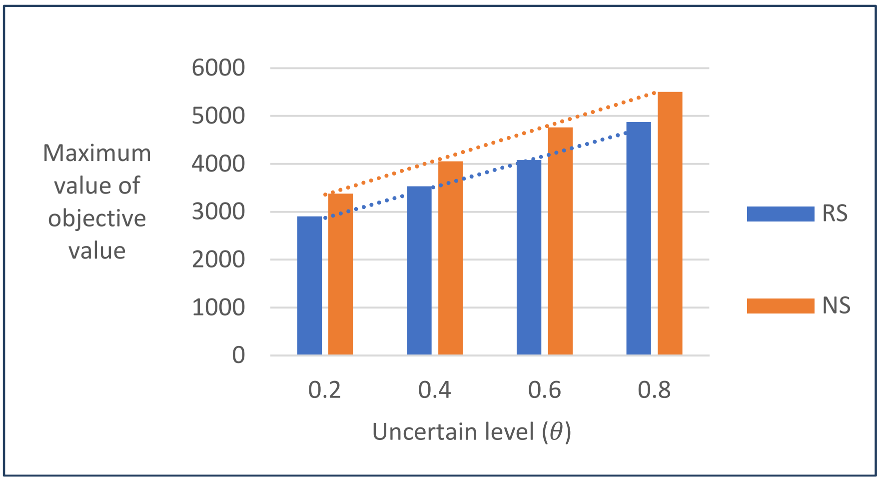

Figure 3. The maximum value of the objective value among 150,000 simulation results for each given uncertain level value (n = 15).

The maximum value of the objective value exhibited that RS was better than NS in all problem sizes and uncertainty levels. That is, the maximum value of the objective value using NS was always larger than that using RS. Here, the objective value means the total threat of the enemy inflicted on friendly forces. The smaller the objective value, the better. As shown in Figure 3, when n = 15 and θ = 0.2, 0.4, 0.6, 0.8, the maximum values of the objective value using RS were 5808.1, 7061.3, 8156.1, and 9755.3, respectively, and those using NS were 6755.3, 8104.5, 9522.5, and 11007.7, respectively. These results prove that the robust solution is the best worst-case solution. This is a very meaningful result considering uncertain war situations. Because enemies are always attempting to create disadvantageous conditions for opponents as much as possible, a solution that considers the worst case is required when determining firing scheduling and this solution must be excellent. The solution searched by the methodology proposed in this study always provides better results in the worst case than the NS.

Table 4. Simulation experimental results to compare objective values at various target numbers (n) and uncertainty levels (θ).

n	θ	Objective Values									
		N		Mean		Min		Max		SD	
		RS	NS	RS	NS	RS	NS	RS	NS	RS	NS
5	0.2	78556	71444	219.4	224.3	14.9	16.5	854.0	944.4	100.4	101.1
	0.4	80509	69491	224.1	232.9	13.7	14.2	1037.2	1101.6	110.3	113.5
	0.6	87927	62073	231.2	243.8	11.0	15.5	1234.8	1286.5	127.6	131.1
	0.8	82380	67620	247.4	252.2	7.9	11.8	1440.2	1481.5	146.4	152.8
10	0.2	79594	70406	1005.4	1017.2	126.1	153.2	2836.8	3224.9	339.2	335.8
	0.4	81412	68588	1048.8	1074.9	129.8	116.2	3317.4	3830.4	392.5	384.7
	0.6	81318	68682	1109.7	1137.1	115.3	125.4	3818.8	4619.8	453.0	450.4
	0.8	77851	72149	1208.8	1215.2	78.0	115.4	4590.6	5483.9	544.5	526.2
15	0.2	79777	70223	2419.8	2440.9	487.1	581.7	5808.1	6755.3	667.4	672.1
	0.4	78634	71366	2551.7	2572.0	335.1	411.5	7061.3	8104.5	782.5	789.4
	0.6	77349	72651	2738.9	2750.1	276.8	412.1	8156.1	9522.5	931.5	933.9
	0.8	76909	73091	2942.0	2961.9	319.5	398.0	9755.3	11007.7	1082.8	1094.7
20	0.2	78184	71816	4522.8	4554.4	1133.6	1428.9	9839.7	10619.7	1098.4	1098.2
	0.4	79791	70209	4828.3	4874.5	1223.0	1178.7	11858.9	12586.1	1318.2	1314.6
	0.6	79531	70469	5201.3	5264.8	1265.3	1277.1	13889.3	14999.9	1597.1	1568.7
	0.8	77533	72467	5698.1	5729.0	958.4	1106.9	15636.2	16937.6	1852.1	1844.3

4. Conclusions and Discussions

This study proposed a method that determined firing scheduling to minimize the total threat from enemies inflicted on friendly forces. Here, the firing sequence was determined by considering parameters such as enemy's threat level, threat time, and time required to destroy the target. However, the enemy's threat level and threat time are enemy-related parameters, for which it is difficult to obtain the quantitative true value. Thus, those parameters should be considered as uncertain parameters. This is because enemies strive to deceive opponents and minimize exposing their actions as much as possible. Accordingly, this study considered enemy threat level and threat time as uncertain parameters and proposed a methodology to find the best worst-case solution. This study assumed the random realization of enemy threat level and threat time parameters as arbitrary war scenarios to identify the most robust solution and developed a method to find this solution using the robust optimization concept.

It is advantageous to determine firing scheduling in war as quickly as possible. This is because more operation preparation time will be available by reducing the time to determine firing scheduling. In this study, the fire scheduling model was divided into master and subproblems to identify the most robust solution. In the master problem, GA was utilized to overcome the complexity of global search and to shorten the solution search time. The commercial optimization program CPLEX was unable to identify the solution within three hours when the target number (n) was eight or larger in the numerical experiment, whereas the methodology proposed in this study was able to identify the solution even if the target number was large.

Fire scheduling is a special circumstance that requires decision making in a relationship with enemies as competitors. In such war scenarios, parameters that affect decision making are uncertain and worse cases are likely to occur. For competitors, it would be advantageous to create the worst situations for opponents. Thus, a robust solution for fire scheduling must be found regardless of uncertain circumstances. The model proposed in this study provides the best solution in the worst situation. The RS performance obtained using the proposed methodology was verified through numerical experiments, which indicated that RS always had better results than those of NS in the worst case, regardless of uncertainty level and problem size.

For future research, a model that determines fire scheduling will be developed to consider realistic limitations such as preparation time for changing targets during artillery firing and uncertainty of the enemy's intangible combat power. Next, studies on air defense weapon scheduling and missile scheduling should also consider uncertainty as a similar field of this study. Thus, a methodology based on the methodology proposed in this study must be developed to solve problems by considering unique operation situations and constraints that should be considered for each field.

Author Contributions: Y.B.C. developed the model and wrote the majority of the manuscript. H.Y.Y . and S.H.J. performed the experiments and wrote the experimental section of the paper. J.Y.K reviewed and edited the manuscript. K.S.K. developed the overall idea and the basic outline of the paper.

Funding: This work was supported by the Ministry of Education of the Republic of Korea and the National Research Foundation of Korea. (NRF-2017S1A5B8060156)

Conflicts of Interest: The authors declare no conflict of interest

Appendix A

Proof of Theorem 1. Given the solution (sequence of shooting), the objective and constraints to find the worst scenario among the shooting sequences are as follows.

$$\text{maximize } \sum_{k=1}^{n}\left\{\tilde{f}_k(t_k-\tilde{a}_k)+\frac{\tilde{f}_k^{\,2}}{2r_k}\right\} \tag{A1}$$

subject to

$$t_1 = a_1 \tag{A2}$$

$$t_k = \max\left[\left(t_{k-1}+\frac{\tilde{f}_{k-1}}{r_{k-1}}\right), \tilde{a}_k\right] \qquad \mathrm{k} \geq 2,\ \ \tilde{f}_k \in U_k,\ \tilde{a}_k \in U_k \tag{A3}$$

In this case, the combination scenario of $\tilde{f}_k$ and $\tilde{a}_k$, which absolutely maximizes the objective equation for all k, is a worst scenario.

The following three cases can be considered depending on the value of k. □

Case 1. *When* $\boldsymbol{k}=1$,

Let F_w *be Equation (31). Then,*

$F_w = \sum_{k=1}^{n}\left\{\tilde{f}_k(t_k-\tilde{a}_k)+\frac{\tilde{f}_k^{\,2}}{2r_k}\right\} = \tilde{f}_1(t_1-\tilde{a}_1)+\frac{\tilde{f}_1^{\,2}}{2r_1}+\tilde{f}_2(t_2-\tilde{a}_2)+\frac{\tilde{f}_2^{\,2}}{2r_2}+\sum_{k=3}^{n}\left\{\tilde{f}_k(t_k-\tilde{a}_k)+\frac{\tilde{f}_k^{\,2}}{2r_k}\right\}$

$t_1 = a_1,\ t_2 = \max\left[\left(t_1+\frac{\tilde{f}_1}{r_1}\right),\ \tilde{a}_2\right]$.

① *If* $t_1 = \tilde{a}_1$ *and* $t_2 = t_1 + \frac{\tilde{f}_1}{r_1}$,

then $F_w = \frac{\tilde{f}_1^{\,2}}{2r_1} + \tilde{f}_2\left(\tilde{a}_1+\frac{\tilde{f}_1}{r_1}-\tilde{a}_2\right)+\frac{\tilde{f}_2^{\,2}}{2r_2}+\sum_{k=3}^{n}\left\{\tilde{f}_k(t_k-\tilde{a}_k)+\frac{\tilde{f}_k^{\,2}}{2r_k}\right\}$.

That is, $t_1+\frac{\tilde{f}_1}{r_1} \geq \tilde{a}_2,\ \tilde{a}_1+\frac{\tilde{f}_1}{r_1}-\tilde{a}_2 \geq 0$.

$\forall k$ $\tilde{f}_k$, $\tilde{a}_k$, r_k, *and* t_k *are nonnegative values, and* $\forall k$ $\tilde{f}_k$, *and* $\tilde{a}_k$ *are independent values.*

Therefore, when $\tilde{f}_1 = \overline{f_1}$, $\tilde{a}_1 = \overline{a_1}$, F_w *is maximized, and this is a worst scenario.*

② *If* $t_1 = \tilde{a}_1$ and $t_2 = \tilde{a}_2$,

then $F_w = \frac{\tilde{f}_1^{\,2}}{2r_1}+\frac{\tilde{f}_2^{\,2}}{2r_2}+\sum_{k=3}^{n}\left\{\tilde{f}_k(t_k-\tilde{a}_k)+\frac{\tilde{f}_k^{\,2}}{2r_k}\right\}$.

Therefore, when $\tilde{f}_1 = \overline{f_1}$, F_w *is maximized and this is a worst scenario.*

Case 2. *When $2 \leq \boldsymbol{k} \leq \boldsymbol{n-1}$,*

$$F_w = \sum_{q=1}^{k-1}\left\{\tilde{f}_q(t_q - \tilde{a}_q) + \frac{\tilde{f}_q^{\,2}}{2r_q}\right\} + \tilde{f}_k(t_k - \tilde{a}_k) + \frac{\tilde{f}_k^{\,2}}{2r_k} + \tilde{f}_{k+1}(t_{k+1} - \tilde{a}_{k+1}) + \frac{\tilde{f}_{k+1}^{\,2}}{2r_{k+1}} + \sum_{q=k+2}^{n}\left\{\tilde{f}_q(t_q - \tilde{a}_q) + \frac{\tilde{f}_q^{\,2}}{2r_q}\right\}$$

$$t_k = \max\left[\left(t_{k-1} + \frac{\tilde{f}_{k-1}}{r_{k-1}}\right), \tilde{a}_k\right].$$

① *If* $t_k = t_{k-1} + \frac{\tilde{f}_{k-1}}{r_{k-1}}$ *and* $t_{k+1} = t_k + \frac{\tilde{f}_k}{r_k}$,

then $F_w = \sum_{q=1}^{k-1}\left\{\tilde{f}_q(t_q - \tilde{a}_q) + \frac{\tilde{f}_q^{\,2}}{2r_q}\right\} + \tilde{f}_k\left(t_{k-1} + \frac{\tilde{f}_{k-1}}{r_{k-1}} - \tilde{a}_k\right) + \frac{\tilde{f}_k^{\,2}}{2r_k} + \tilde{f}_{k+1}\left(t_k + \frac{\tilde{f}_k}{r_k} - \tilde{a}_{k+1}\right) + \frac{\tilde{f}_{k+1}^{\,2}}{2r_{k+1}} + \sum_{q=k+2}^{n}\left\{\tilde{f}_q(t_q - \tilde{a}_q) + \frac{\tilde{f}_q^{\,2}}{2r_q}\right\}.$

That is, $t_{k-1} + \frac{\tilde{f}_{k-1}}{r_{k-1}} \geq \tilde{a}_k$, $t_k + \frac{\tilde{f}_k}{r_k} \geq \tilde{a}_{k+1}$.

Therefore, when $\tilde{f}_k = \overline{f_k}$, $\tilde{a}_k = \underline{a_k}$, F_w *is maximized, and this is a worst scenario.*

② If $t_k = t_{k-1} + \frac{\tilde{f}_{k-1}}{r_{k-1}}$ *and* $t_{k+1} = \tilde{a}_{k+1}$,

then $F_w = \sum_{q=1}^{k-1}\left\{\tilde{f}_q(t_q - \tilde{a}_q) + \frac{\tilde{f}_q^{\,2}}{2r_q}\right\} + \tilde{f}_k\left(t_{k-1} + \frac{\tilde{f}_{k-1}}{r_{k-1}} - \tilde{a}_k\right) + \frac{\tilde{f}_k^{\,2}}{2r_k} + \frac{\tilde{f}_{k+1}^{\,2}}{2r_{k+1}} + \sum_{q=k+2}^{n}\left\{\tilde{f}_q(t_q - \tilde{a}_q) + \frac{\tilde{f}_q^{\,2}}{2r_q}\right\}.$

That is, $t_{k-1} + \frac{\tilde{f}_{k-1}}{r_{k-1}} \geq \tilde{a}_k$.

Therefore, when $\tilde{f}_k = \overline{f_k}$, $\tilde{a}_k = \underline{a_k}$, F_w *is maximized, and this is a worst scenario.*

③ *If* $t_k = \tilde{a}_k$ *and* $t_{k+1} = t_k + \frac{\tilde{f}_k}{r_k} = \tilde{a}_k + \frac{\tilde{f}_k}{r_k}$,

then $F_w = \sum_{q=1}^{k-1}\left\{\tilde{f}_q(t_q - \tilde{a}_q) + \frac{\tilde{f}_q^{\,2}}{2r_q}\right\} + \frac{\tilde{f}_k^{\,2}}{2r_k} + \tilde{f}_{k+1}\left(\tilde{a}_k + \frac{\tilde{f}_k}{r_k} - \tilde{a}_{k+1}\right) + \frac{\tilde{f}_{k+1}^{\,2}}{2r_{k+1}} + \sum_{q=k+2}^{n}\left\{\tilde{f}_q(t_q - \tilde{a}_q) + \frac{\tilde{f}_q^{\,2}}{2r_q}\right\}.$

That is, $\tilde{a}_k + \frac{\tilde{f}_k}{r_k} \geq \tilde{a}_{k+1}$.

Therefore, when $\tilde{f}_k = \overline{f_k}$, $\tilde{a}_k = \overline{a_k}$, $\mathrm{F_w}$ *is maximized, and this is a worst scenario.*

④ *If* $t_k = \tilde{a}_k$ *and* $t_{k+1} = \tilde{a}_{k+1}$,

then $F_w = \sum_{q=1}^{k-1}\left\{\tilde{f}_q(t_q - \tilde{a}_q) + \frac{\tilde{f}_q^{\,2}}{2r_q}\right\} + \frac{\tilde{f}_k^{\,2}}{2r_k} + \frac{\tilde{f}_{k+1}^{\,2}}{2r_{k+1}} + \sum_{q=k+2}^{n}\left\{\tilde{f}_q(t_q - \tilde{a}_q) + \frac{\tilde{f}_q^{\,2}}{2r_q}\right\}.$

Therefore, when $\tilde{f}_k = \overline{f_k}$, F_w *is maximized and this is a worst scenario.*

Case 3. *When $\boldsymbol{k} = \boldsymbol{n}$,*

$$F_w = \sum_{k=1}^{n-1}\left\{\tilde{f}_k(t_k - \tilde{a}_k) + \frac{\tilde{f}_k^{\,2}}{2r_k}\right\} + \tilde{f}_n(t_n - \tilde{a}_n) + \frac{\tilde{f}_n^{\,2}}{2r_n}$$

① *If* $t_n = t_{n-1} + \frac{\tilde{f}_{n-1}}{r_{n-1}}$,

then $F_w = \sum_{k=1}^{n-1}\left\{\tilde{f}_k(t_k - \tilde{a}_k) + \frac{\tilde{f}_k^{\,2}}{2r_k}\right\} + \tilde{f}_n\left(t_{n-1} + \frac{\tilde{f}_{n-1}}{r_{n-1}} - \tilde{a}_n\right) + \frac{\tilde{f}_n^{\,2}}{2r_n}$ and $t_{n-1} + \frac{\tilde{f}_{n-1}}{r_{n-1}} \geq \tilde{a}_n$.

Therefore, when $\tilde{f}_n = \overline{f_n}$, $\tilde{a}_n = \overline{a_n}$, F_w *is maximized, and this is a worst scenario.*

② *If* $t_n = \tilde{a}_n$,

then $F_w = \sum_{k=1}^{n-1}\left\{\tilde{f}_k(t_k - \tilde{a}_k) + \frac{\tilde{f}_k^{\,2}}{2r_k}\right\} + \frac{\tilde{f}_n^{\,2}}{2r_n}$

Therefore, when $\tilde{f}_n = \overline{f_n}$, F_w *is maximized and this is a worst scenario.*

In conclusion, for solution $x \in X$, *the worst scenario is a when* $\tilde{f}_k$ is $\overline{f_k}$, *and* $\tilde{a}_k$ *occurs when* $\underline{a_k}$ *or* $\overline{a_k}$.

References

1. Shim, Y.; Atkinson, M.P. Analysis of artillery shoot-and-scoot tactics. *Nav. Res. Logist. (NRL)* **2018**, *65*, 242–274.
2. Manne, A.S. A target-assignment problem. *Oper. Res.* **1958,** *6*, 346–351.
3. Lloyd, S.P.; Witsenhausen, H.S. Weapons allocation is NP-complete. In Proceedings of the 1986 Summer Computer Simulation Conference, Reno, NV, USA, 28–30.July 1986; pp. 1054–1058
4. Lee, Z.J.; Su, S.F.; Lee, C.Y. A genetic algorithm with domain knowledge for weapon-target assignment problems. *J. Chin. Inst. Eng.* **2002**, *25*, 287–295.
5. Erdem, E.; Ozdemirel, N. An evolutionary approach for the target allocation problem. *J. Oper. Res. Soc.* **2003**, *54*, 958–969.
6. Lee, Z.J.; Lee, W.L. A hybrid search algorithm of ant colony optimization and genetic algorithm applied to weapon-target assignment problems. In *Lecture Notes in Computer Science, Proceedings of the International Conference on Intelligent Data Engineering and Automated Learning, Hong Kong, China, 21–23 March, 2003*; Springer: Berlin/Heidelberg, Germany, 2003; pp. 278–285.
7. Ahuja, R.K.; Kumar, A.; Jha, K.C.; Orlin, J.B. Exact and heuristic algorithms for the weapon-target assignment problem. *Oper. Res.* **2007**, *55*, 1136–1146.
8. Kwon, O.; Lee, K.; Park, S. Targeting and scheduling problem for field artillery. *Comput. Ind. Eng.* **1997**, *33*, 693–696.
9. Cha, Y.-H.; Kim, Y.-D. Fire scheduling for planned artillery attack operations under time-dependent destruction probabilities. *Omega* **2010**, *38*, 383–392.
10. Choi, Y.B.; Jin, S.H.; Kim, K.S. Deterministic and Robust Optimization Approach for Single Artillery Unit Fire Scheduling Problem. *Appl. Sci.* **2017**, *7*, 1038.
11. Choi, Y.B.; Jin, S.H.; Kim, K.S.; Do Chung, B. A robust optimization approach for an artillery fire-scheduling problem under uncertain threat. *Comput. Ind. Eng.* **2018**, *125*, 23–32.
12. Panwalkar, S.S.; Smith, M.L.; Seidmann, A. Common due date assignment to minimize total penalty for the one machine scheduling problem. *Oper. Res.* **1982**, *30*, 391–399.
13. Hall, N.G.; Kubiak, W.; Sethi, S.P. Earliness–tardiness scheduling problems, II: Deviation of completion times about a restrictive common due date. *Oper. Res.* **1991**, *39*, 847–856.
14. Hino, C.M.; Ronconi, D.P.; Mendes, A.B. Minimizing earliness and tardiness penalties in a single-machine problem with a common due date. *Eur. J. Oper. Res.* **2005**, *160*, 190–201.
15. M'Hallah, R. Minimizing total earliness and tardiness on a single machine using a hybrid heuristic. *Comput. Oper. Res.* **2007**, *34*, 3126–3142.
16. Yousefi, M.; Yusuff, R.M. Minimising earliness and tardiness penalties in single machine scheduling against common due date using imperialist competitive algorithm. *Int. J. Prod. Res.* **2013**, *51*, 4797–4804.
17. Yuce, B.; Fruggiero, F.; Packianather, M.S.; Pham, D.T.; Mastrocinque, E.; Lambiase, A.; Fera, M. Hybrid Genetic Bees Algorithm applied to single machine scheduling with earliness and tardiness penalties. *Comput. Ind. Eng.* **2017**, *113*, 842–858.
18. Birge, J.R.; Louveaux, F. *Introduction to Stochastic Programming*; Springer Science & Business Media: Berlin, Germany, 2011.
19. Kouvelis, P.; Yu, G. *Robust Discrete Optimization and Its Applications*; Springer Science & Business Media: Berlin, Germany, 2013; Volume 14.
20. Soyster, A.L. Convex programming with set-inclusive constraints and applications to inexact linear programming. *Oper. Res.* **1973**, *21*, 1154–1157.
21. Ben-Tal, A.; Nemirovski, A. Robust solutions of uncertain linear programs. *Oper. Res. Lett.* **1999**, *25*, 1–13.
22. Bertsimas, D.; Sim, M. The price of robustness. *Oper. Res.* **2004**, *52*, 35–53.
23. Holland, J.H. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligenc*; MIT Press: Cambridge, MA, USA, 1992.
24. Back, T. *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithm*; Oxford University Press: Oxford, UK, 1996.
25. Gen, M.; Cheng, R. *Genetic Algorithms and Engineering Optimization*; John Wiley & Sons: Hoboken, NJ, USA, 2000; Volume 7.
26. Clark, J.; Goldberg, D.; Deb, K. Genetic algorithms, noise, and the sizing of populatons. *Complex Syst.* **1992**, *6*, 333–362.
27. Larranaga, P.; Kuijpers, C.M.H.; Murga, R.H.; Inza, I.; Dizdarevic, S. Genetic algorithms for the travelling salesman problem: A review of representations and operators. *Artif. Intell. Rev.* **1999**, *13*, 129–170.

28. Gorissen, B.L.; Yanıkoğlu, İ.; den Hertog, D. A practical guide to robust optimization. *Omega* **2015**, *53*, 124–137.
29. Pongcharoen, P.; Chainate, W.; Thapatsuwan, P. Exploration of genetic parameters and operators through travelling salesman problem. *ScienceAsia* **2007**, *33*, 215–222.