Colonel Blotto Game Aided Attack-Defense Analysis in Real-World Networks

Sanghai Guan*‡, Jingjing Wang*‡, Chunxiao Jiang†, Zhu Han§, Yong Ren*, and Abderrahim Benslimane¶
*Department of Electronic Engineering and †Tsinghua Space Center, Tsinghua University, Beijing, 100084, China
‡Tsinghua National Laboratory for Information Science and Technology (TNList), Tsinghua University, Beijing, 100084, China
§University of Houston, Houston, TX, 77004, USA
¶CERI/LIA Laboratory – BP 1228, University of Avignon, 84911 Avignon Cedex 9, France
Email: gsh17@mails.tsinghua.edu.cn

***Abstract*—Large scale network systems such as Internet, smart grids and social networks become an indispensable part of our daily life. However, due to their inherent vulnerability as well as the limited management and operational capability, these network systems are constantly under the threat of malicious attackers. In such attack-defense scenarios, it is particularly significant to make the best use of defenders' limited resources and capability. In this paper, we propose a networked Colonel Blotto game, where the attackers and defenders allocate the limited resources on network nodes, and their utility depends on certain network performance metrics, which are defined for evaluating the performance of the whole network system. Furthermore, considering the complexity of the equilibrium analysis in large scale network systems, a co-evolution based algorithm is proposed for obtaining the practical action sets as well as achieving the mixed-strategy Nash equilibrium. Finally, relying on three real-world network systems, i.e., computer networks, Internet of vehicles and online social networks, simulation results show the effectiveness and feasibility of our proposed model, which is conducive to the design, management and maintenance of real-world network systems.**

***Index Terms*—Colonel Blotto game, resource allocation, network attack-defense security.**

I. Introduction

Network systems, such as Internet, smart grids, social networks, etc., play a critical role in human society. However, various security vulnerabilities threaten their normal operation and provide opportunities for malicious attackers, who can trigger huge damage just by attacking few key nodes. By contrast, it is also beneficial of improving the reliability of the network system by emphatically protecting these weak key nodes. Considering the limitation of attack-defense resources and capability of both the attacker and defender, rationally allocating depletable attack-defense resource on the whole network system becomes an important issue [1]–[4]. In order to model such attack-defense scenarios, game theory is a powerful tool, and the Colonel Blotto game [5] is a useful model for attack-defense resource allocation, where two players are in charge of the force assignment for a number of battlefields and attempt to win as many battlefields as possible. The Colonel Blotto game has been widely studied and applied in a range of fields [6], such as military, information forecasting, social science, communication and computer networks, etc [7]. As for communication and computer networks, Wu et al. [8] investigated the optimal power selection problem against jamming attacks. Furthermore, Fuchs and Khargonekar [9] constituted a game model for resource allocation with asymmetric information in wireless sensor networks. Hajimirsadeghi et al. [10] proposed a Colonel Blotto game based dynamic spectrum allocation scheme in the multi-user environment.

However, these game models just establish a simple and linear relationship between the global utility and the results on each battlefield. In practical systems, the global utility and the result of each battlefield often have a complex and implicit relationship. Moreover, with the increase of the number of resources and battlefields, the number of feasible actions grows exponentially. Hence, most related works just concentrate on simple toy systems and efficient solutions for large-scale network systems are needed. To address the aforementioned issues, in this paper, we propose a novel networked Colonel Blotto game to study the attack-defense problem in network systems. The original contributions of this paper can be summarized as follows:

- A networked Colonel Blotto game model is constituted for the ubiquitous attack-defense resource allocation in network systems. Moreover, several performance metrics are defined for evaluating the network performance as well as for formulating the utility of proposed game.
- As for games in large scale network systems with enormous action sets of both players, considering the complexity of finding the equilibrium, we propose a genetic algorithm based co-evolution algorithm for generating practical action sets and for searching quality strategies.
- Our networked Colonel Blotto game model is applied to three large-scale network systems, i.e., Internet, vehicular networks and online social networks. The real-world data-driven simulations verify its validity and feasibility.

The remaining content is arranged as follows. We introduce our networked Colonel Blotto game model and network performance metrics in Section II. Section III provides the detailed analysis of this zero-sum game and presents the co-evolution based algorithms for seeking quality strategies. Simulation results based on the three real-world dataset are shown in Section IV, followed by the conclusions in Section V.

978-1-5386-4727-1/18/$31.00 ©2018 IEEE

II. Game Model

A. Networked Colonel Blotto Game

The networked Colonel Blotto game is a one-shot two-player zero-sum game, where two players are the defender and the attacker, respectively. Firstly, the network system is defined as an undirected graph denoted by $\boldsymbol{G} = \{\mathbb{V}, \mathbb{E}\}$, where $\mathbb{V} = \{v_1, v_2, \ldots, v_N\}$ represents the set of nodes and $\mathbb{E} = \{e_1, e_2, \ldots, e_M\}$ is the set of edges. N represents the total number of nodes, while M denotes the total number of edges. Each edge can be expressed by the set of two nodes it connects. For example, $e_k = \{v_i, v_j\}$ represents that e_k is the edge that connects nodes v_i and v_j. As for the defender, the quantity of defense resources is A_1, which can be allocated on nodes for preventing potential attacks. Hence, the action of the defender can be represented as $\boldsymbol{a}_1 = [a_1^1, a_1^2, \ldots, a_1^N]$, where $a_1^i \geq 0$ stands for the quantity of defense resources that allocated on node v_i, and we have $\sum_{i=1}^N a_1^i = A_1$. By contrast, the quantity of attack resources is A_2, and the action of the attacker is $\boldsymbol{a}_2 = [a_2^1, a_2^2, \ldots, a_2^N]$, where a_2^i also satisfies $a_2^i \geq 0$ and $\sum_{i=1}^N a_2^i = A_2$. The action sets of the defender and attacker are $\mathbb{A}_1$ and $\mathbb{A}_2$, respectively. Moreover, we assume that the self-defense capability of node v_i is $a_0^i \geq 0$, and $\boldsymbol{a}_0 = [a_0^1, a_0^2, \ldots, a_0^N]$.

Then, the result of the "battle" on each node depends on the quantity of the attack-defense resources that two players allocate. Hence, we can divide the nodes into two sets, i.e., the set of nodes $\mathbb{V}_1$ controlled by the defender, and the set of nodes $\mathbb{V}_2$ controlled by the attacker. Then, the result of the "battle" on node v_i follows:

$$v_i \in \begin{cases} \mathbb{V}_1, & \text{if } a_0^i + a_1^i \geq a_2^i, \\ \mathbb{V}_2, & \text{if } a_0^i + a_1^i < a_2^i. \end{cases} \tag{1}$$

Moreover, edges can also be divided into three categories according to the nodes they connect. These three sets are denoted as $\mathbb{E}_{11}$, $\mathbb{E}_{12}$ and $\mathbb{E}_{22}$. Specifically, edge $e_k \in \mathbb{E}_{ij}$ $(i, j \in \{1, 2\},\ i \leq j)$ indicates that the two nodes it connects are from $\mathbb{V}_i$ and $\mathbb{V}_j$. Fig. 1 shows the relationship between nodes' attack-defense resource allocated and their affiliation.

In order to compare the performance of the whole network, we denote the original network system as $\boldsymbol{G}'$, while the network system after the game is $\boldsymbol{G}''$. In the original network system $\boldsymbol{G}'$, we assume that $a_1^i = a_2^i = 0$. Finally, the utility function of the game can be given by:

$$u_1(\boldsymbol{a}_1, \boldsymbol{a}_2) = -u_2(\boldsymbol{a}_1, \boldsymbol{a}_2) = f(\boldsymbol{G}'') - f(\boldsymbol{G}'), \tag{2}$$

where u_1 represents the utility of the defender, while u_2 is the utility of the attacker. Moreover, $f(\cdot)$ denotes the evaluation function of the network performance. Hence, the defender's goal is to minimize the performance loss, while the attacker aims for maximizing it, which constitutes a zero-sum game.

B. Network Performance Evaluation

In this subsection, we will introduce some commonly used network characteristics to construct evaluation function $f(\cdot)$. For the convenience of presentation, we adopt the adjacency matrix $\boldsymbol{W} = (w_{ij})_{N \times N}$ to represent the network topology. In an unweighted graph, $w_{ij} \in \{0, 1\}$ represents the existence of an edge $\{v_i, v_j\}$, while $w_{ij} \geq 0$ denotes the weight of the edge $\{v_i, v_j\}$ in a weighted graph.

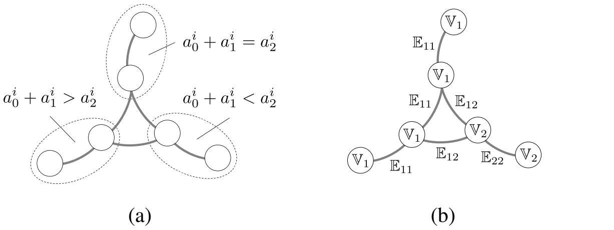


Fig. 1. The relationship between the nodes' attack-defense resources allocated and their affiliation. (a) The quantity of attack-defense resources allocated on each node. (b) The categories that nodes and edges belong to.

1) Network Connectivity: If some nodes are controlled and damaged by the attacker, the network connectivity will change. Therefore, the weight of edge w_{ij} can be defined as:

$$w_{ij} = \begin{cases} 1, & \text{if } \{v_i, v_j\} \in \mathbb{E}_{11}, \\ 0, & \text{if } \{v_i, v_j\} \notin \mathbb{E}_{11}, \end{cases} \tag{3}$$

in an unweighted graph. Because there may exist unconnected parts in the network, the network can be divided into one or more sub-networks. the sub-network with most nodes is named as the giant component. If the giant component contains n nodes, the network connectivity based evaluation function can be denoted as:

$$f(\boldsymbol{G}) = n. \tag{4}$$

2) Average Path Length: Sometimes, attacks may not damage the network's connectivity, but may still influence the performance of edges. Here the path $\boldsymbol{p}_{i_1, i_K}$ between nodes v_{i_1} and v_{i_K} can be represented by an ordered but not repeated node sequence, i.e., $\boldsymbol{p}_{i_1, i_K} = [v_{i_1}, v_{i_2}, \ldots, v_{i_K}]$. As for a pair of adjacent nodes v_{i_k} and $v_{i_{k+1}}$ on the path, there exists $w_{i_k, i_{k+1}} > 0$. The length of a path is defined as the total weight of the edges it includes, i.e.,

$$r(\boldsymbol{p}_{i_1, i_K}) = \sum_{[v_{i_k}, v_{i_{k+1}}] \in \boldsymbol{p}_{i_1, i_K}} w_{i_k, i_{k+1}}, \tag{5}$$

where $[v_{i_k}, v_{i_{k+1}}]$ denotes two adjacent nodes on the path. Note that there often exist multiple paths between two nodes. Hence, the shortest length of the path between two nodes can be given by:

$$r_{ij}^\star = \min_{\boldsymbol{p}_{ij}} r(\boldsymbol{p}_{ij}). \tag{6}$$

The average path length of the network is calculated as:

$$\bar{r} = \frac{\sum_{i \neq j} r_{ij}^\star}{N(N-1)}. \tag{7}$$

Thus, the average path length based evaluation function can be formulated as:

$$f(\boldsymbol{G}) = -\bar{r}. \tag{8}$$

3) Transmission Capability: In transmission scenarios such as computer virus in computer networks and rumors in social networks, the susceptible-infection (SI) propagation model is commonly adopted. In this model, nodes have two states, i.e., the susceptible state (S) and the infected state (I). The susceptible node can be infected by its neighboring infected nodes. Here we take the time after the game as time step $t = 0$, where the nodes in $\mathbb{V}_2$ are initial infected nodes, while the nodes in $\mathbb{V}_1$ are the initial susceptible nodes. Relying on the SI propagation model, at each time step t, node v_i may be infected and added into $\mathbb{V}_2$ with the probability of:

$$p_i(t)=\begin{cases}\dfrac{\sum_{\{j:\,\{v_i,v_j\}\in\mathbb{E}_{12}\}} c_j}{\sum_{\{j:\,\{v_i,v_j\}\in\mathbb{E}\}} c_j}, & \text{if } v_i\in\mathbb{V}_1,\\[2ex] 1, & \text{if } v_i\in\mathbb{V}_2,\end{cases} \tag{9}$$

where c_j is the influence of node v_j. Then, we define the average diffusion time $\bar{t}$ as the expected time that the proportion of infected nodes reaching a threshold β, i.e.,

$$\bar{t}=\mathbf{E}\left(\arg\min_{t}\frac{|\mathbb{V}_2|}{N}\ge\beta\right), \tag{10}$$

where $|\mathbb{V}_2|$ is the number of infected nodes. Therefore, the transmission capability based performance evaluation function can be given by:

$$f(\boldsymbol{G})=\bar{t}. \tag{11}$$

III. Strategies Analysis

A. Strategies in Small-Scale Network Systems

In this subsection, we will discuss about the solution method of our game model in small-scale network systems. Here we firstly assume that the amount of resources are integers, i.e., $A_1, A_2, a_l^i\in\mathbb{N}$ $(l=0,1,2,\ i=1,2,\dots,N)$. Hence, the action sets of both players $\mathbb{A}_1$ and $\mathbb{A}_2$ are finite, and the game becomes a zero-sum matrix game. In this matrix game, we can easily find the existence of pure strategy Nash equilibrium. Moreover, the mixed strategies of both players are represented as $\boldsymbol{s}_1=[s_1^1,s_1^2,\dots,s_1^{K_1}]$ and $\boldsymbol{s}_2=[s_2^1,s_2^2,\dots,s_2^{K_2}]$, where s_l^k $(l=1,2)$ is the probability that player l takes action $\boldsymbol{a}_l^k$, and K_1, K_2 are the size of their action sets. In solving the mixed strategy Nash equilibrium, the main problem is that the number of feasible actions is acutely increased with the augment of the number of nodes and the amount of resources. For the efficiency of finding the solution, we formulate it into a pair of mutually dual linear programming problems as:

$$\begin{aligned} \max\quad & v\\ \text{s.t.}\quad & \sum_{k_1=1}^{K_1} s_1^{k_1}\, u_1(\boldsymbol{a}_1^{k_1},\boldsymbol{a}_2^{k_2})\ge v, && k_2=1,\dots,K_2,\\ & \sum_{k_1=1}^{K_1} s_1^{k_1}=1,\\ & s_1^{k_1}\ge 0, && k_1=1,\dots,K_1,\end{aligned}\tag{12}$$

as well as

$$\begin{aligned} \min\quad & w\\ \text{s.t.}\quad & \sum_{k_2=1}^{K_2} s_2^{k_2}\, u_1(\boldsymbol{a}_1^{k_1},\boldsymbol{a}_2^{k_2})\le w, && k_1=1,\dots,K_1,\\ & \sum_{k_2=1}^{K_2} s_2^{k_2}=1,\\ & s_2^{k_2}\ge 0, && k_2=1,\dots,K_2.\end{aligned}\tag{13}$$

By solving the prime problem (12) and its dual problem (13), we get the optimal solution $(\boldsymbol{s}_1^\star, v^\star)$ and $(\boldsymbol{s}_2^\star, w^\star)$, respectively. Due to strong duality property, we have $v^\star = w^\star$, which also equal the defender's expected utility, i.e.,

$$\mathbf{E}(u_1)=u_1(\boldsymbol{s}_1^\star,\boldsymbol{s}_2^\star)=\sum_{k_1=1}^{K_1}\sum_{k_2=1}^{K_2} s_1^{k_1}\, s_2^{k_2}\, u_1(\boldsymbol{a}_1^{k_1},\boldsymbol{a}_2^{k_2}). \tag{14}$$

As for the cases that the amount of attack-defense resources are continuous value, it is still an open challenge to solve the equilibrium analytically because the utility $\mathbf{E}(u_1)$ is usually not a continuous function of $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ in our model. However, we can approximate the Nash equilibrium and the expected utility by the previous method. Therefore, we provide a general solution as Algorithm 1 which supports accurate analysis of small-scale network systems with a dozen of nodes. However, its computational complexity raises rapidly with the increase of network scale.

```
Algorithm 1: General equilibrium solution algorithm
Input  Network system $\boldsymbol{G}'$, evaluation function $f(\cdot)$;
Initialize  Set $A_1$, $A_2$ and $\boldsymbol{a}_0$ considering the real scenario,
    solving accuracy and computing complexity;
Generate the action sets $\mathbb{A}_1=\{\boldsymbol{a}_1^1,\boldsymbol{a}_1^2,\dots,\boldsymbol{a}_1^{K_1}\}$ and
    $\mathbb{A}_2=\{\boldsymbol{a}_2^1,\boldsymbol{a}_2^2,\dots,\boldsymbol{a}_2^{K_2}\}$ of both players;
Calculate the utility $u_1(\boldsymbol{a}_1^{k_1},\boldsymbol{a}_2^{k_2})$ for every pair of $\boldsymbol{a}_1^{k_1}$, $\boldsymbol{a}_2^{k_2}$ in
    $\mathbb{A}_1$, $\mathbb{A}_2$, and construct the payoff matrix;
Solve the linear programming problems in (12) and (13);
Output  Nash equilibrium strategy $(\boldsymbol{s}_1^\star,\boldsymbol{s}_2^\star)$ and the expected
    utility $\mathbf{E}(u_1)$;
```

B. Strategies in Large-Scale Network Systems

In real network systems, attackers and defenders usually have commonly used patterns for attacking and defending, which can be regarded as some commonly chosen actions. Moreover, the rational defender and attacker will only take the actions yielding high expected utility as its strategy. Therefore, in order to simplify the analysis, we can assume that the action set of the player is composed of only a small part of quality practical actions from all feasible actions, namely the *practical action set*. In order to find the practical action sets of both players, we propose a co-evolution based algorithm as Algorithm 2 inspired by the genetic algorithm [11].

Specifically, first of all, we set K_l $(l=1,2)$ as the number of actions in the action set for both players, and generate random vectors $\boldsymbol{g}_l^k=[g_l^{k(1)},g_l^{k(2)},\dots,g_l^{k(N)}]$, where

$l=1,2$, $k=1,2,\dots,K_l$, and $g_l^{k(i)} \geq 0$ $(i=1,2,\dots,N)$ are independent and identically distributed random numbers in terms of either uniform distribution in $[0,1]$, exponential distribution or others. Then, the initial random actions can be given by:

$$\boldsymbol{a}_l^k = A_l \cdot \frac{\boldsymbol{g}_l^k}{\sum_{i=1}^{N} g_l^{k(i)}}, \tag{15}$$

Here, $\boldsymbol{g}_l^k$ is also called the *chromosome* of action $\boldsymbol{a}_l^k$, and $g_l^{k(i)}$ $(i=1,2,\dots,N)$ are named *genes* that concatenate it. Moreover, action set $\mathbb{A}_l$ composed by $\boldsymbol{a}_l^k$, is a *population*, and the set of chromosomes $\mathbb{G}_l = \{\boldsymbol{g}_l^1, \boldsymbol{g}_l^2, \dots, \boldsymbol{g}_l^{K_l}\}$ is a *gene pool*. In addition, according to the rational assumption of players, in order to avoid the case that $a_2^i \leq a_0^i$, we adjust the attacker's action $\boldsymbol{a}_2^k = [a_2^{k(1)}, a_2^{k(2)}, \dots, a_2^{k(N)}]$ as:

$$a_2^{k(i)} = \begin{cases} 0, & \text{if } i \notin \mathbb{I}^k, \\ a_2^{k'(i)} \cdot \dfrac{A_2}{\sum_{i \in \mathbb{I}^k} a_2^{k'(i)}}, & \text{if } i \in \mathbb{I}^k, \end{cases} \tag{16}$$

where $a_2^{k'(i)}$ is the attacker's original allocated resources generated by (15), while $\mathbb{I}^k$ is the set of node's index i that satisfies $a_2^{k'(i)} > a_0^i$. Given both players' initial action sets $\mathbb{A}_l = \{\boldsymbol{a}_l^1, \boldsymbol{a}_l^2, \dots, \boldsymbol{a}_l^{K_l}\}$, according to (2), the average utility of each action is:

$$\overline{u_1^{k_1}} = \frac{1}{K_2} \sum_{k_2=1}^{K_2} u_1(\boldsymbol{a}_1^{k_1}, \boldsymbol{a}_2^{k_2}), \tag{17}$$

$$\overline{u_2^{k_2}} = \frac{1}{K_1} \sum_{k_1=1}^{K_1} u_2(\boldsymbol{a}_1^{k_1}, \boldsymbol{a}_2^{k_2}). \tag{18}$$

Thus, we can obtain $\overline{\boldsymbol{u}_l} = [\overline{u_l^1}, \overline{u_l^2}, \dots, \overline{u_l^{K_l}}]$. Then, we can generate the child actions of the next generation. In our algorithm, $\mu_l K_l$ actions with the highest utility will be directly added into the action set of the next generation, where μ_l is the proportion. The remaining actions are generated by crossover and mutation. In the process of crossover, we first select two parent chromosomes $\boldsymbol{g}_l^x$ and $\boldsymbol{g}_l^y$ with certain probability as:

$$p_l^k = \frac{2(K_l + 1 - h_l^k)}{K_l(K_l+1)}, \tag{19}$$

where h_l^k is the rank of the average utility of action $\boldsymbol{a}_l^k$ in $\mathbb{A}_l$ in descending order. Then we randomly choose half of the genes that inherited from $\boldsymbol{g}_l^x$, and compose node set $\mathbb{V}_x$. The remaining nodes compose $\mathbb{V}_y = \mathbb{V} \setminus \mathbb{V}_x$, which indicates the the genes inherited from $\boldsymbol{g}_l^y$. Hence, we can denote the crossover gene as:

$$g_l^{k'(i)} = \begin{cases} g_l^{x(i)}, & \text{if } v_i \in \mathbb{V}_x, \\ g_l^{y(i)}, & \text{if } v_i \in \mathbb{V}_y. \end{cases} \tag{20}$$

Finally, in order to increase the diversity of the child population, each gene $g_l^{k'(i)}$ mutates with probability γ_l, i.e.,

$$g_l^{k(i)} = \begin{cases} g^{\text{rand}}, & \text{with probability } \gamma_l, \\ g_l^{k'(i)}, & \text{with probability } 1-\gamma_l, \end{cases} \tag{21}$$

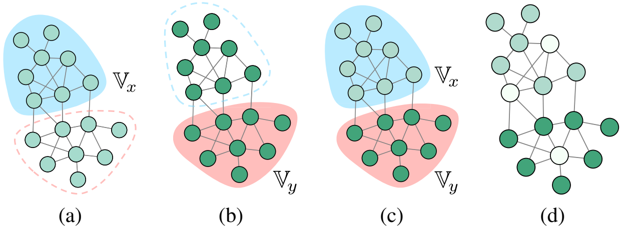


Fig. 2. The process of generating child chromosome $\boldsymbol{g}_l^k$ from parent chromosomes $\boldsymbol{g}_l^x$ and $\boldsymbol{g}_l^y$. (a) parent chromosome $\boldsymbol{g}_l^x$. (b) parent chromosome $\boldsymbol{g}_l^y$. (c) crossover chromosome $\boldsymbol{g}_l^{k'}$. (d) child chromosome $\boldsymbol{g}_l^k$.

Algorithm 2: Co-evolutionary equilibrium solution algorithm based on practical action sets

```
Input  Network system $G'$, evaluation function $f(\cdot)$, number
  of iterations $T$, resources $A_1$, $A_2$, sizes of action sets $K_1$,
  $K_2$, proportion of actions inherited directly $\mu_1$, $\mu_2$, mutation
  probability $\gamma_1$, $\gamma_2$;
Initialize  Generate chromosomes $g_l^k$ randomly to build initial
  gene pools $\mathbb{G}_1$, $\mathbb{G}_2$ and action sets $\mathbb{A}_1$, $\mathbb{A}_2$ as (15), (16);
for $t = 1,2,\dots,T$ do
    Calculate the utility for every pair of actions in $\mathbb{A}_1$ and $\mathbb{A}_2$
      according to $f(\cdot)$ as (2), and calculate the average utility of
      every action $\overline{u_1}$, $\overline{u_2}$ as (17), (18);
    Calculate the probability that being selected in crossover
      $p_1 = [p_1^1, p_1^2, \dots, p_1^{K_1}]$, $p_2 = [p_2^1, p_2^2, \dots, p_2^{K_2}]$ for every
      action in $\mathbb{A}_1$ and $\mathbb{A}_2$ as (19);
    for $l = 1,2$ do
        Generate empty gene pool $\mathbb{G}_l' = \emptyset$ and action set $\mathbb{A}_l' = \emptyset$;
        for $k = 1,2,\dots,K_l$ do
            if $k \leq \mu_l K_l$ then
                Choose the action with $k$th highest utility in $\mathbb{A}_l$
                  according to $\overline{u_l}$, and add it to $\mathbb{A}_l'$ as $a_l^k$;
            else
                Choose parent chromosomes $g_l^x$, $g_l^y$ according to $p_l$;
                Divide the node index set $\mathbb{V}$ into $\mathbb{V}_x$ and $\mathbb{V}_y$;
                Generate crossover chromosome $g_l^{k'}$ by $\mathbb{V}_x$, $\mathbb{V}_y$ and
                  $g_l^x$, $g_l^y$ as (20);
                Generate child chromosome $g_l^k$ from $g_l^{k'}$ as (21) with
                  mutation probability $\gamma_l$, and add it to $\mathbb{G}_l'$;
                Generate child action $a_l^k$ as (15), (16) and add to $\mathbb{A}_l'$;
        Update $\mathbb{G}_l = \mathbb{G}_l'$ and $\mathbb{A}_l = \mathbb{A}_l'$;
Calculate the utility $u_1(a_1^{k_1}, a_2^{k_2})$ for every pair of $a_1^{k_1}$, $a_2^{k_2}$ in
  practical action sets $\mathbb{A}_1$, $\mathbb{A}_2$;
Solve the linear programming problems in (12) and (13);
Output  Nash equilibrium strategy $(s_1^\star, s_2^\star)$ and the expected
  utility $\mathbf{E}(u_1)$;
```

where g^{rand} is a random value with the same distribution with the initial genes. Hence, we can obtain child chromosome $\boldsymbol{g}_l^k$ composed by genes $g_l^{k(i)}$. Then we generate child action according to (15), and add it to the action set of the next generation. Fig. 2 illustrates the process of crossover and mutation. Lastly, we can solve the mixed strategy Nash equilibrium based on the practical action sets.

TABLE I
APPLICATIONS OF THE GAME MODEL

Scenarios	**Internet security**	**Communication timeliness of IoV**	**Rumor spread control**
Network system	computer networks	Internet of vehicles (IoV)	online social networks
Nodes	Internet autonomous systems	taxis	Sina Weibo[1] users
Edges	network routes	wireless connections	friend relationships
Defenders	Internet administrators	network schedulers	opinion supervisors
Attackers	hackers	interferers	rumormongers
Defenders' action	protect network devices by installing firewalls, upgrading hardware and software	enhance anti-jamming capacity of devices by increasing transmitting power	increase the resistance and discernment to rumors of social network users by opinion supervision
Attackers' action	attack key network devices by DDoS, identity spoofing, malicious intrusion	interferes the communication of vehicles on specific location by jamming	spread rumors to users and turn them into initial rumor disseminators
Resources	maintenance budget, computing resources	power consumption, devices budget	supervisory capacity, dissemination capacity
Primary goals	protect / break up the Internet connectivity	anti-interfere / interfere with communication timeliness	suppress / promote the spread process of rumors
Game rule	Eq. (3)	Eq. (22)	Eq. (9)
Performance metric	$f(\boldsymbol{G}) = n$	$f(\boldsymbol{G}) = -\bar{r}$	$f(\boldsymbol{G}) = \bar{t}$
Original dataset	University of Oregon Route Views Project [2]	Beijing Taxi GPS Dataset in T-Drive Project [12]	Microblog PCU Dataset in UCI ML Repository [3]
Number of nodes	300	125	279
Number of edges	400	425	313
Degree distribution	power-law	homogeneous	power-law
Network features	free-scale, small-world	ring network	free-scale, small-world

[1] https://weibo.com/ [2] http://www-personal.umich.edu/~mejn/netdata/ [3] http://archive.ics.uci.edu/ml

IV. APPLICATIONS AND SIMULATIONS

In this section, we will introduce several applications of our game model in realistic scenarios based on real-world dataset. Specifically, we select three scenarios, i.e., Internet security, communication timeliness of wireless vehicular networks and rumor spread control in online social networks, which correspond three proposed evaluation metrics. Firstly, we give an overview of the game models and the network systems of these applications in Table I. Moreover, the network topologies can be found in Fig. 3. Because these real network systems are large-scale, our analysis is mainly based on the co-evolution algorithm. In all the simulations, we set the parameters of the algorithm as $K_1 = K_2 = 50$, $\mu_1 = \mu_2 = 0.4$, and $\gamma_1 = \gamma_2 = 0.15$. Moreover, after iterations in the co-evolution process, we select ten actions with the highest average utility as the practical action set for each player and solve the mixed Nash equilibrium strategies. In order to show the strategies of both players visualized, we calculate the expected resources allocated on each node by taking the weighted average of each action according to the mixed Nash equilibrium, and the results in the case of $A_1 = A_2 = 100$ are illustrated in Fig. 3.

In addition, Fig. 4 provides the relationships between the resources A_1, A_2 and expected utility $\mathbf{E}(u_1)$. Moreover, the blue mesh on it illustrates the expected utility $\mathbf{E}(u_1)$ where the attacker's action set are randomly generated and the defender's action set is still generated by the co-evolution algorithm. Similarly, the red mesh shows $\mathbf{E}(u_1)$ where the defender's action set are randomly generated. We can find that the practical action set generated by the co-evolution algorithm overwhelms the randomly generated action set, which reveals the effectiveness and validity of our proposed algorithm. The following is the further explanation and discussion.

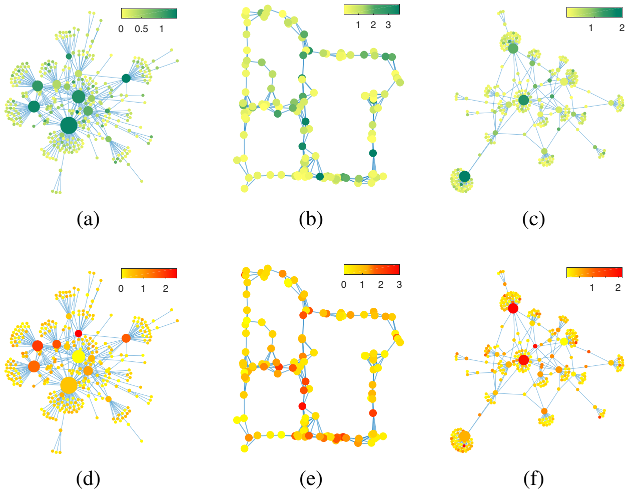
A. Internet Security

In Internet attack-defense confrontation, we assume that the network devices occupied by the attacker will break down, and hence the weight of these nodes' neighboring edges will become zero, which can be represented as (3). In addition, we set nodes' self-defense capacity $a_0^i = 0.01 \cdot d_i$, which is proportional to the nodes' degree. According to the simulation results shown in Fig. 3a and Fig. 3d, the expected utility of the defender is $\mathbf{E}(u_1) = -198.5$ when $A_1 = A_2 = 100$. Hence, there are about 200 network nodes separated from the Internet backbone under given parameters. In fact, because there are a few nodes with large degree and there exist hierarchical structures, this network is highly vulnerable to targeted attacks. Moreover, the attacker tends to allocate much resources on nodes with high degree, which makes their neighboring nodes separated from the giant component, as well as on nodes with high centrality to make the whole network collapse.

B. Communication Timeliness of Vehicular Networks

In Internet of Vehicles (IoV), vehicles usually transmit information through multi-hop communications, and hence the timeliness is a key problem. However, in open wireless communication environment, malicious attackers can interfere the communication of vehicle devices through jamming. In simulation, we assume that the maximum communication distance of vehicles is 250 meters and all nodes' self-defense capacity $a_0^i = 0.1$. The delay of communication links is represented by the weight of edges. Because malicious interference will cause serious decline of data rate, we assume that:

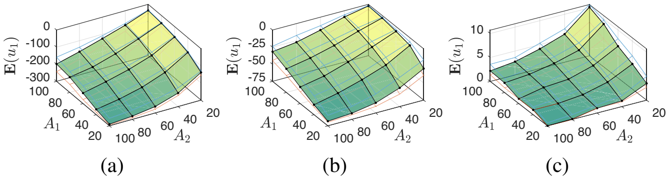
$$w_{ij} = \begin{cases} 1, & \text{if } \{v_i, v_j\} \in \mathbb{E}_{11}, \\ 10, & \text{if } \{v_i, v_j\} \in \mathbb{E}_{12} \cup \mathbb{E}_{22}, \end{cases} \tag{22}$$



(a) (b) (c)

(d) (e) (f)

Fig. 3. Expected resource allocation of both players. (The top three subfigures are for the defenders, and the following three are for the attackers.)



(a) (b) (c)

Fig. 4. Expected utility of the defender under different A_1 and A_2.

which indicates that the delay of the succeed interfered devices will increase tenfold. As shown in Fig. 3b and Fig. 3e, both players tend to allocate more resources on the nodes with high centrality. In particular, the gateway nodes, which are the nodes must be passed in numerous shortest paths, play an important role. This is mainly because when a regular node is controlled by the attacker, there still exist other short paths. However, if a gateway node is controlled, data transmission has only to suffer huge transmission delay by passing this interfered node or detouring to another street. Moreover, increasing the density of vehicles or increasing vehicles' maximum communication distance will create more intensive links between vehicles, which is beneficial for improving the anti-interfere capacity and timeliness of communication.

C. Rumor Spread Control in Online Social Networks

Now we analyze the rumor spreading in a small community of Sina Weibo, which is a popular online social network in China. Here the undirected edge in the network represents the relationship of "friends", i.e., "following each other", of two users. According to the characteristics of social networks, we use the betweenness centrality to denote the influence c_k of node v_k in transmission. Moreover, we set the threshold proportion $\beta = 0.8$ and node's self-defense capacity $a_0^i = 0.01 \cdot d_i$. According to Fig. 3c and Fig. 3f, two players mainly focus on two kinds of nodes. One is the nodes with high influence. The other is the hub nodes connecting the small sub-communities, which also play critical roles in rumor spread. Moreover, in Fig. 4c, we simply set $f(\boldsymbol{G}') = 0$ for the convenience of elaborating. Hence the social network of friends has strong transmission capacity, and it is difficult for the defender to suppress the emergence and spread of rumors unless he/she has much more resources than the attacker.

V. Conclusions

In this paper, we modeled the attack-defense resource allocation in network systems as a networked zero-sum Colonel Blotto game. In contrast to the traditional Colonel Blotto game model, our proposed game broadens the application fields of this kind of resource allocation games. Moreover, we proposed three kinds of network performance metrics based on network connectivity, average path length and transmission capacity, respectively. Furthermore, the co-evolution based algorithm for obtaining the Nash equilibrium strategies based on practical action sets improved the feasibility of strategies analysis. Sufficient simulations based on three real-world network systems proved the effectiveness of our model.

Acknowledgement

This work was funded by Tsinghua National Laboratory for Information Science and Technology (TNList) Cross-discipline Foundation, and also supported by the China Equipment Development Program (Grant No. 6141A02022615).

References

[1] L. Wang, S. Ren, B. Korel, K. A. Kwiat, and E. Salerno, "Improving system reliability against rational attacks under given resources," *IEEE Transactions on Systems, Man & Cybernetics: Systems*, vol. 44, no. 4, pp. 446–456, Apr. 2014.

[2] C. Jiang, Y. Chen, Y. Gao, and K. R. Liu, "Joint spectrum sensing and access evolutionary game in cognitive radio networks," *IEEE Transactions on Wireless Communications*, vol. 12, no. 5, pp. 2470–2483, May. 2013.

[3] J. Wang, C. Jiang, Z. Han, T. Q. S. Quek, and Y. Ren, "Private information diffusion control in cyber physical systems: A game theory perspective," in *IEEE International Conference on Computer Communication and Networks (ICCCN)*, Vancouver, Canada, Mar. 2017.

[4] C. Jiang, Y. Chen, K. Liu, and Y. Ren, "Renewal-theoretical dynamic spectrum access in cognitive radio networks with unknown primary behavior," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 3, pp. 406–416, Mar. 2013.

[5] E. Borel, "The theory of play and integral equations with skew symmetric kernels," *Econometrica: Journal of the Econometric Society*, vol. 21, no. 1, pp. 97–100, Jan. 1953.

[6] B. Roberson, "The Colonel Blotto game," *Economic Theory*, vol. 29, no. 1, pp. 1–24, Jan. 2006.

[7] P. H. Chia and J. Chuang, "Colonel Blotto in the phishing war," in *International Conference on Decision and Game Theory for Security*, College Park, MD, Nov. 2011, pp. 201–218.

[8] Y. Wu, B. Wang, and K. R. Liu, "Optimal power allocation strategy against jamming attacks using the Colonel Blotto game," in *IEEE Global Telecommunications Conference (GLOBECOM)*, Honolulu, HI, Dec. 2009.

[9] Z. E. Fuchs and P. P. Khargonekar, "A sequential Colonel Blotto game with a sensor network," in *IEEE American Control Conference (ACC)*, Montréal, Canada, Jun 2012, pp. 1851–1857.

[10] M. Hajimirsadeghi, G. Sridharan, W. Saad, and N. B. Mandayam, "Internetwork dynamic spectrum allocation via a Colonel Blotto game," in *Annual IEEE Conference on Information Science and Systems (CISS)*, Princeton, NJ, Mar. 2016, pp. 252–257.

[11] D. Whitley, "A genetic algorithm tutorial," *Statistics and Computing*, vol. 4, no. 2, pp. 65–85, Jun. 1994.

[12] J. Yuan, Y. Zheng, X. Xie, and G. Sun, "T–Drive: Enhancing driving directions with taxi drivers' intelligence," *IEEE Transactions on Knowledge & Data Engineering*, vol. 25, no. 1, pp. 220–232, Jan. 2013.