Iranian Journal of Numerical Analysis and Optimization
Vol. 12, No. 3 (Special Issue), 2022, pp 542–566
DOI:10.22067/ijnao.2022.71302.1046
https://ijnao.um.ac.ir/

Research Article

Deception in multi-attacker security game with nonfuzzy and fuzzy payoffs

S. Esmaeeli, H. Hassanpour*, and H. Bigdeli

Abstract

There is significant interest in studying security games for defense optimization and reducing the effects of attacks on various security systems involving vital infrastructures, financial systems, security, and urban safeguarding centers. Game theory can be used as a mathematical tool to maximize the efficiency of limited security resources. In a game, players are smart, and it is natural for each player (defender or attacker) to try to deceive the opponent using various strategies in order to increase his payoff. Defenders can use deception as an effective means of enhancing security protection by giving incorrect information, hiding specific security resources, or using fake resources. However, despite the importance of deception in security issues, there is no considerable research on this field, and most of the works focus on deception in cyber environments. In this paper, a mixed-integer linear programming problem is proposed to allocate forces efficiently in a security game with multiple attackers using game theory analysis. The important subjects of information are their credibility and reliability. Especially when the defender uses deceptive defense forces, there are more ambiguity and uncertainty. Security game with Z-number payoffs is considered to apply both ambiguities in the payoffs and the reliability of earning these payoffs. Finally, the proposed method is illustrated by some numerical examples.

* Corresponding author

Received 5 July 2021; revised 15 April 2022; accepted 1 May 2022

S. Esmaeeli
Department of Mathematics, University of Birjand, Birjand, I.R. of Iran. e-mail: s.esmaeely@birjand.ac.ir

H. Hassanpour
Department of Mathematics, University of Birjand, Birjand, I.R. of Iran. e-mail: hhassanpour@birjand.ac.ir

H. Bigdeli
Researcher, Institute for the Study of War, Command and Staff University, Tehran, I.R. of Iran army. e-mail: hamidbigdeli92@gmail.com

AMS subject classifications (2020): 91Axx; 90C70; 90C29.

Keywords: Security game; Deceptive resource; Mixed-integer programming; Fuzzy theory; Z-number.

1 Introduction

Game theory has many applications in real-world problems, in many fields such as economics, military, politics, and so on (e.g., see [6, 2, 37]). In real-world game problems, we may encounter various types of uncertainty or inaccuracy in information (payoffs). Many researchers have studied game theory with different types of information ambiguity [3, 38, 39]. Seikh, Dutta, and Li [36] studied matrix games with rough interval payoffs and investigated two different solution methodologies to solve such a game. Karmakar, Seikh, and Castillo [24] developed a matrix game in a type-2 intuitionistic fuzzy environment. Bigdeli, Hassanpour, and Tayyebi [5] introduced two multiobjective linear programming problems to compute the optimistic and pessimistic values of fuzzy multiobjective games and their corresponding Pareto optimal strategies for each of the players by considering the concept of nearest interval approximation.

Security in maintaining military order and defense has always been a significant concern in human societies. In recent years, economic and political security has also become important. Limitations of resources such as money, personnel, and equipment have made it necessary to optimize the allocation of security resources. Security games have been successfully applied to solve many real-world security problems [1, 18, 26, 41]. They are also effective tools for arguing about the allocation of limited security resources and patrolling problems [1, 13, 25].

There has been a great deal of interest in research on game theory for security in airports, ports, transportation, and other infrastructures. Over the past decade, game theory has been used in various military sectors, computer network security applications, anti-ballistic missile defense systems, wildlife protection, and so on. Lye and Wing [27] proposed a game-theoretic method for analyzing security in computer networks. Brown et al. [8] described a new two-sided optimization model for planning the pre-positioning of defensive missile interceptors to counter an attacking threat. Conitzer and Sandholm [11] proposed a method to perform optimal random strategies in security games. Tarjom, Clempner, and Poznyak [42] used a method to calculate the Nash equilibrium in the case of one defender and several attackers. With respect to wildlife protection, Fang et al. [18] used repetitive interactions between rangers and hunters in protected areas to plan a patrol strategy that allowed rangers to collect hunting signals over time. Bigdeli, Hassanpour, and Tayyebi [7] proposed a model for solving a multiobjective security game with fuzzy payoffs and its application in a metro security system.

Most security games use the Stackelberg game because security forces typically commit to specific security policies to arrange their forces. Thus attackers are empowered to model their attacks under surveillance to take advantage of any potential weakness of the defender. Furthermore, the main assumption in Stackelberg security games is that limited security resources must be deployed strategically, considering differences in priorities of targets requiring security coverage and the responses of the adversaries to the security position (e.g., see [7, 4, 42, 43, 44]).

Previous studies assume the perfect surveillance of the defender's strategies despite the deceptions, while it is natural that if one of the players can deceive another, he will not hesitate. Defender's deceptive actions can affect the attacker's view of the defender's strategy, thus on the attacker's best response, and vice versa. Despite being relatively ignored in academia, in the military, deception is as old as war or politics. There are many examples of military deception in history. The story of the Trojan horse in Ancient Greece is perhaps the most famous ancient military deception. Also, in ancient China, many generals used to resort to deception ruses [30].

As a more recent example, World War II armies deceived their enemies by designing and building air tanks and wooden artillery. Thus, enemy forces would overestimate the enemy's defense capabilities and waste their ammunition or endanger their equipment. In another example, on a Japanese island in the Pacific Ocean, wicker planes deceived many American pilots. They spent a significant portion of their ammunition attacking unreal models by thinking only that the planes were real. For further study, in [12, 21, 19], there are numerous examples of deception in the First and Second World Wars.

Although research on deception in security games has increased in recent years, there is no noteworthy research in this field. Moreover, most authors focus on deception in cyber environments (e.g., see [14, 20, 28, 32, 40, 47]). Recently, deceptive methods have also been used to defend information systems. Cohen and Koike [10] provided a comprehensive discussion of deception to increase the security of information systems and concluded that "deception" is a positive factor for the defender and a negative factor for the attacker. In the security-military sector, Yin et al. [45] examined how fake resources and concealing the real resources of the defender might affect the attacker's beliefs and thus affect his best response. The authors [17] proposed a mathematical model to solve a security game in a fuzzy environment, in which the defender uses unrealistic resources when confronted with only one attacker. In the real world, it is important for players to have complete confidence in their information. Especially in situations where the defender uses deceptive defense forces, there are more ambiguity and uncertainty. Therefore, not only players can not accurately estimate their payoffs, but also they cannot be 100 % sure of these approximated estimates. Therefore, using fuzzy set theory in such games is necessary. There is no research on multi-attacker

security games with deceptive resources and fuzzy payoffs based on the best knowledge of the authors.

The security game has also been studied in [7, 17, 41]. In the multi-attacker security game solved in [41], the players' payoffs are considered to be crisp numbers. In [17], a security game problem in the fuzzy environment having only one attacker was solved. The fuzzy order used in [17] increases the number of constraints. In addition, the proposed method cannot be generalized to the case of multiple attackers. In [7], a multi-attacker security game with triangular fuzzy payoffs was solved, in which the authors considered the pessimistic situation and obtained an efficient solution for a cautious defender. In this paper, a security game problem with different types of attackers and different types of defense forces, such as real, secret, and fake, in a fuzzy environment is considered.

The remainder of the paper is organized as follows: In Section 2, some required concepts of fuzzy set theory are given. In Section 3, Stackelberg games are introduced, and the concept of efficient strategy in these games with multi-follower is defined. A security game with different types of attackers is introduced in Section 4. In Section 5, a security game problem is considered in which the defender's strategies can include deceptive protection covers, and a multiobjective optimization problem is proposed to obtain an efficient strategy for the defender. In Section 6, the players' payoffs are considered as Z-numbers, and a multiobjective optimization problem is proposed to get the efficient coverage of the defender when he uses deceptive resources. In Section 7, four numerical examples are provided to illustrate the proposed method. Finally, the conclusion is made in Section 8.

2 Basic concepts and definitions

In this Section, some concepts that are used in the paper are given.

Definition 1. A fuzzy set $\tilde{A}$ defined on a universe X is given as $\tilde{A} = \{(x, \mu_{\tilde{A}}(x))|x \in X\}$, where $\mu_{\tilde{A}} : X \longrightarrow [0,1]$ is the continuous membership function of $\tilde{A}$. The membership value $\mu_{\tilde{A}}(x)$ describes the degree of belongingness of $x \in X$ in $\tilde{A}$.

The support of a fuzzy set $\tilde{A}$ on X is defined by

$$supp(\tilde{A}) = \{x \in X \mid \mu_{\tilde{A}}(x) > 0\}.$$

A fuzzy number is a fuzzy set $\tilde{A}$ on the real line $\mathbb{R}$ with a continuous membership function $\mu_{\tilde{A}}$ that can be described as follows [15, 22]:

$$\mu_{\tilde{A}}(x) = \begin{cases} 0 & \text{for all } x \in (-\infty, a_1], \\ f_A(x) & \text{for all } x \in [a_1, a_2], \\ 1 & \text{for all } x \in [a_2, a_3], \\ g_A(x) & \text{for all } x \in [a_3, a_4], \\ 0 & \text{for all } x \in [a_4, \infty), \end{cases} \tag{1}$$

where f_A represents a continuous and monotonically increasing function on $[a_1, a_2]$ and g_A is a continuous and monotonically decreasing function on $[a_3, a_4]$.

The α-level set of a fuzzy number $\tilde{A}$ is defined by the ordinary set $\tilde{A}_\alpha = \{x \in X |\, \mu_{\tilde{A}}(x) \geq \alpha\}$ for $\alpha \in (0, 1]$, and for $\alpha = 0$, $\tilde{A}_\alpha = cl\{x \in X | \mu_{\tilde{A}}(x) > 0\}$ where cl means closure of the set [9]. For $\alpha \in (0, 1]$, the α-level set of a fuzzy number is a closed and bounded interval, denoted as $\tilde{A}_\alpha = [f_A^{-1}(\alpha), g_A^{-1}(\alpha)]$, where $f_A^{-1}(\alpha) = inf\{x |\, \mu_{\tilde{A}}(x) \geq \alpha\}$ and $g_A^{-1}(\alpha) = sup\{x |\, \mu_{\tilde{A}}(x) \geq \alpha\}$.

Definition 2. [22] The expected interval of a fuzzy number $\tilde{A}$, denoted by $EI(\tilde{A})$, is defined as follows:

$$EI(\tilde{A}) = [\int_0^1 f_A^{-1}(\alpha) d\alpha, \int_0^1 g_A^{-1}(\alpha) d\alpha].$$

A fuzzy number $\tilde{A}$ on $\mathbb{R}$ is said to be a triangular fuzzy number if its membership function $\mu_{\tilde{A}} : \mathbb{R} \longrightarrow [0, 1]$ is

$$\mu_{\tilde{A}}(x) = \begin{cases} (x - a^1)/(a^2 - a^1), & a^1 \leq x \leq a^2, \\ (a^3 - x)/(a^3 - a^2), & a^2 \leq x \leq a^3, \\ 0, & otherwise, \end{cases} \tag{2}$$

where a^1 and a^3 represent the beginning and end points of the support of $\tilde{A}$, respectively, and a^2 is the median value (center).

The triangular fuzzy number defined above, is denoted by $\tilde{A} = (a^1, a^2, a^3)$. The addition of two triangular fuzzy numbers $\tilde{A} = (a^1, a^2, a^3)$ and $\tilde{B} = (b^1, b^2, b^3)$, and the multiplication of the triangular fuzzy number $\tilde{A}$ by $k \in \mathbb{R}$ using the extension principle of Zadeh [34] are obtained as follows:

$$\tilde{A} + \tilde{B} = (a^1, a^2, a^3) + (b^1, b^2, b^3) = (a^1 + b^1, a^2 + b^2, a^3 + b^3). \tag{3}$$

$$k\tilde{A} = \begin{cases} (ka^1, ka^2, ka^3), & k \geq 0, \\ (ka^3, ka^2, ka^1), & k < 0. \end{cases} \tag{4}$$

Proposition 1. [31] If $\tilde{A}$ is a triangular fuzzy number, then its expected interval can be computed as follows:

$$EI(\tilde{A}) = [\frac{1}{2}(a^1 + a^2), \frac{1}{2}(a^2 + a^3)].$$

Let $A = [A^L, A^R]$ and $B = [B^L, B^R]$ be two intervals. Then,

$$A + B = [A^L + B^L, A^R + B^R],\ A - B = [A^L - B^R, A^R - B^L], \quad (5)$$

$$\lambda A = \begin{cases} [\lambda A^L, \lambda A^R], & \lambda \geq 0, \\ [\lambda A^R, \lambda A^L], & \lambda < 0, \end{cases} \quad (6)$$

where λ is a real scalar.

Traditional fuzzy sets were developed to model the uncertainty made by human doubt when extracting information. However, the classical fuzzy sets do not account for the reliability of the obtained information. To overcome this limitation, Zadeh [46] proposed Z-numbers.

Definition 3. [23] A Z-number is an ordered pair of fuzzy numbers denoted as $Z = (\tilde{A}, \tilde{R})$. The first component $\tilde{A}$ is a restriction on the values which a real-valued uncertain variable Y can take. The second component $\tilde{R}$ is a measure of reliability for the first component.

In above definition, the membership function of the first component $\tilde{A}$, is $\mu_{\tilde{A}} : X \to [0, 1]$, where X is an arbitrary universal set and the membership function of the second component is $\mu_{\tilde{R}} : [0, 1] \to [0, 1]$.

In this paper, both parts of Z-numbers are considered to be triangular fuzzy numbers. To manipulate the problem involving Z-numbers, first, we convert Z-numbers to triangular fuzzy numbers in three steps, using the method presented by Kang et al. [23]. Consider a Z-number $Z = (\tilde{A}, \tilde{R})$.

Step 1. Convert the second component to a crisp number α as follows:

$$\alpha = \frac{\int_0^1 x\mu_{\tilde{R}}(x)dx}{\int_0^1 \mu_{\tilde{R}}(x)dx}. \quad (7)$$

Step 2. Use α as the weight of the first part (restriction). The weighted Z-number can be denoted as $\tilde{Z}^\alpha = \{(x, \mu_{\tilde{Z}^\alpha}(x))|\mu_{\tilde{Z}^\alpha}(x) = \alpha\mu_{\tilde{A}}(x), x \in X\}$.

Step 3. Convert the irregular fuzzy number (weighted restriction) to regular fuzzy number. The regular fuzzy set can be denoted as

$$\tilde{Z}' = \{(x, \mu_{\tilde{Z}'}(x))|\mu_{\tilde{Z}'}(x) = \mu_{\tilde{A}}(\frac{x}{\sqrt{\alpha}}), x \in \sqrt{\alpha}X\}.$$

Example 1. For the triangular fuzzy number $\tilde{A} = (a^1, a^2, a^3)$ by some simple calculations, one can see from (7) that

$$\alpha = \frac{a^1 + a^2 + a^3}{3}.$$

Let we have an uncertain variable, which takes the value of "almost 3" with the reliability of "almost 0.9". One can represents "almost 3" by the triangular fuzzy number $(2, 3, 4)$ (e.g.), and its reliability by the triangular fuzzy number $(0.8, 0.9, 1)$. Then we have the Z-number $Z = ((2, 3, 4), (0.8, 0.9, 1))$ to represent such an uncertainty. To handle such a Z-number payoff in our game problem, first we convert its reliability to a crisp number as follows:

$$\alpha = \frac{a^1 + a^2 + a^3}{3} = 0.9.$$

Then, we convert the weighted Z-number to triangular fuzzy number according to the proposed approach. So we have

$$\tilde{Z}' = (2\sqrt{0.9}, 3\sqrt{0.9}, 4\sqrt{0.9}) = (1.8974, 2.8461, 3.7948).$$

3 Stackelberg game

Stackelberg games, also known as the leader-follower games, were first introduced in 1952 by the German economist Van Stackelberg to model leadership and commitment. In Stackelberg games, the first player is the leader who chooses a strategy first, then the second player, called the follower, observes the leader's strategy and selects a counter-strategy accordingly. In other words, the game has two players and two stages. In stage 1, the leader's action set is $[0, \infty)$, whereas the follower's only available action is to "do nothing". In stage 2, the follower's action set is $[0, \infty)$, and the leader's only available action is to "do nothing". The problem in this game is to find the optimal strategy for the leader, assuming that the follower optimizes his payoff according to the logical observations that depend on the chosen strategy of the leader. The leader is committed to his decision, which means that if he selects a strategy, then he cannot change it. Therefore, to obtain Stacklerberg's solution, first, the maximum value of the follower's payoff is obtained for the various strategies of the leader. The payoff of the leader is optimized on the best response of the follower. The solution from the above process is called the Stackelberg solution, which can be calculated by the following bilevel linear programming problem[29]:

$$\begin{aligned} \max_{x} \quad & z_1(x, y) = c_1 x + d_1 y \\ & \text{where } y \text{ solves} \\ & \max_{y} \quad z_2(x, y) = c_2 x + d_2 y \\ & \text{subject to } Ax + By \leq b, \\ & \qquad\qquad x \geq 0,\ y \geq 0, \end{aligned} \tag{8}$$

where c_1 and c_2 are n_1-dimensional row coefficient vectors, d_1 and d_2 are n_2-dimensional row coefficient vectors, A is an $m \times n_1$, B is an $m \times n_2$ coefficient matrix, and b is an m-dimensional column constant vector. Moreover, $z_1(x, y)$ and $z_2(x, y)$, respectively, represent the payoff functions of the leader and follower, and x and y represent the strategy of the leader and the follower, respectively.

If the leader commits to the strategy x, the optimal solution $y^*(x)$ is obtained as the logical solution of the follower, by solving the low-level problem

of (8). Assuming that the follower gives a logical solution $y^*(x)$, the leader maximizes his objective function $z(x, y^*(x))$. In this case, the obtained solution is called the Stackelberg solution. This problem can be solved using bilevel programming method (see, e.g., [29]). In this paper, we use the Karush–Kuhn–Tucker (KKT) optimality conditions.

In a Stackelberg game with multi-followers, the leader has to maximize his payoff in the face of several types of followers. He has to choose a strategy to get the most payoff against all of the followers.

First, the followers choose their strategies, so each of them plays his best response. The leader must decide how to play against all of them in order to earn the highest possible payoff. He cannot play his best response against all the followers. Because if he plays his best against one of the followers, he may suffer a significant loss against another, which will reduce his final payoff. Therefore, to obtain Stacklerberg's solution, a multiobjective problem must be solved. Let us call this solution an efficient strategy, defined mathematically here.

Definition 4. Consider a Stackelberg game with p followers. Suppose that y^j is the chosen strategy of the follower type j and that x^j is the chosen strategy of the leader against the follower j. Let $U_l^j(x^j, y^j)$ and $U_f^j(x^j, y^j)$ be the payoffs of leader and follower type j, respectively, for the selected strategies. We call the strategy $x^* = (x^{1*}, x^{2*}, \ldots, x^{p*})$ the efficient strategy for leader, whenever (x^*, y^*) is an efficient solution of the following multiobjective programming problem

$$U_l(x^*, y^*) = \max_x (U_l^1(x^1, y^{1*}), \ldots, U_l^p(x^p, y^{p*})),$$

where y^{j*} represents the best response of the follower type j to the leader's x^j strategy.

4 Security game with multi-attackers

The security game precisely matches the Stackelberg game if we consider the defender as the leader and the attacker(s) as the follower(s). Thus, in this game, the defender commits to a strategy first. Then, the attackers optimize their payoffs, considering the action chosed by the defender. The defender must first commit to a strategy for placing his resources (manpower, equipment, ammunition, etc.) on targets. Then the attackers decide which targets they attack.

Let $T = \{1, \ldots, n\}$ be a set of targets, which may be attacked by p attackers, and assume that the defender has m security forces to protect the targets. The defender and each of the attackers, as the players of this game, try to earn the most payoffs. The attackers select targets that cause the most damage to the defender. On the other hand, the defender aims

to optimize resource assignments to minimize damage. Thus, each player has different strategies for achieving his goal. Each pure strategy of each attacker is to select a target to attack. The mixed strategy of attacker type j is $A^j = (a_1^j, \ldots, a_n^j)$, defined as follows:

$$a_t^j \geq 0, \quad \text{for all } t \in T, \quad \sum_{t=1}^{n} a_t^j = 1, \quad j = 1, \ldots p,$$

where a_t^j is the portion of the force of attacker type j used in attacking to the target t.

Each pure strategy of the defender is choosing a set of targets that have to be protected. If the defender considers only pure strategies, some targets may not be covered, and the attackers may use this weakness to attack them. Note that security resources are limited, and the defender may not be able to cover all the targets fully. Given the limited resources, we define the defender's mixed strategy as $C = (c_1, \ldots, c_n)$, where

$$0 \leq c_t \leq 1, \quad \text{for all } t \in T, \quad \sum_{t=1}^{n} c_t \leq m.$$

In fact, c_t is the amount of coverage of the target $t \in T$ and indicates the probability of the defender succeeding in preventing an attack on the target t. The constraint $0 \leq c_t \leq 1$ ensures that the amount of coverage of the target t have to be less than or equal to one unit of force required for the target t and to prevent force loss. The constraint $\sum_{t=1}^{n} c_t \leq m$ ensures that all of the allocated covers have not to be more than the number of available covering forces.

Suppose that defender and the attacker type j choose strategies C and A^j, respectively. The expected payoffs of the defender and the attacker type j, are

$$\begin{aligned} U_d^j(C, A^j) &= \sum_{t=1}^{n} a_t^j U_d^j(C, t), \quad j = 1, \ldots, p, \\ U_a^j(C, A^j) &= \sum_{t=1}^{n} a_t^j U_a^j(C, t), \quad j = 1, \ldots, p, \end{aligned} \tag{9}$$

if the target t is attacked by a_t^j unit of the force of attacker type j and covered by cover c_t, where

$$\begin{aligned} U_d^j(C, t) &= c_t U_d^{c,j}(t) + (1 - c_t) U_d^{u,j}(t), \\ U_a^j(C, t) &= c_t U_a^{c,j}(t) + (1 - c_t) U_a^{u,j}(t). \end{aligned} \tag{10}$$

In (10), $U_d^{c,j}(t)$ ($U_d^{u,j}(t)$) is defender's payoff when the target t is selected by attacker type j and covered (uncovered) by the defender. Similarly, $U_a^{c,j}(t)$ and $U_a^{u,j}(t)$ are defined for the attacker type j.

This security game, as a Stackelberg game, has several followers (attackers), wherein the defender first selects a strategy, and the attackers surveil the defender's actions. Each attacker tries to maximize his payoff by choosing a strategy that is the best response to the defender's fixed strategy. This is while the defender has to maximize his payoff against several types of attackers. He has to decide how to cover the various targets to get the most payoff. In other words, we are looking for an efficient strategy for the defender. The defender has to consider the set of best responses of attackers to each of his strategies.

An efficient strategy is obtained by solving the following bilevel multiobjective program:

$$
\begin{array}{lll}
(P_1) & Max & (U_d^1(C,A^1), U_d^2(C,A^2), \ldots, U_d^p(C,A^p)) \\
& s.t. & \sum_{t=1}^n c_t \le m, \\
& & 0 \le c_t \le 1, \quad t = 1, \ldots, n, \\
& & \left.\begin{array}{ll} & \textit{where } A^j \textit{ solves} \\ Max & U_a^j(C,A^j) \\ s.t. & \sum_{t=1}^n a_t^j = 1, \\ & a_t^j \ge 0,\ t = 1, \ldots, n, \end{array}\right\} j = 1, \ldots, p,
\end{array}
$$

where $U_d^j(C,A^j)$ and $U_a^j(C,A^j)$ for $j = 1, \ldots, p$ are given by (9).

Theorem 1. The bilevel multiobjective program (P_1) can be solved by solving the following multiobjective optimization problem:

$$
\begin{array}{lll}
(P_2) & Max & (U_d^1(C,A^1), U_d^2(C,A^2), \ldots, U_d^p(C,A^p)) \\
& s.t. & \sum_{t=1}^n c_t \le m, \qquad (11) \\
& & 0 \le c_t \le 1, \quad t = 1, \ldots, n, \qquad (12) \\
& & \left.\begin{array}{l} a_t^j \ge 0, \\ a_t^j \le M\delta_t^j, \\ \sum_{t=1}^n a_t^j = 1, \\ 0 \le k^j - (c_t U_a^{c,j}(t) + (1-c_t)U_a^{u,j}(t)) \le (1-\delta_t^j)M, \\ k^j \in \mathbb{R}, \delta_t^j \in \{0,1\}, \end{array}\right\} \begin{array}{l} j = 1, \ldots, p, \\ t = 1, \ldots, n, \end{array} \\
& & \qquad (13)
\end{array}
$$

where M is a large positive number, and $U_d^1(C,A^1), U_d^2(C,A^2), \ldots, U_d^p(C,A^p)$ are given by (9).

Proof. We prove that the constraints (13) are equivalent to the low-level problem of (P_1). By keeping C, the optimal policy of the defender fixed, the

optimization problem of attacker type j, which gives his best response to the defender's strategy C, is

$$\begin{array}{ll} Max & U_a^j(C, A^j) \\ s.t. & \sum_{t=1}^n a_t^j = 1, \\ & a_t^j \geq 0, \quad t = 1, \ldots, n, \end{array} \tag{14}$$

There is a scalar k^j that satisfies together with a_t^j the following KKT optimality conditions (Note that keeping C fixed, each low-level problem is a linear programming problem, for which the KKT conditions are necessary and sufficient for optimality):

$$\begin{aligned} & k^j \geq c_t U_a^{c,j}(t) + (1 - c_t) U_a^{u,j}(t), \qquad t = 1, \ldots, n, \\ & a_t^j (k^j - (c_t U_a^{c,j}(t) + (1 - c_t) U_a^{u,j}(t))) = 0, \qquad t = 1, \ldots, n, \\ & \sum_{t=1}^n a_t^j = 1, \\ & a_t^j \geq 0, \qquad t = 1, \ldots, n. \end{aligned} \tag{15}$$

By introducing the binary variables δ_t^j for $t = 1, \ldots, n$, and M as a large positive number, the constraints (15) are equivalently written as follows:

$$\begin{aligned} & a_t^j \leq M \delta_t^j, \qquad t = 1, \ldots, n, \\ & 0 \leq k^j - (c_t U_a^{c,j}(t) + (1 - c_t) U_a^{u,j}(t)) \leq (1 - \delta_t^j) M, \qquad t = 1, \ldots, n, \\ & \sum_{t=1}^n a_t^j = 1, \\ & a_t^j \geq 0, \qquad t = 1, \ldots, n. \end{aligned} \tag{16}$$

□

If the defender knows that each attacker attacks at most one target, the constraints (13) can be equivalently replaced by the following constraints:

$$\left.\begin{array}{l} a_t^j \in \{0, 1\}, \\ \sum_{t=1}^n a_t^j = 1, \\ 0 \leq k^j - (c_t U_a^{c,j}(t) + (1 - c_t) U_a^{u,j}(t)) \leq (1 - a_t^j) M, \end{array}\right\} \begin{array}{l} j = 1, \ldots, p, \\ t = 1, \ldots, n. \end{array} \tag{17}$$

A solution to the multiobjective programming problem (P_2) is an efficient strategy for the defender in the security game with multiple attackers.

There are several methods to get an efficient solution to problem (P_2) (e.g., see [16, 34]). In Section 7, we use the weighted sum method. The weights

of the objective functions in problem (P_2), depending on the importance of them for the defender, can be determined by consultation with experts or using methods such as AHP[1] and TOPSIS[2].

5 Deception in multi-attacker security game

In a security game, depending on the available budget, a defender can use deceptive resources to increase his payoff or to reduce the attackers' desire to attack targets. For example, in military operations, the security of various urban or regional centers, different political ceremonies, and so on, defense forces use some types of covert resources and some types of fake ones. Depending on the type of protected targets, the defender uses different deceptive resources, with different probability of deception failure. For example, hidden cameras for protected targets, secret police forces, air marshals on the flight lines, and fake resources are some deceptive resources. The defense force must be able to have the best arrangement of these resources against the attackers according to the available budget. In this section, we consider m real forces and two kinds of deceptive resources: the first kind has a positive effect on the defender's payoff. For example, secret forces have positive effects on the defender's payoff because they have defensive power. The second kind has no effect on the defender's payoff and only reduces the attackers' desire to attack targets. These kinds of deceptive resources do not affect the defense of a target, but they can at least disturb the view of the attackers, and they can reduce the intensity of their attacks. For example, fake resources cause errors in the attacker's observations but do not have defensive power. Therefore they do not increase the defender's payoff. We denote the set of the first (second) kind of deceptive resources by D_1 (D_2). Accordingly, the defender's payoff for a target $t \in T$ is

$$U_d^j(C,t) = c_t U_d^{c,j}(t) + (1-c_t)U_d^{u,j}(t) + \sum_{i \in D1} (c_{t,i} U_d^{cij}(t) + (1-c_{t,i})U_d^{uij}(t)). \tag{18}$$

In (18), for $i \in D_1$ and $t \in T$, $c_{t,i}$ is the amount of deceptive resource coverage, and $U_d^{cij}(t)$ $(U_d^{uij}(t))$ is the defender's payoff from deceptive resource coverage (uncoverage) i against the attacker type j. Note that, obviously, the defender's payoff from using a deceptive resource $i \in D_1$ and a real resource are not the same necessarily. Also, obviously $c_t + \sum_{i \in D_1} c_{t,i} \leq 1$ because more coverage for the target t is useless for the defender. If the importance of a real cover unit differs from a deceptive cover unit, then the mentioned constraint is changed to $wc_t + \sum_{i \in D_1} w_i c_{t,i} \leq 1$, where w and w_i are the weights of real cover and each unit cover type i, respectively.

[1] Analytic Hierarchy Process

[2] Technique for Order of Preference by Similarity to Ideal Solution

To determine the attackers' payoffs, we look at the amount of coverage that they observe and their reaction. Using deceptive coverage resources by the defender is not always 100 percent successful in deceiving the attackers. It is natural that each of them has a failure probability. Suppose that r_i is the probability of deceptive resource's failure for $i \in D_1 \cup D_2$. If the vector of defender choices is $C = (c_1 + \sum_{i\in D_1\cup D_2} c_{1,i}, \ldots, c_n + \sum_{i\in D_1\cup D_2} c_{n,i})$, then the attacker's observation is $E = (e_1, \ldots, e_n)$ in which

$$e_t = c_t + \sum_{i\in D_1\cup D_2} r_i c_{t,i}, \qquad t = 1, \ldots, n. \tag{19}$$

It is assumed that the failure probability of the deceptive resource depends only on its structure. Therefore, the failure probability of one type of deceptive resource is the same for all attackers. Then the payoff of attacker type j is

$$\begin{aligned} U_a^j(E,t) =& c_t U_a^{c,j}(t) + (1-c_t)U_a^{u,j}(t) \\ &+ \sum_{i\in D_1\cup D_2} (r_i c_{t,i} U_a^{cij}(t) + (1-r_i)(1-c_{t,i})U_a^{uij}(t)), \quad t = 1, \ldots, n. \end{aligned} \tag{20}$$

In (20), $U_a^{cij}(t)$ ($U_a^{uij}(t)$) is the payoff of attacker type j in attacking to the target t with (without any) deceptive resource coverage i.

Now, suppose that the defender's budget to create deceptive resources is B, and that he can purchase deceptive resource type i at the cost of B_i per unit. Then to obtain an efficient defense strategy, he has to consider the following constraints:

$$\sum_{t=1}^{n} \sum_{i\in D_1\cup D_2} B_i c_{t,i} \le B. \tag{21}$$

Based on the above discussion, the efficient strategy of the defender is obtained by solving the following multiobjective mixed-integer linear program:

$$\begin{array}{lll} (P_3) & Max & (U_d^1(C,A^1), U_d^2(C,A^2), \ldots, U_d^p(C,A^p)) \\ & s.t. & \sum_{t=1}^{n} c_t \le m, \\ & & 0 \le c_t \le 1 \qquad t = 1, \ldots, n, \\ & & c_t + \sum_{i\in D_1} c_{t,i} \le 1, \qquad t = 1, \ldots, n, \\ & & \sum_{t=1}^{n} \sum_{i\in D_1\cup D_2} B_i c_{t,i} \le B, \\ & & \left.\begin{array}{l} \sum_{t=1}^{n} a_t^j = 1, \\ a_t^j \ge 0, \\ a_t^j \le M\delta_t^j, \\ 0 \le k^j - U_a^j(E,t)) \le (1-\delta_t^j)M, \\ k^j \in \mathbb{R}, \delta_t^j \in \{0,1\}, \end{array}\right\} \begin{array}{l} j = 1, \ldots, p, \\ t = 1, \ldots, n, \end{array} \end{array}$$

where M is a large positive number.

6 Deception in multi-attacker security game in a fuzzy environment

In the real world, the information in a security game is often vague due to the lack of sufficient evidence. For example, a defender may not accurately identify any type of attacker, and attackers may not recognize and/or control different kinds of deceptive resources. Even if they know to some extent what the deceptive resource is, they cannot be 100% sure of what they have seen. In this situation, showing the payoffs in the form of Z-numbers is an appropriate suggestion for expressing ambiguity. The first component represents the player's payoff from selecting a strategy, and the second component shows the measure of the reliability of this selection. In our study, both components of Z-numbers are considered to be triangular fuzzy numbers. For a strategy profile (C, A), the payoff of attacker type j is $\overline{U}_a^j = (\tilde{U}_a^j, \tilde{R}_a^j)$, wherein $\tilde{U}_a^j$ and $\tilde{R}_a^j$ represent the payoff of attacker type j and the reliability of earning this payoff, respectively. The same definition is applied to the defender, and his payoff against attacker type j is denoted by $\overline{U}_d^j = (\tilde{U}_d^j, \tilde{R}_d^j)$. To solve the problem, we convert the Z-numbers to triangular fuzzy numbers by the procedure described in Section 2. Finally, considering the described conversion, we have a triangular fuzzy number for each player's payoff.

Now we have the following programming problem, in which some parameters are triangular fuzzy numbers:

$$
\begin{aligned}
(P_4)\quad Max \quad & (\tilde{U}_d^1(C, A^1), \tilde{U}_d^2(C, A^2), \ldots, \tilde{U}_d^p(C, A^p)) \\
s.t. \quad & \sum_{t=1}^{n} c_t \leq m, \\
& 0 \leq c_t \leq 1, \qquad t = 1, \ldots, n, \\
& c_t + \sum_{i \in D_1} c_{t,i} \leq 1, \qquad t = 1, \ldots, n, \\
& \sum_{t=1}^{n} \sum_{i \in D_1 \cup D_2} B_i c_{t,i} \leq B, \\
& \left.\begin{array}{l} \sum_{t=1}^{n} a_t^j = 1, \\ 0 \leq a_t^j \leq M\delta_t^j, \\ \tilde{U}_a^j(E, t) \leq \tilde{k}^j, \\ \tilde{k}^j \leq (1 - \delta_t^j)\tilde{M} + \tilde{U}_a^j(E, t), \\ k^j \in \mathbb{R}, \qquad \delta_t^j \in \{0, 1\}, \end{array}\right\} \begin{array}{l} j = 1, \ldots, p, \\ t = 1, \ldots, n. \end{array}
\end{aligned}
$$

To solve the problem $(P4)$, let for $s = a, d$ and $j = 1, \ldots, p$, $EI(\tilde{U}_s^j(C, A^j)) = [U_s^{jL}(C, A^j), U_s^{jR}(C, A^j)]$ and $EI(\tilde{k}) = [k^{jL}, k^{jR}]$ be the expected intervals corresponding to fuzzy numbers $\tilde{U}_s^j(C, A^j)$ and $\tilde{k}$, which are calculated according to Proposition 1. Then problem (P_4) is transformed into the following

interval programming problem:

$$
\begin{aligned}
(P_5)\ Max \quad & ([U_d^{1L}(C,A^1), U_d^{1R}(C,A^1)], \ldots, [U_d^{pL}(C,A^p), U_d^{pR}(C,A^p)]) \\
& s.t. \\
& \textstyle\sum_{t=1}^{n} c_t \le m, \\
& 0 \le c_t \le 1, && t = 1, \ldots, n, \\
& c_t + \textstyle\sum_{i \in D_1} c_{t,i} \le 1, && t = 1 \cdots, n, \\
& \textstyle\sum_{t=1}^{n} \sum_{i \in D_1 \cup D_2} B_i c_{t,i} \le B, \\
& \textstyle\sum_{t=1}^{n} a_t^j = 1, && j = 1, \ldots, p, \\
& 0 \le a_t^j \le M\delta_t^j, && j = 1, \ldots, p, \ t = 1, \ldots, n, \\
& [U_a^{jL}(E,t)), U_a^{jR}(E,t))] \le [k^{jL}, k^{jR}], && j = 1, \ldots, p, \ t = 1, \ldots, n, \\
& [k^{jL}, k^{jR}] \le (1 - \delta_t^j)[M, M] + [U_a^{jL}(E,t)), U_a^{jR}(E,t))], && j = 1, \ldots, p, \ t = 1, \ldots, n, \\
& k^{jL}, k^{jU} \in \mathbb{R}, \qquad \delta_t^j \in \{0,1\}, && j = 1, \ldots, p.
\end{aligned}
$$

There are several methods for solving (P_5). In most of them, the main idea is based on intervals' comparison. Instead, Saati, Memariani, and Jahanshahloo [33] proposed a new approach in which a variable is defined corresponding to each interval so that it maximizes the objective functions while satisfying the constraints. More clearly, to solve problem (P_5), we solve the following problem:

$$
\begin{aligned}
(P_6)\ Max\ & (u_1, \ldots, u_p) \\
& s.t.\ \textstyle\sum_{t=1}^{n} c_t \le m, \\
& 0 \le c_t \le 1, \qquad t = 1, \ldots, n, \\
& c_t + \textstyle\sum_{i \in D_1} c_{t,i} \le 1, \qquad t = 1, \ldots, n, \\
& \textstyle\sum_{t=1}^{n} \sum_{i \in D_1 \cup D_2} B_i c_{t,i} \le B, \\
& U_d^{jL}(C,A^j) \le u_j \le U_d^{jR}(C,A^j), \qquad j = 1, \ldots, p, \\
& U_a^{jL}(E,t)) \le v_j \le U_a^{jk}(E,t)), \qquad j = 1, \ldots, p, \\
& k^{jL} \le k_j \le k^{jU}, \qquad j = 1, \ldots, p \\
& \left.\begin{aligned}
\textstyle\sum_{t=1}^{n} a_t^j &= 1, \\
0 \le a_t^j &\le M\delta_t^j, \\
v_j &\le k_j, \\
k_j &\le (1 - \delta_t^j)M + v_j, \\
k^{jL}, k^{jU} \in \mathbb{R}, &\qquad \delta_t^j \in \{0,1\},
\end{aligned}\right\} \begin{array}{l} j = 1, \ldots, p, \\ t = 1, \ldots, n, \end{array}
\end{aligned}
$$

in which

$$
\begin{aligned}
& u_j \in [U_d^{jL}(C,A^1), U_d^{1R}(C,A^j)], && j = 1, \ldots, p, \\
& v_j \in [U_a^{jL}(E,t)), U_a^{jk}(E,t))], && j = 1, \ldots, p, \\
& k_j \in [k^{jL}, k^{jU}], && j = 1, \ldots, p.
\end{aligned}
$$

In fact, by solving problem (P_6), the best choices of the variables u_j, v_j, and k_j are determined from their corresponding intervals so that both maximize the objective functions and satisfy the constraints.

Now, once again, we have a multiobjective problem with crisp parameters. There are several methods to get an efficient solution to this problem (e.g., see [16, 34]). In the solved examples in Section 7, we use the weighted sum method.

Remark 1. The proposed method was extended to solve a multi-attacker security game having Z-numbers payoffs. However, it can be used if the payoffs are triangular fuzzy numbers or real numbers as well. In the first case, steps 1-3 in Section 2 to convert Z-numbers to triangular fuzzy numbers are removed, and in the second case, we have to solve the problem (P_3).

Remark 2 (Comparison with similar works)**.** As mentioned in Remark 1, our method can also be used to solve a security game with triangular fuzzy payoffs. Such a problem was also considered in [7]. Bigdeli, Hassanpour, and Tayyebi [7] have used a pessimistic approach to solve the problem, but our method solves the problem without considering a pessimistic or optimistic point of view. Therefore it is natural to obtain different solutions by the two methods. Furthermore, there is no significant difference between the two methods in view of computational complexity. Therefore, in a security game with triangular fuzzy payoffs, a pessimistic decision-maker can use the method of [7]. The special feature of our work is that we have considered a security game with Z-numbers payoffs and deceptive resources, but in [7], it did not cover these issues.

7 Numerical examples

In this section, we give four examples. In the first example, the defender uses only real resources. In the second example, the defender uses three types of deceptive resources: one fake and two types of secret resources. In both examples, the players' payoffs are considered to be real numbers. In the third example, the defender uses two types of deceptive resources, and the players' payoffs are Z-numbers. The final example is an example solved in [7]. We solve it by our method and compare the solutions obtained from the two methods. All of the optimization problems in examples were solved by Lingo software.

Example 2. In a security game, suppose that three attackers intend to attack four targets and that a defender has $m = 2$ forces to protect these targets. The players' payoffs are given in Tables 2–4. The weights assigned to the tables are 0.2, 0.3, and 0.5, respectively.

By solving the problem (P_2) by the weighted sum method, the following efficient strategy is obtained:

Table 1: Game matrix of defender and attacker type 1 in Example 2

	target 1		target 2		target 3		target 4	
	covered (c)	uncovered (u)	c	u	c	u	c	u
defender	1.5	-0.5	5	-6	2	-1	9	-8
attacker	-1.5	2	-4	5	-2	3	-4	9

Table 2: Game matrix of defender and attacker type 2 in Example 2

	target 1		target 2		target 3		target 4	
	c	u	c	u	c	u	c	u
defender	2	-0.5	6	-5	3	-2	11	-10
attacker	-1	1	-3	4	-2	3	-4	8

$$C = (0.34, 0.55, 0.40, 0.62).$$

Since the defender has two covering resources (two defense forces), it is concluded that 17, 27.5, 20, and 31 percent of the forces should be assigned to the targets t_1, t_2, t_3, and t_4, respectively, and 4.5 % of defense forces are not assigned.

As the tables show, the target t_4 has greater payoffs for the defender than the other targets. Also, for all three attackers, this target has greater payoffs than the other targets. Therefore, it is more likely to attack this target. In the case of the target t_1 is the opposite. In the solution obtained by our method, the highest coverage was obtained for the target t_4, and the lowest coverage was achieved for the target t_1.

Example 3. Consider a security game in which three attackers intend to attack four targets. The defender has $m = 1$ real security force to protect the targets. Decision-makers (experts) have provided the following information: The defender uses three types of deceptive resources. He uses an experienced and trained secret force with a 0.2 probability of being exposed. At the same time, a real force acts as a covert force with a lower cost and 0.4 probability of being exposed (secret normal force). The payoff of an experienced secret force is 1.3 times more than that of a real security force. The probability that the attacker will not distinguish these fake resources is 0.4 (i.e., his failure probability is 0.6). The required budget for each deceptive force unit is 1,

Table 3: Game matrix of defender and attacker type 3 in Example 2

	target 1		target 2		target 3		target 4	
	c	u	c	u	c	u	c	u
defender	1	-0.5	6	-4.5	3	-1	10	-9
attacker	-1	0.5	-4	5	-3	4	-6	10

Table 4: Game matrix of defender and attacker type 1 in Example 3

	cover's type	target 1		target 2		target 3		target 4	
		c	u	c	u	c	u	c	u
defender	real	5	-3	8	-9	2	-2.5	3	-5
attacker	real/secret normal force	-2	3	-4	6	-3	5	-4	5
	experienced secret force	-3	3	-5	6	-4	5	-5	5
	fake	3	-3	6	-6	5	-5	3	-5

Table 5: Game matrix of defender and attacker type 2 in Example 3

	cover's type	target 1		target 2		target 3		target 4	
		c	u	c	u	c	u	c	u
defender	real	4	-1	10	-7	1.5	-1	2	-2.5
attacker	real/secret normal force	-3	2.5	-2	1.5	-2	1	-3	1
	experienced secret force	-4	2.5	-3	1.5	-2	1	-1	1
	fake	2.5	-2.5	1.5	-1.5	1	1	2	-1

3, and 7, respectively, for fake, secret normal, and experienced secret force, and the defender's available budget is 12. The players' payoffs are given in Tables 4–6.

Solving the problem (P_3) by weighted sum method with equal weights for the objective functions yields the solution given in Table 3.

This means that in order to protect four targets with the mentioned security resources, the defender must plan the presence of the real security resource with 42% in the target 1, 50% in the target 3, and 7% in the target 4. The target 2 does not require a real resource, and it is sufficient to be protected by an experienced secret force unit and 0.79 fake force unit. Likewise, the defender must deploy other deceptive security resources according to Table 3.

Example 4. Consider a security game with three targets and two attackers. The defender uses $m = 1$ real security force and two secret sources to protect the targets. Secret forces are exposed to 0.3 probability. The required budget for a secret force unit is 5, and the defender's available budget is 9. The value of each unit of secret force is 1.5 times that of a real force unit. The players' payoffs are Z-numbers given in Tables 8 and 9.

Table 6: Game matrix of defender and attacker type 3 in Example 3

	cover's type	target 1		target 2		target 3		target 4	
		c	u	c	u	c	u	c	u
defender	real	5	-2	6	-4	3	-1	4	-3
attacker	real/secret normal force	-3	3	-2	5	-2	4	-1	1
	experienced secret force	-3	3	-2	5	-2	4	-1	1.5
	fake	3	-3	5	-4	4	-5	1	-2

Table 7: Amounts of targets coverages in Example 3

	i=real	i=experienced secret force	i= secret normal force	i= fake
t=1	0.42	0	0	0
t=2	0	1	0	0.79
t=3	0.5	0	0.5	0.2
t=4	0.07	0	0.5	0

Table 8: Game matrix of defender and attacker type 1 in Example 4

		defender		attacker type 1	
		c	u	c	u
t_1	real	((6,6,7), (0.8,.0.9,1))	((-3,-2,-2), (0.8,.0.9,1))	((-3,-3,-2), (0.8,.0.9,1))	((2,3,4), (0.7,0.8,.0.9))
	secret	((3,3,4), (0.6,0.7,0.8))	((-2,-2,-1), (0.6,0.7,0.8))	((-4,-3,-2), ,(0.6,0.7,0.8))	((1,2,3), (0.6,0.7,0.8))
t_2	real	((6,6,7), (0.7,0.8,0.9))	((-2,-1.5,-1), ,(0.7,0.8,0.9))	((-5,-4,-3), (0.7,0.8,0.9))	((2,3,3), (0.7,0.8,0.9))
	secret	((3,4,5), (0.6,0.7,0.8))	((-2,-1,-1), (0.6,0.7,0.8))	((-2,-2-1), (0.6,0.7,0.8))	((1,2,3), (0.6,0.7,0.8))
t_3	real	((2,4,4), (0.8,.0.9,1))	((-1.5,-1,-1), (0.8,.0.9,1))	((-3,-2,-1), (0.8,.0.9,1))	((1,2,2), (0.8,.0.9,1))
	secret	((2,2,3), (0.6,0.7,0.8))	((-3,-2,-1), (0.6,0.7,0.8))	((-2,-2,-1), (0.6,0.7,0.8))	((1,2,3), (0.6,0.7,0.8))

Table 9: Game matrix of defender and attacker type 2 in Example 4

		defender		attacker type 2	
		c	u	c	u
t_1	real	((5,5,6), (0.8,.0.9,1))	((-2,-2,-1), (0.8,.0.9,1))	((-2,-2,-1), (0.8,.0.9,1))	((1,2,3), (0.8,.0.9,1))
	secret	((5,6,6), (0.6,0.7,0.8))	((-3,-2,-1), (0.6,0.7,0.8))	((-4,-3,-2), (0.6,0.7,0.8))	((1,2,3), ,(0.6,0.7,0.8))
t_2	real	((4,4,5), (0.7,0.8,0.9))	((-1,-0.5,0), (0.7,0.8,0.9))	((-2,-1,-1), (0.7,0.8,0.9))	((2,2,3), (0.7,0.8,0.9))
	secret	((5,6,6), (0.6,0.7,0.8))	((-2,-2,-1), (0.6,0.7,0.8))	((-2,-2,-1), (0.6,0.7,0.8))	((2,3,4), (0.6,0.7,0.8))
t_3	real	((3,3,4), (0.7,0.8,0.9))	((-3,-2,-2), (0.7,0.8,0.9))	((-4,-3,-3), (0.7,0.8,0.9))	((1,2,4), (0.7,0.8,0.9))
	secret	((3,3,4), (0.6,0.7,0.8))	((-1.5,-1,-0.5), (0.6,0.7,0.8))	((-3,-2,-1), (0.6,0.7,0.8))	((1,3,4), (0.6,0.7,0.8))

Table 10: Game matrix of defender and attacker type 1 as triangular fuzzy numbers in Example 4

		defender		attacker type 1	
		c	u	c	u
t_1	real	(5.75,5.75,6.6)	(-2.8,-1.8,-1.8)	(-2.8,-2.8,-1.9)	(1.8,2.8,3.8)
	secret	(2.5,2.5,3.3)	(-1.6,-1.6,-0.8)	(-3.3,-2.5,-1.67)	(0.8,1.6,2.5)
t_2	real	(5.3,5.3,6.2)	(-1.78,-1.2,-0.8)	(-4.4,-3.5,-2.6)	(1.78,2.68,2.68)
	secret	(3.3,4.1,4.1)	(-1.6,-0.8,-0.8)	(-1.67,-1.67,-0.8)	(0.8,1.67,2.5)
t_3	real	(1.8,3.8,3.8)	(-1.2,-0.8,-0.8)	(-2.8,-1.9,-0.9)	(0.94,1.9,1.9)
	secret	(1.6,1.6,2.5)	(-2.5,-1.6,-0.8)	(-1.67,-1.67,-0.83)	(0.8,1.67,2.5)

Table 11: Game matrix of defender and attacker type 2 as triangular fuzzy numbers in Example 4

		defender		attacker type 2	
		c	u	c	u
t_1	real	(4.7,4.7,5.6)	(-1.7,-1.7,-0.8)	(-2.84,-1.89,-0.94)	(0,1.89,2.84)
	secret	(4.2,5,5)	(-2.5,-1.6,-0.8)	(-1.67 ,-0.83,-0.83)	(0.83,1.67,2.5)
t_2	real	(3.5,3.5,4.4)	(-0.8,-0.4,0)	(-1.7,-0.8,-0.8)	(1.78,2.68,2.68)
	secret	(4.1,4.1,5)	(-1.7,-0.8,-0.8)	(-1.67,-1.67,-0.83)	(1.67,2.5,3.34)
t_3	real	(2.5,2.5,3.3)	(-2.6,- 1.8,-1.8)	(-3.5,-2.68,-2.68)	(0.89,1.78,3.57)
	secret	(2.5,2.5,3.3)	(-1.2,-0.8,-0.4)	(-2.5,-2.5,-1.67)	(0.83,2.5,3.3)

Now, for the given player's payoffs, we calculate the $\sqrt{\alpha}$ values from (7), and apply them as the weights of payoffs. Then we have triangular fuzzy payoffs given in Tables 10 and 1.

Solving the problem (P_6) by the weighted sum method (with equal weights for the objective functions) for these data yields the following solution:

$$C_{real} = (0.52, 0.23, 0.04),\ C_{secret} = (0, 0.83, 0.97).$$

This means that the defender should allocate 52%, 23%, and 4% of his real forces to the targets 1, 2, and 3, respectively. Because of the constraint $\sum_{t=1}^{n} c_t \leq m$, not all resources will necessarily be allocated in the optimal solution. In this example, 79% of the real resources are used and 21% of them remain unused. Also, with the available budget, he can allocate 41.5% of the two secret forces (i.e., 0.83 of the two units) to the target 2 and 48.5% (i.e., 0.97 of the two units) to the target 3.

Example 5. Consider the security game with two targets and three attackers solved in [7]. The players' payoffs are given in Tables 12–14.

Solving this example by our method (Problem P_6, without deceptive resources) yields the payoff 4.7 and the cover $C = C_{real} = (0.05, 0.95)$. This example was solved in [7] with a pessimistic viewpoint and the defender's payoff was obtained 3.19 and $C = (0.29, 0.79)$, which is not better than our

Table 12: Game matrix of defender and attacker type 1 in Example 5

	target 1		target 2	
	c	u	c	u
defender	(3,5,6)	(-3,-2,-1)	(9,10,11)	(2,3,5)
attacker	(-2,-1,0)	(2,4,5)	(-2,-1,0)	(9,10,11)

Table 13: Game matrix of defender and attacker type 2 in Example 5

	target 1		target 2	
	c	u	c	u
defender	(0,1,2)	(0,0,0)	(1,2,4)	(-3,-2,-1)
attacker	(-2,-1,0)	(0,1,2)	(0,0,0)	(3,5,6)

solution. Such a result was expected because the solution of [7] is a pessimistic solution.

8 Conclusions

Optimization of force allocation is an important issue in war situations for enemy points of attack, and in any situation (whether war or not), for sensitive centers and infrastructure. A motivated attacker monitors defense forces and takes advantage of the pattern of forces. Defenders must be able to predict the attacker's reaction to different defensive strategies with the highest probability. On the other hand, resource limitation is a major problem in many security areas. Game theory can be used as a valuable tool to analyze these issues and especially to determine the optimal strategy in case of a conflict of interests. Security games are used to solve various security issues according to the type and number of attackers and defenders.

In this paper, a mathematical model was proposed to allocate defense forces in a security game with several attackers. Defenders can use deceptive resources to reduce attack, intensity, productivity, or costs. Applying these resources can fail with certain probabilities. Given these probabilities and budget constraints, a mathematical model was introduced to optimize the allocation of these deceptive resources. In the proposed model, the available

Table 14: Game matrix of defender and attacker type 1 in Example 5

	target 1		target 2	
	c	u	c	u
defender	(1,2,4)	(-2,-1,0)	(2,3,5)	(-3,-2,-1)
attacker	(-3,-2,-1)	(0,1,2)	(-5,-3,-2)	(2,4,5)

budget, the importance of targets for attackers and defenders, and their possible strategies were considered to optimize the allocation of forces. Also, when the defender uses deceptive resources, the ambiguity in the amount of players' payoffs for both players increases. Hence, the players' payoffs were considered as Z-numbers. Then, the problem was solved in a two-stage procedure. In the first stage, the Z-numbers were converted to triangular fuzzy numbers, and in the second stage, the triangular fuzzy numbers were converted to intervals using their expected intervals. Then the interval programming problem was solved by an available method in the literature. Finally, the applicability of the proposed methods was illustrated by some numerical examples.

There are various types of uncertain data, for example, intuitive fuzzy numbers, type-2 fuzzy numbers, and so on. The introduced model handles the payoffs of real, triangular fuzzy numbers, and Z-numbers. However, it cannot be used for other types of fuzzy numbers (or types of uncertainty). As a suggestion, security games with multi-attacker can be solved with other kinds of uncertainty in payoffs.

References

1. Basilico, N., Gatti, N., and Amigoni, F. *Leader-follower strategies for robotic patrolling in environments with arbitrary topologies*, 8th International Conference on Autonomous Agents and Multi-Agent Systems, (2009) 57–64.

2. Bigdeli, H. and Hassanpour, H. *Modeling and solving multiobjective security game problem using multiobjective bilevel problem and its application in metro security system*, Journal of Electronical and Cyber Defence, Special Issue of the International Conference on Combinatorics, Cryptography and Computation (In Persian), (2017) 31–38.

3. Bigdeli, H. and Hassanpour, H. *An approach to solve multi-objective linear production planning games with fuzzy parameters*, Yugosl. J. Oper. Res. 28(2), (2018) 237–248.

4. Bigdeli, H. and Hassanpour, H. *Solving defender-attacker game with multiple decision makers using expected-value Model*, Casp. J. Math. Sci. (CJMS) (2020).

5. Bigdeli, H., Hassanpour, H. and Tayyebi, J. *Optimistic and pessimistic solutions of single and multi-objective matrix games with fuzzy payoffs and analysis of some military cases*, Scientific Journal of Advanced Defense Science and Technology (In Persian), (2017) 133–145.

6. Bigdeli, H., Hassanpour, H. and Tayyebi, J. *Constrained bimatrix games with fuzzy goals and its application in nuclear negotiations*, Iran. J. Numer. Anal. Optim., 8(1), (2018) 81–110.

7. Bigdeli, H., Hassanpour, H. and Tayyebi, J. *Multiobjective security game with fuzzy payoffs*, Iran. J. Fuzzy Syst. 16(1), (2019) 89–101.

8. Brown, G., Carlyle, M., Diehl, D., Kline, J. and Wood, K. *A two-sided optimization for theater ballistic missile defense*, Oper. Res. 53(5), (2005) 745–763.

9. Buckley, J.J. *Joint solution to fuzzy programming problems*, Fuzzy Sets Syst. 72(2), (1995) 215–220.

10. Cohen, F. and Koike, D. *Misleading attackers with deception*, In Proceedings from the Fifth Annual IEEE SMC Information Assurance Workshop, (2004) 30–37.

11. Conitzer, V. and Sandholm, T. *Computing the optimal strategy to commit to*, 7th ACM conference on Electronic commerce, (2006) 82–90.

12. Daniel, D.C. and Herbig, K.L. *Strategic military deception*, New York: Pergamon Press, 1981.

13. Dickerson, J.P., Simari, G.I., Subrahmanian, V.S. and Kraus, S. *A graph-theoretic approach to protect static and moving targets from adversaries*, 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1, (2010) 299–306.

14. Do, C.T., Tran, N.H., Hong, C., Kamhoua, C.A., Kwiat, K.A., Blasch, E., Ren, S., Pissinou, N. and Iyengar, S.S. *Game theory for cyber security and privacy*, ACM Comput. Surv. (CSUR), 50(2), (2017) 1–37.

15. Dubois, D. and Prade, H. *Fuzzy sets and statistical data*, Eur. J. Oper. Res. 25(3), (1986) 345–356.

16. Ehrgott, M. *Multicriteria optimization*, Springer Science & Business Media, 2005.

17. Esmaieli, S., Hassanpour, H. and Bigdeli, H. *Lexicographic programming for solving security game with fuzzy payoffs and computing optimal deception strategy*, Defensive Future Study Researches Journal (In Persian), 5(16), (2020) 89–108.

18. Fang, F., Nguyen, T.H., Pickles, R., Lam, W.Y., Clements, G.R., An, B., Singh, A., Tambe, M. and Lemieux, A. *Deploying PAWS: field optimization of the protection assistant for wildlife security*, In Twenty-Eighth IAAI Conference, (2016).

19. Frank. Jr. and Willard C. *Politico military deception at sea in the Spanish civil war, 1936-39.*, Intell. Natl. Secur. 5(3), (1990) 84–112.

20. Fugate, S. and Ferguson-Walter, K. *Artificial intelligence and game theory models for defending critical networks with cyber deception*, AI Mag. 40(1), (2019) 49–62.

21. Hamilton, D.L. *Deception in Soviet military doctrine and operations*, NAVAL POSTGRADUATE SCHOOL MONTEREY CA, 1986.

22. Heilpern, S. *The expected valued of a fuzzy number*, Fuzzy sets Syst. 47, (1992) 81–86.

23. Kang, B., Wei, D., Li, Y. and Deng, Y. *A method of converting Z-number to classical fuzzy number*, J. Inf. Comput. Sci. 9(3), (2012) 703–709.

24. Karmakar, S., Seikh, M.R. and Castillo, O. *Type-2 intuitionistic fuzzy matrix games based on a new distance measure: Application to biogas-plant implementation problem*, Appl. Soft Comput. 106, (2021) p.107357.

25. Korzhyk, D., Conitzer, V. and Parr, R. *Complexity of computing optimal Stackelberg strategies in security resource allocation games*, 24th AAAI Conference on Artificial Intelligence, (2010) 805–810.

26. Letchford, J. and Vorobeychik, Y. *Computing randomized security strategies in networked domains*, Applied adversarial Reasoning and Risk Modeling, In Workshops at the Twenty-Fifth AAAI Conference on Artificial Intelligence, 2011.

27. Lye, K. and Wing, J.M. *Game strategies in network security*, Int. J. Inf. Secur. 4(1), (2005) 71–86.

28. McQueen, M.A. and Boyer, W.F. *Deception used for cyber defense of control systems*, 2nd Conference on Human System Interactions, (2009) 624–631.

29. Nishizaki, I. and Sakawa, M. *Stackelberg solutions to multiobjective two-level linear programming problems*, J. Optim. Theory Appl. 103(1), (1999) 161–182.

30. Oikonomakis, P. *Strategic military deception prerequisites of success in technological environment*, 2016.

31. Ren, A., Wang, Y. and Xue, X. *Interactive programming approach for solving the fully fuzzy bilevel linear programming problem*, Knowl Based Syst. 99, (2016) 103–111.

32. Rowe, N.C., Custy, E.J. and Duong, B.T. *Defending cyberspace with fake honeypots*, J. Comput. 2(2), (2007) 25–36.

33. Saati, S.M., Memariani, A. and Jahanshahloo, G.R. *Efficiency analysis and ranking of DMUs with fuzzy data*, Fuzzy Optim. Decis. Mak. 1(3) (2002) 255-267.

34. Sakawa, M. *Fuzzy sets and interactive multiobjective optimization*, Plenumpress, New York and London, 1993.

35. Sakawa, M. and Nishizaki, I. *Cooperative and noncooperative multi-level programming*, Springer, New York and London, 2009.

36. Seikh, M.R., Dutta, S. and Li, D.F. *Solution of matrix games with rough interval pay-offs and its application in the telecom market share problem*, Int. J. Intell. Syst. 36(10), (2021) 6066–6100.

37. Seikh, M.R., Karmakar, S. and Castillo, O. *A novel defuzzification approach of Type-2 fuzzy variable to solving matrix games, An application to plastic ban problem*, Iran. J. Fuzzy Syst. 18(5), (2021) 155–172.

38. Seikh, M.R., Karmakar, S. and Nayak, P.K. *Matrix games with dense fuzzy payoffs*, Int. J. Intell. Syst. 36(4), (2021) 1770–1799.

39. Seikh, M.R., Karmakar, S. and Xia, M. *Solving matrix games with hesitant fuzzy pay-offs*, Iran. J. Fuzzy Syst. 17(4), (2020) 25–40.

40. Sokri, A. *Optimal resource allocation in cyber-security: A game theoretic approach*, Procedia Comput. Sci. 134, (2018) 283–288.

41. Tambe, M. *Security and game theory: algorithms, deployed systems, lessons learned*, Cambridge University Press, 2011.

42. Trejo, K.K., Clempner, J.B. and Poznyak, A.S. *A Stackelberg security game with random strategies based on the extraproximal theoretic approach*, Eng. Appl. Artif. Intell. 37, (2015) 145–153.

43. Trejo, K.K., Kristal K., Clempner, J.B. and Poznyak, A.S. *Adapting strategies to dynamic environments in controllable Stackelberg security games*, IEEE 55th Conference on Decision and Control (CDC), (2016) 5484–5489.

44. Wang, A., Cai, Y., Yang, W. and Hou, Z. *A Stackelberg security game with cooperative jamming over a multiuser OFDMA network*, IEEE Wireless Communications and Networking Conference (WCNC), (2013) 4169–4174.

45. Yin, Y., An, B., Vorobeychik, Y. and Zhuang, J., *Optimal deceptive strategies in security games: A preliminary study*, In Proc. of AAAI, 2013.

46. Zadeh, L.A. *A note on Z-numbers,* Inform. Sci. 181(14) (2011) 2923–2932.

47. Zhu, Q. *Game theory for cyber deception: A tutorial*, 6th Annual Symposium on Hot Topics in the Science of Security, (2019) 1–3.

How to cite this article
S. Esmaeeli, H. Hassanpour and H. Bigdeli . *Iranian Journal of Numerical Analysis and Optimization*, 2022; 12(3 (Special Issue), 2022): 542-566. doi: 10.22067/ijnao.2022.71302.1046.