# Coalitional Security Games

Qingyu Guo[1], Bo An[2], Yevgeniy Vorobeychik[3], Long Tran-Thanh[4], Jiarui Gan[2], Chunyan Miao[2]
[1]Joint NTU-UBC Research Centre of Excellence in Active Living for the Elderly, NTU, Singapore
[2]School of Computer Engineering, Nanyang Technological University, Singapore
[3]Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN
[4]Electronics and Computer Science, University of Southampton, UK
[1,2]{qguo005,boan,jrgan,ascymiao}@ntu.edu.sg,[3]yevgeniy.vorobeychik@vanderbilt.edu,[4]ltt08r@ecs.soton.ac.uk

## ABSTRACT

Game theoretic models of security, and associated computational methods, have emerged as critical components of security posture across a broad array of domains, including airport security and coast guard. These approaches consider terrorists as motivated but independent entities. There is, however, increasing evidence that attackers, be it terrorists or cyber attackers, communicate extensively and form coalitions that can dramatically increase their ability to achieve malicious goals. To date, such cooperative decision making among attackers has been ignored in the security games literature. To address the issue of cooperation among attackers, we introduce a novel *coalitional security game (CSG)* model. A CSG consists of a set of attackers connected by a (communication or trust) network who can form coalitions as connected subgraphs of this network so as to attack a collection of targets. A defender in a CSG can delete a set of edges, incurring a cost for deleting each edge, with the goal of optimally limiting the attackers' ability to form effective coalitions (in terms of successfully attacking high value targets). We first show that a CSG is, in general, hard to approximate. Nevertheless, we develop a novel branch and price algorithm, leveraging a combination of column generation, relaxation, greedy approximation, and stabilization methods to enable scalable high-quality approximations of CSG solutions on realistic problem instances.

## Keywords

Game Theory; Security; Optimization; Stackelberg Games

## 1. INTRODUCTION

Recent decades have seen a number of major terrorist attacks, such as WTC 9/11 attack, Jemaah Islamiyahs Bali bombing, and 7/7 London bombing, that have killed thousands of lives and caused significant economic losses. An important reason for the increasing threat of terrorism is cooperation between terrorist groups [22]. For example, three terrorist groups in Africa have been reported to share funds, training, and explosive materials with each other [27], and Chechen terrorists were reported to obtain weapons from terrorist organizations in the Middle East [19]. Such sharing of skills and resources among terrorist groups is common because it significantly increases their capability of achieving malicious goals, such as attacking high-value targets.

An important way to prevent individual terrorist groups from forming powerful coalitions is to cut off connections between them. This can be done by blocking their bank accounts, increasing surveillance of strategic exchange points, setting sentries in arterial roads, etc. However, since it is impractical to block all possible ways attackers can communicate, a central decision problem is to choose a subset of such connections to block so as to minimize expected efficacy of formed coalitions and resulting attacks. Addressing this problem entails several challenges: i) the set of possible combinations of edges to cut is exponential in the number of connections among attackers; ii) the number of possible coalitions to account for is exponential in the number of attackers; iii) coalition stability is an important consideration in assessing which coalitions will form, and it significantly increases problem difficulty; iv) the decision space of attackers includes the choice of targets to attack, which must also be accounted for by the defender.

While there is existing research on terrorist networks, it has been limited in scope to either social network analysis of terrorist groups [13, 29, 21, 12], or using cooperative game theory as a means for identifying key members of terrorist networks [17, 18]. The related research in security games, on the other hand, tends to model attackers as independent actors (indeed, only a single attack by a single attacker or group is typically considered) [30, 11, 31, 15, 32, 35, 10, 34, 33]. The ability and proclivity of attackers to form coalitions is thus largely ignored within the security games research.

To address the problem of optimally inhibiting formation of attack coalitions, we introduce a formal Coalitional Security Game (CSG) model. We show that the associated problem is MAX SNP-hard for the defender, indicating no polynomial time approximation scheme unless P=NP, and the decision problem is NP-hard for the attackers. To overcome the computational challenges, we develop a sophisticated branch and price algorithm, involving a novel combination of column generation and linear programming relaxation. Since the slave problem of column generation is formulated as a bilevel mixed-integer linear program (BMILP), we further improve the performance of the algorithm by using a novel linear relaxation approximation to reformulate the slave problem as an easily solvable single level MILP with formal constant factor approximation guarantee. Furthermore, we provide an interior-point stabilization to improve convergence properties of column generation by generating

**Appears in:** *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016), J. Thangarajah, K. Tuyls, S. Marsella, C. Jonker (eds.), May 9–13, 2016, Singapore.*

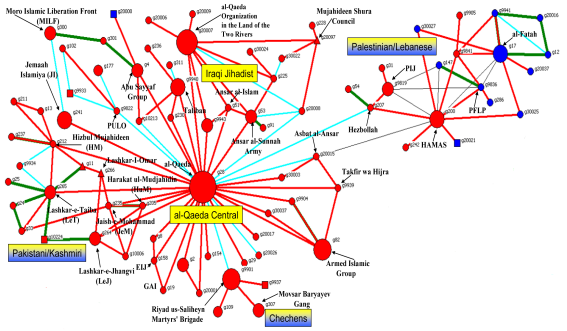



**Figure 1: Mideast Terrorist Network [1].**

an interior dual optimal solution of the master problem, and novel heuristics for generating multiple columns in each iteration to support the linear relaxation approximation and further speed up the column generation. Extensive experimental results show that our algorithms 1) provide orders of magnitude improvement in speed over the exact integer programming solution; 2) enable scalable high-quality approximation solutions of realistic CSG problem instances, outperforming existing classic heuristic search algorithms significantly; and 3) are remarkably robust against uncertainties of attackers' payoffs.

## 2. MOTIVATING DOMAIN

Cooperation among terrorist groups is common and necessary for their survival. During the past few decades, almost half of terrorist groups have had an ally [22]. Various kinds of cooperation are found among them, including transferring funds, weapons support, training, and other sharing of critical skills and resources. For example, the Al-Taqwa banking system financed the activities of multiple terrorist organizations, including Hamas [16]. Levitt [16] pointed out that there is significant overlap and cooperation between Palestinian terrorist groups like Hamas and other groups, such as al-Qaeda in the area of terrorist financing and logistical support. In 2000, Al-Qaeda held training camps which served as open universities, educating terrorists from a wide array of local and international terrorist groups [24]. Terrorist groups also cooperate with each other to coordinate attacks. For example, the Popular Resistance Committee (PRC) is a conglomeration of members of Islamic Jihad, Hamas and the various terrorist groups, and has conducted several infamous terrorist attacks, including the roadside bombing attack in February 14, 2002 [4].

Since cooperation among terrorist groups is achieved through communication channels, such as front companies, charities for transferring funds [16], and transportation hubs including roadways, ports, and rivers for weapon supply, training, and coordinated attacks, an important measure for preventing terrorist groups from forming coalitions is to cut off connections among them. For example, in 2003, FBI Assistant Director John Pistole testified to Congress that investigations into the financial activities of terrorist supporters in the United States helped prevent four different terrorist attacks abroad [23]. Moreover, the US strategy for combating terrorism asserts that *"The interconnected nature of terrorist organizations necessitates that we pursue them across the geographic spectrum to ensure that all linkages between the strong and the weak organizations are broken, leaving each of them isolated, exposed, and vulnerable to defeat"* [6].

In order to cut connections among terrorist groups, security agencies can shut down the front companies and charities, which are providing and transferring money for terrorist groups, to stem the flow of funds between terrorist groups [16]. Besides, blocking critical transportation hubs can significantly reduce efficiency of coordination attempts such as weapon support and terrorist training. Nevertheless, activities aimed at inhibiting communication among attackers, such as blocking roadways by setting sentries are costly, and it is infeasible to cut all the connections among terrorist groups. This motivates our investigation of optimal allocation of security resources to block attacker linkages, taking into account the associated costs, as well as the nature of coalitions that would form as a result.

## 3. COALITIONAL SECURITY GAMES

A *coalitional security game* (CSG) is modeled based on the *coalitional skill game* [2], and consists of $T$ types of targets, each with a large number of copies, which is reasonable since there is a large pool of potential targets of terrorism around the world. There is a set $N$ of terrorists (individuals or groups) who want to attack these targets. Each terrorist $i$ has a set $S_i$ of skills, and attacking a target of type $t \in T$ requires a set $S(t)$ of skills. Let $S = \bigcup_{i \in N} S_i$. We assume $S(t) \subseteq S$ for all $t \in T$ without loss of generality. Each type $t \in T$ has a value $p_t > 0$ for the terrorists. Typically, each terrorist $i$ has a capacity $m_{is} \in \mathbb{N}$ for each of his skills $s \in S_i$, so that he can use $s$ in at most $m_{is}$ attacks. For example, a terrorist can provide weapons for a certain number of terrorist attacks.

The terrorists form coalitions to share skills to launch more attacks. Similar to existing work [18], we use a graph $G = (N, E)$ to represent the cooperation network of terrorists, where $N$ is the set of terrorists and $E$ is the set of edges representing connections between pairs of terrorists. We denote by $(i, j)$ an edge connecting terrorists $i$ and $j$. A set $C \subseteq N$ of terrorists can form into a coalition only when the *induced subgraph* $G_C = (C, E(C))$ is connected, where $E(C) = \{(i, j) \in E | i \in C, j \in C\}$. We use the set $C$ to represent the coalition formed.

A terrorist coalition can choose multiple targets to attack simultaneously, so long as it possesses sufficient required skills and resources. For example, in 2008, a group of terrorists carried out a series of twelve coordinated shooting and bombing attacks in Mumbai [14], and in the September 11 attack, 4 airplanes were hijacked to attack several targets in 3 different cities. We model these types of threats through the definition of the value of a coalition. Formally, let $\mathbf{a} = \langle a_t \rangle$ be an attacking plan of a coalition $C$, where $a_t$ is the number of targets of type $t$ that coalition $C$ plans to attack. An attacking plan $\mathbf{a}$ is feasible if

$$\sum_{t \in T,\, s \in S(t)} a_t \leq \sum_{i \in C,\, s \in S_i} m_{is},\ \forall s \in S, \qquad (1)$$

The payoff $u(\mathbf{a})$ for an attacking plan $\mathbf{a}$ is the sum of values of all targets attacked, i.e., $u(\mathbf{a}) = \sum_t a_t p_t$. We assume that terrorists are utility-maximizers. Thus, once a coalition is formed, they choose an attack plan which maximizes payoff over all feasible plans.

$$v(C) = \max_{\mathbf{a}:\ \text{satisfying Eq.(1)}} u(\mathbf{a}). \qquad (2)$$

The value $v(C)$ of a coalition $C$ is defined as the maximum achievable payoff in Eq.(2). We use a payoff vector $\mathbf{y} = \langle y_i \rangle \geq \mathbf{0}$ to denote the payoff for each terrorist $i$. A payoff vector represents how the value of every coalition is divided among their members, so that for a coalition structure $CS$ (i.e., a partition of $N$ into disjoint coalitions) we require $\sum_{i \in C} y_i \leq v(C)$ for any $C \in CS$. The pair $(CS, \mathbf{y})$ is called an *outcome* of a coalitional game.

## 3.1 Stable Coalition Structure

Since the terrorists are self-interested and profit driven, we assume that a coalition structure $CS$ formed by attackers must be stable in the sense that any subset of attackers has no (or little) incentive to break off into another coalition for higher payoff. To enforce coalition structure stability, we adopt the widely-used solution concept of $\epsilon$-*core* [7].

DEFINITION 1 ($\epsilon$-**core**). *The $\epsilon$-core, for $\epsilon > 0$, is the set of all outcomes $(CS, \mathbf{y})$ such that for any coalition $C \subseteq N$, $\sum_{i \in C} y_i \geq v(C) - \epsilon$.*

Attackers always prefer the outcome in $\epsilon$-core with as lower $\epsilon$ value as possible, and the minimal value $\epsilon^*$ for which the $\epsilon$-core is non-empty is called the *least-core value* of the game, with the corresponding $\epsilon^*$-core called the *least-core*. For a coalition structure $CS$, if there exists an outcome $(CS, \mathbf{y})$ in the least-core, we call $CS$ a *stable coalition structure*. Although computing the least-core is extremely hard for general coalitional games [7], given the coalition value function $v$ defined in Eq.(2), the coalitional game turns out to be *superadditive* (LEMMA 1). In this case, the attackers are willing to form as large coalitions as they can, and the coalition structure $CS^*$, whose induced subgraphs are all connected components, is a stable coalition structure (LEMMA 2).

LEMMA 1. *Given the value function defined in Eq.(2), the terrorists' coalitional game $G = (N, v)$ is superadditive, i.e., $v(C \cup D) \geq v(C) + v(D)$ for every pair of disjoint coalitions $C, D \subseteq N$.*

PROOF. For any pair of disjoint coalitions $C, D \subseteq N$, let $\mathbf{a}^C$ and $\mathbf{a}^D$ be the corresponding optimal attacking plans, and $v(C) = \sum_t a_t^C p_t$ and $v(D) = \sum_t a_t^D p_t$. For coalition $C \cup D$, the attacking plan $\mathbf{a}^{C \cup D}$, where $a_t^{C \cup D} = a_t^C + a_t^D$, is feasible since $\sum_{t \in T,\, s \in S(t)} (a_t^C + a_t^D) \leq \sum_{i \in C \cup D,\, s \in S_i} m_{is}$ $\forall s \in S$. Therefore, $v(C \cup D) \geq v(C) + v(D)$. □

LEMMA 2. *The coalition structure consisting of all coalitions whose induced subgraphs are connected components of graph $G(N, E)$ is a stable coalition structure, and it has the maximum total value among all coalition structures.*

PROOF. Since a coalition cannot be formed by terrorists in different connected components of $G$, we can consider each connected component independently.

Let $G' = (N', E')$ be a connected component of $G$. Let $(CS, \mathbf{y})$ be an outcome of $G'$ in the least-core. Now if all terrorists in $G'$ forms into one coalition $N'$ (i.e., the grand coalition whose induced subgraph is $G'$), we have $v(N') \geq \sum_{C \in CS} v(C)$ due to superadditivity. Therefore, we obtain a new outcome $(CS^*, \mathbf{y}^*) = (\{N'\}, \mathbf{y} + \frac{v(N') - \sum_{C \in CS} v(C)}{|N'|})$, such that $\mathbf{y}^* \succeq \mathbf{y}$ and $\sum_{i \in C} y_i^* \geq \sum_{i \in C} y_i \geq v(C) - \epsilon^* > 0$ for all $C \subseteq N'$. This means $(CS^*, \mathbf{y}^*)$ is in the least-core of $G'$. Obviously, $CS^*$ has the maximum value since $v(N') \geq \sum_{C \in CS} v(C)$ holds for arbitrary $CS$. Applying the result to all connected components, we can obtain a stable coalition structure consisting of all coalitions whose induced subgraphs are connected components of $G$. □

## 3.2 Defender Strategy

The defender's goal is to optimally cut off connections within terrorists to minimize threat due to attacks by formed coalitions. We use a symmetric matrix $\mathbf{B} = \langle B_{ij} \rangle$ to represent the defender's strategy, such that $B_{ij} = 1$ if edge $(i, j)$ is blocked and $B_{ij} = 0$ otherwise. We let $B_{ii} = 0$ for all $i \in N$. Blocking an edge $(i, j)$ incurs a cost, $\lambda_{ij}$. When the defender adopts strategy $\mathbf{B}$, the blocked edges are removed from the network, resulting in a new network $G(\mathbf{B}) = (N, E(\mathbf{B}))$ where $E(\mathbf{B}) = E \backslash \{(i, j) \in E | B_{ij} = 1\}$. Given $\mathbf{B}$, the attackers play a coalitional skill game on the induced graph $G(\mathbf{B}) = (N, E(\mathbf{B}))$, and form a coalition structure $CS^*(\mathbf{B})$ consisting of all coalitions whose induced subgraphs are connected components of $G(\mathbf{B})$. The defender's utility is then

$$U_d(\mathbf{B}) = -\sum\nolimits_{C \in CS^*(\mathbf{B})} v(C) - \sum\nolimits_{(i,j) \in E} B_{ij} \lambda_{ij}$$

# 4. COMPLEXITY ANALYSIS

We first investigate the computational complexity of finding the optimal strategies for both the defender and the attackers, and show that the defender's and the attackers' decision-making problems are MAX SNP-hard and NP-hard respectively.

## 4.1 Defender's Decision-Making Problem

We reduce the *k-TERMINAL CUT* problem [8], whose MAX SNP-hardness has been proved for any fixed $k \geq 3$, to the defender's decision-making problem by a *linear reduction* which preserves the approximation property (MAX SNP-hardness) [20]. The following definitions will be useful to introduce our key result (THEOREM 1).

DEFINITION 2 (*k*-**TERMINAL CUT**). *Given a graph $G = (N, E)$, a set $H = \{h_1, \cdots, h_k\}$ of $k$ specified vertices or terminals, and a positive weight $w_{ij}$ for each edge $(i, j) \in E$, find a minimum weight set of edges $E^* \subseteq E$ such that the removal of $E^*$ from $E$ disconnects each terminal from all the other terminals.*

DEFINITION 3 (**Linear reduction**). *Let $A$ and $B$ be two optimization problems (either maximization or minimization). We say that $A$ linearly reduces to $B$ if there are two polynomial time algorithms $f$ and $g$ and constants $\alpha$, $\beta > 0$ such that:*

1. *Given an instance $a$ of $A$, algorithm $f$ produces an instance $b = f(a)$ of $B$ such that the cost (objective value) of an optimal solution for $b$, $opt(b)$, is at most $\alpha \cdot opt(a)$.*
2. *Given $a$, $b = f(a)$, and any solution $y$ of $b$, algorithm $g$ produces a solution $x$ of $a$ such that $|cost(x) - opt(a)| \leq \beta |cost(y) - opt(b)|$, where $cost(x)$ and $cost(y)$ refer to the objective values of $x$ and $y$.*

THEOREM 1. *Finding an optimal defender strategy for a coalitional security game is MAX SNP-hard.*

PROOF. W.l.o.g, we apply linear reduction to reduce 3-TERMINAL CUT problem to the defender's decision-making problem. Let $a$ and $b$ be instances of 3-TERMINAL

CUT and defender's decision-making problem respectively. Let $E'$ and $\mathbf{B}$ be solutions for $a$ and $b$ correspondingly, and $opt(a)$, $opt(b)$ are costs of the optimal solutions for $a$ and $b$ respectively, i.e., the costs of optimal cuts $E^*$ and optimal defender strategy $\mathbf{B}^*$. We first present a polynomial time algorithm $f$ to reduce 3-TERMINAL CUT to defender's decision-making problem. Here for any input instance $a = \langle G, H, w\rangle$, $f$ outputs an instance $b$ of defender's decision-making problem with the following components.

1. $G$ as the cooperation network, and $N$ (nodes of $G$) as the set of terrorists. The blocking cost $\lambda_{ij}$ of edge $(i,j) \in E$ equals to $w_{ij}$.

2. $S = \{s_1, s_2, s_3, s_4\}$ as the skill set; $S_{h_1} = \{s_1, s_4\}$, $S_{h_2} = \{s_2, s_4\}$ and $S_{h_3} = \{s_3, s_4\}$ as skills for terrorists $h_1$, $h_2$ and $h_3$ who are referred to as the *terminal attackers*; $S_i = \{s_4\}$ as skill for all the other terrorists $i \notin \{h_1, h_2, h_3\}$; and $m_{is} = 1$ for all $i \in N$, $s \in S$.

3. A set $T$ of four target types $\{t_1, t_2, t_3, t_4\}$ with $S(t_1) = \{s_1, s_2, s_4\}$, $S(t_2) = \{s_1, s_3, s_4\}$, $S(t_3) = \{s_2, s_3, s_4\}$ and $S(t_4) = \{s_1, s_2, s_3, s_4\}$; $p_1 = p_2 = p_3 = M > W_{max} \cdot |E|$ and $p_4 = 2M$ where $W_{max} = \max_{(i,j)\in E} \lambda_{ij}$.

Obviously, $f$ reduces $a$ to $b$ in polynomial time. For a given defender strategy $\mathbf{B}$, the cost (defender's total loss) is $cost(\mathbf{B}) = \sum_{(i,j)\in E} w_{ij}B_{ij} + \widetilde{F}_{\mathbf{B}}$, where $\widetilde{F}_{\mathbf{B}} = 0$ if any pair of terminal attackers are not connected under $\mathbf{B}$, $\widetilde{F}_{\mathbf{B}} = M$ if any two terminal attackers are connected while the rest one is disconnected to both of them, and $\widetilde{F}_{\mathbf{B}} = 2M$ if all three terminals are connected. Then we can have a polynomial time algorithm $g$, which simply constructs a solution $E' = \{(i,j)|B_{ij} = 1\}$ for $a$ from the solution $\mathbf{B}$ of $b = f(a)$.

Next, we will show that there exists a constant $\alpha$ such that $opt(b)$, is at most $\alpha \cdot opt(a)$. Let the optimal solution of $a$ be $E^*$, i.e., $opt(a) = \sum_{(i,j)\in E^*} w_{ij}$. Let $\mathbf{B}^*$ be a solution of $b$, such that $B^*_{ij} = 1$ if $(i,j) \in E^*$, and $B^*_{ij} = 0$ otherwise. We show by contradiction that $\mathbf{B}^*$ is optimal for $b$.

Suppose the optimal defender strategy is another one $\mathbf{B}$. We have $cost(\mathbf{B}^*) - cost(\mathbf{B}) = \sum_{(i,j)\in E} w_{ij}(B^*_{ij} - B_{ij}) - \widetilde{F}_{\mathbf{B}}$ since $\widetilde{F}_{\mathbf{B}^*} = 0$. If $\widetilde{F}_{\mathbf{B}} = 0$, which means that any two terminals are not connected under $\mathbf{B}$, then according to the fact that $E^*$ is the minimum weight set of edges such that the removal of them disconnects each terminal from all the others, we have $cost(\mathbf{B}^*) - cost(\mathbf{B}) = \sum_{(i,j)\in E} w_{ij}(B^*_{ij} - B_{ij}) \le 0$, which means that $\mathbf{B}^*$ is optimal. Otherwise, $\widetilde{F}_{\mathbf{B}} \neq 0$, we have $\widetilde{F}_{\mathbf{B}} \ge M > W_{max} \cdot |E|$. It follows that $cost(\mathbf{B}^*) - cost(\mathbf{B}) < \sum_{(i,j)\in E} w_{ij}B^*_{ij} - W_{max} \cdot |E| \le 0$, which contradicts that $\mathbf{B}$ is optimal. We conclude that $\mathbf{B}^*$ is optimal to $b$. Therefore, $\alpha = 1$ is a satisfying constant.

Finally, when $\beta = 1$, $|cost(E') - opt(a)| \le |cost(E') + \widetilde{F}_{\mathbf{B}} - opt(a)| = \beta|cost(\mathbf{B}) - opt(b)|$ holds for any $\mathbf{B}$. We conclude that the transformation is a linear reduction, and the defender's decision-making problem is MAX SNP-hard. □

Theorem 1 indicates that computing an optimal defender strategy does not admit a polynomial time approximation scheme unless P=NP. Indeed, what makes the defender's problem particularly challenging is the fact that the attacker's problem is hard as well, as we will demonstrate later.

## 4.2 Attackers' Decision-Making Problem

For a given coalition $C$, the attackers' decision-making problem of choosing the optimal attacking plan $\mathbf{a}$ is NP-hard, as the following theorem asserts.

THEOREM 2. *Computing the attackers' optimal attacking plan is NP-hard.*

PROOF. We reduce EXACT-COVER, an NP-complete problem, to coalition's decision-making problem. The EXACT-COVER is defined as follows: Given a finite set $Q$ and a collection $CQ$ of subsets of $Q$, does $CQ$ contain an exact cover for $Q$, i.e., a subcollection $CQ' \subseteq CQ$ such that every element in $Q$ occurs in exactly one member of $CQ'$?

The reduction can be done as follows. First, let the set of skills $S = Q$, and the number of attackers $|C| = |Q|$. For each attacker $i$, let $S_i = \{s_i\}$. For each member $Q_j \in CQ$, we assign a target $t_j$ into $T$ with $S(t_j) = \{s_i | q_i \in Q_j\}$, where $q_i$ is the $i$-th element of $Q$, with value $p_j = |S(t_j)|$. Let $m_{is} = 1$ for any attacker $i$ and skill $s$.

Next, we show that the optimal attacking plan $\mathbf{a}$ satisfies $u(\mathbf{a}) \ge |Q|$ iff EXACT-COVER has a "yes" solution.

**If direction:** If there exists a "yes" solution to EXACT-COVER and $CQ^*$ is the corresponding exact cover of $Q$, then the attacking plan $\mathbf{a}$, where $a_j = 1$ when $Q_j \in CQ^*$, and $a_j = 0$ otherwise, satisfies $u(\mathbf{a}) \ge |Q|$ since $u(\mathbf{a}) = \sum_j a_j p_j = \sum_{Q_j \in CQ^*} |S(t_j)| = |Q|$.

**Only if direction:** The attacker has an optimal attacking plan $\mathbf{a}$ satisfying $u(\mathbf{a}) \ge |Q|$ only if EXACT-COVER has a "yes" solution. According to Eq.(1), any attacking plan $\mathbf{a}$ attacking two targets $t', t''$ such that $S(t') \cap S(t'') \neq \emptyset$ is not feasible since $a_{t'} + a_{t''} > \sum_{i\in C,\, s\in S_i} m_{is} = 1$ for $s \in S(t')$ and $s \in S(t'')$. Thus $u(\mathbf{a}) < |Q|$ holds for any feasible attacking plan if $CQ$ has no exact cover for $Q$.

Therefore, we conclude that computing the optimal plan $\mathbf{a}$ for attackers is NP-hard. □

Despite these rather negative results, we nevertheless undertake the task of devising a method for computing optimal CSG solutions, showing the challenges can be overcome for realistic problem instances, even if not in the worst case.

# 5. SOLUTION APPROACH

We now turn to the computational issues in CSGs. We first propose an *integer program* (IP) to solve it exactly. Because the IP has an exponentially many variables, we propose a *branch and price* algorithm to tackle it.

## 5.1 Coalition Enumeration Approach

We start with the IP computing the optimal solution. Suppose $\mathcal{C}$ is the set of all coalitions for which the induced subgraphs on $G = (N, E)$ are connected, and $C_k \in \mathcal{C}$ is the $k^{th}$ coalition in $\mathcal{C}$ with coalition value denoted by $v_k$. Let $\alpha_{ki} = 1$ if $i \in C_k$ and $\alpha_{ki} = 0$ otherwise. Let the binary variables $\mathbf{x}$ represent the attackers' strategy of forming a coalition structure $CS$, such that $x_k = 1$ iff $C_k \in CS$. If we suppose that all coalitions in $\mathcal{C}$ as well as their associated values have been precomputed, the defender's optimization problem can be formulated as the following integer program, with the objective of minimizing the defender's loss and Eqs.(3b)–(3c) restricting the stable coalition structure:

$$\min_{\mathbf{x},\mathbf{B}} \quad \sum\nolimits_{C_k\in\mathcal{C}} v_k x_k + \sum\nolimits_{(i,j)\in E} B_{ij}\lambda_{ij} \tag{3a}$$

$$\text{s.t.} \quad \sum\nolimits_{C_k \in \mathcal{C}} \alpha_{ki} x_k = 1 \qquad i \in N \qquad (3b)$$

$$\sum\nolimits_{C_k \in \mathcal{C}} \alpha_{ki}\alpha_{kj} x_k \geq 1 - B_{ij} \quad (i,j) \in E \quad (3c)$$

$$\mathbf{x} \in \{0,1\}^{|\mathcal{C}|}, \mathbf{B} \in \{0,1\}^{|N|\times|N|} \qquad (3d)$$

Eq.(3b) ensures that each attacker is in exactly one coalition, so that $CS$ denoted by $\mathbf{x}$ is a partition of $N$. Eq.(3c) means that once an edge $(i,j) \in E$ is not blocked by the defender, i.e., $B_{ij} = 0$, then $i$ and $j$ must be in the same coalition of $CS$. Let $\mathcal{C}_\mathbf{B}$ be the set of coalitions whose induced subgraphs are connected components of $G(\mathbf{B})$. According to Eqs.(3b)–(3c), for a feasible solution $\mathbf{x}$, $x_k = 1$ if and only if $C_k \in \mathcal{C}_\mathbf{B}$ or $C_k$ is a superset of several coalitions in $\mathcal{C}_\mathbf{B}$. With the superadditive coalition's value and the minimizing objective, the solution $\mathbf{x}$ with $x_k = 1$ for $C_k \in \mathcal{C}_\mathbf{B}$ is always optimal for a fixed defender strategy $\mathbf{B}$, corresponding with LEMMA 2. Although IP (3) can obtain the optimal defender strategy $\mathbf{B}^*$, it has exponentially many ($|\mathcal{C}| + |E|$) variables, which makes it impossible to scale up to large game instances with standard *branch and cut* method adopted by popular commercial solvers, such as CPLEX. Therefore, we propose a *branch and price* framework, which combines *branch and bound* and *column generation*.

## 5.2 Branch and Price Framework

Before we introduce our branch and price framework, the following lemma shows that the exponentially large number of binary variables $\mathbf{x}$ in IP (3) can be relaxed without sacrificing optimality (note that $\mathbf{B}$ remains to be binary). This observation will be useful to significantly reduce the size of the branch and bound tree.

LEMMA 3. *The formulation* (3) *is equivalent to its relaxed formulation where* $\mathbf{x}$ *is continuous.*[1]

Figure 2 shows the flow of the branch and price framework. The left part of the figure shows the classic branch and bound tree of solving IP (3) (with $\mathbf{x}$ being relaxed), where the root node in the tree corresponds to the IP (3) and it keeps an *upper bound* (**UB**) on its optimal objective, which can be the objective value of any feasible solution of IP (3). For each created node (an integer program), a *lower bound* (**LB**) is obtained by further relaxing variable $\mathbf{B}$. We refer to the fully relaxed formulation LP relaxation, and denote by $\tilde{\mathbf{B}}^*$ its optimal solution. Two basic operators in branch and bound are *pruning* and *branching*. A node is pruned once its **LB** is not less than **UB** or $\tilde{\mathbf{B}}^*$ is integral. Otherwise, the branching operator will be conducted on a fractional variable $B_{ij}$ in $\tilde{\mathbf{B}}^*$, and two child nodes will be created, one by fixing $B_{ij}$ to 0 and the other with $B_{ij} = 1$. Once a node happens to obtain an **LB** with $\tilde{\mathbf{B}}^*$ being integral, the **UB** of root node will be updated as $\mathbf{UB} = \min\{\mathbf{LB}, \mathbf{UB}\}$. The method terminates when all leaf nodes are pruned, and the final **UB** of root node will be equal to the optimum of IP (3).

Since the LP relaxation of each node in the branch and bound tree has an exponential number of variables, we use column generation to iteratively compute **LB** and $\tilde{\mathbf{B}}^*$ as illustrated by the right part of the Figure 2. Column generation begins by a master LP with a small subset of variables, and solves the slave problem to add new column(s) or variable(s) with negative reduced cost(s) to the master LP, then

[1]Please see online Appendix A for the proof of LEMMA 3 available at: `http://www.ntu.edu.sg/home/boan/papers/AAMAS16_Coalition_Appendix.pdf`

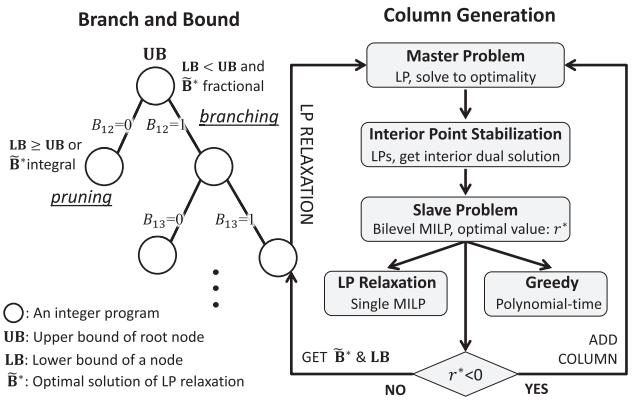



**Figure 2: Branch and Price Framework.**

resolves the master LP, repeating until no such column(s) exists. The column generation approach terminates with the optimal relaxed solution $\tilde{\mathbf{B}}^*$.

The central challenge and novelty of any column generation approach is how to efficiently compute which columns to add, and to guarantee that when this computation adds no new columns, the solution is optimal. The issues involved in adapting a column generation method to our setting are sufficiently non-trivial as to warrant a separate section.

# 6. COLUMN GENERATION

The LP relaxation at each node in Figure 2 is decomposed into *master* and *slave* problems for column generation. The former solves for the relaxed defender strategy $\tilde{\mathbf{B}}$, given a restricted set of coalitions $\mathcal{C}' \subset \mathcal{C}$. The objective for the slave is updated based on the solution of the master, and the slave is solved to identify the best new coalition to be added to the $\mathcal{C}'$ of the master problem, as measured by *reduced cost* (explained later). If no new column can improve the solution the algorithm terminates with an optimal solution.

## 6.1 Master Problem

The master problem (4) starts with a small set of coalitions $\mathcal{C}'$ and solves for the optimal relaxed defender strategy $\tilde{\mathbf{B}}^*$. Note that Eqs.(4a)-(4c) have ensured $\mathbf{x} \leq \mathbf{1}$ and $\tilde{\mathbf{B}} \leq \mathbf{1}$.

$$\min_{\mathbf{x},\tilde{\mathbf{B}}} \quad \sum\nolimits_{C_k \in \mathcal{C}'} v_k x_k + \sum\nolimits_{(i,j)\in E} \tilde{B}_{ij}\lambda_{ij} \qquad (4a)$$

$$\text{s.t.} \quad \sum\nolimits_{C_k \in \mathcal{C}'} \alpha_{ki} x_k = 1 \qquad i \in N \qquad (4b)$$

$$\sum\nolimits_{C_k \in \mathcal{C}'} \alpha_{ki}\alpha_{kj} x_k \geq 1 - \tilde{B}_{ij} \quad (i,j) \in E \quad (4c)$$

$$\mathbf{x} \geq \mathbf{0}, \tilde{\mathbf{B}} \geq \mathbf{0} \qquad (4d)$$

Let $\mathbf{f}$ and $\mathbf{g}$ be dual variables of master constraints (4b) and (4c) respectively. After the master problem is solved, the optimal dual solution $(\mathbf{f}^*, \mathbf{g}^*)$ is computed and passed to the slave problem for finding the column (coalition) to add to the current $\mathcal{C}'$, as we will discuss later.

**Interior Point Stabilization.** Before the slave problem is introduced, it is necessary to understand that the optimal dual solution $(\mathbf{f}^*, \mathbf{g}^*)$ plays a critical role in generating a good column. Since the standard technique computes an optimal dual solution, which is an extreme point of the optimal dual polyhedron and can be characterized by very large

values for some dual variables, an *Interior Point Stabilization* (IPS) method [26] is adopted to compute an optimal dual solution taking values in the center (or at least the interior) of the master problem's optimal dual polyhedron by averaging several extreme points of this polyhedron.

The basic idea of IPS is as follows: once the master problem is solved and the optimal solution $(\mathbf{x}^*, \tilde{\mathbf{B}}^*)$ is obtained, we can depict a polyhedron $\mathcal{D}$ of the dual master problem confined by the complementary slackness conditions [5]. To obtain an extreme point of $\mathcal{D}$, we can simply define a random objective function $\boldsymbol{\mu}^T\mathbf{f} + \boldsymbol{\omega}^T\mathbf{g}$ where $\boldsymbol{\mu}, \boldsymbol{\omega} \sim U(0,1)$, i.e., each element of $\boldsymbol{\mu}$ and $\boldsymbol{\omega}$ is uniformly distributed between 0 and 1, and solve the corresponding LP: $(\mathcal{D}^{\boldsymbol{\mu},\boldsymbol{\omega}}) = \min_{(\mathbf{f},\mathbf{g})\in\mathcal{D}} \boldsymbol{\mu}^T\mathbf{f} + \boldsymbol{\omega}^T\mathbf{g}$. After solving several instances of $(\mathcal{D}^{\boldsymbol{\mu},\boldsymbol{\omega}})$ with multiple random objective coefficients $(\boldsymbol{\mu}, \boldsymbol{\omega})$, we can get some extreme points of $\mathcal{D}$. Taking the average of these extreme points provides an interior point of $\mathcal{D}$ that gives much more centered dual values. For the ease of reading, we provide the details of IPS in online Appendix B[1].

## 6.2 Slave Problem

The slave problem finds the best column (i.e., coalition) to add to the current coalitions in $\mathcal{C}'$. This is done using *reduced cost*, which captures the total change in the defender's utility if a candidate coalition is added to $\mathcal{C}'$. The candidate coalition with minimum reduced cost improves the defender's utility most [5]. The reduced cost $r_k$ of coalition $C_k$, associated with variable $x_k$, is given in Eq.(5), where the dual solution $(\mathbf{f}^*, \mathbf{g}^*)$ measures the influence of the associated constraint on the objective and can be calculated using standard techniques or the IPS method.

$$r_k = v_k - \sum\nolimits_{i\in N} \alpha_{ki} f_i^* - \sum\nolimits_{(i,j)\in E} \alpha_{ki}\alpha_{kj} g_{ij}^* \tag{5}$$

The slave problem then boils down to solving $\min_k r_k$ (i.e., minimizing reduced cost). Clearly, if we were to simply iterate through all coalitions $k$ (of which there are an exponential number), nothing would be gained by column generation. We then exploit the structure of this problem by first formulating the reduced cost minimization problem as a *bilevel mixed-integer linear program* (BMILP) (6), where $(\mathbf{f}^*, \mathbf{g}^*)$ is the optimal dual solution of the master problem (4). What makes this problem bilevel is the fact that, in minimizing reduced cost, we must also compute the value of an associated coalition, since we have avoided precomputing all coalitions' values with branch and price.

In the BMILP (6), the coalition $C$ is represented by binary variable $\boldsymbol{\alpha}$ such that $\alpha_i = 1$ if $i \in C$ and $\alpha_i = 0$ otherwise. The upper level constraints ensure that the coalition $C$ induced a connected subgraph, i.e., $C \in \mathcal{C}$, while the lower level program computes the coalition value $v(C)$.

Eq.(6b) ensures that $\xi_{ij} = \alpha_i\alpha_j$ for edge $(i,j) \in E$. Eqs.(6c)–(6f) enforce $\boldsymbol{\alpha}$ to induce a connected subgraph of $G$ using flow balance techniques [28]: According to constraints (6c) and (6d), $w_i = 1$ if and only if $i$ is the smallest index such that $\alpha_i = 1$, which will be set as the starting point of the flows $\mathbf{h}^+$ and $\mathbf{h}^-$; $h_{lij}^+$ denotes the amount of flow from starting point to ending point $l$, such that passes through edge $(i,j)$ in positive direction $i \to j$, while $h_{lij}^-$ in negative direction $j \to i$; Eq.(6f) restricts the flow to pass through only edges that are in the subgraph induced by $C$, i.e., $(i,j)$ with $\xi_{ij} = 1$; Eq.(6e) is the flow balance constraint with $\mathcal{N}(i)$ denoting the set of vertices connected with $i$, making sure there exists a path, in subgraph induced by $C$, from starting point $i : w_i = 1$ to any $l \in C$ ($\alpha_l = 1$). Therefore the subgraph induced by $C$ is restricted to be connected.

The lower level program (6g) computes the coalition's value $v(C)$ where Eq.(6h) restricts $C$ to have enough skill capacities to conduct the attacking plan $\mathbf{a}$ according to Eq.(1).

$$\min_{\substack{\boldsymbol{\alpha},\boldsymbol{\xi},\mathbf{w}\\ \mathbf{h}^+,\mathbf{h}^-}} \quad \sum\nolimits_{t\in T} p_t a_t - \mathbf{f}^{*\mathbf{T}}\boldsymbol{\alpha} - \mathbf{g}^{*\mathbf{T}}\boldsymbol{\xi} \tag{6a}$$

$$\text{s.t.} \quad \xi_{ij} \le \alpha_i, \xi_{ij} \le \alpha_j,$$
$$\xi_{ij} \ge \alpha_i + \alpha_j - 1 \quad \forall (i,j) \in E \tag{6b}$$
$$w_i \le \alpha_i,$$
$$w_i \le 1 - \alpha_j \quad \forall i,j \in N : j < i \tag{6c}$$
$$\sum\nolimits_{i\in N} w_i = 1 \tag{6d}$$
$$\sum\nolimits_{j\in\mathcal{N}(i)} h_{lij}^+ - \sum\nolimits_{j\in\mathcal{N}(i)} h_{lij}^- \ge$$
$$w_i - (1-\alpha_l), \quad \forall i,l \in N : i \ne l \tag{6e}$$
$$h_{lij}^+ \le \xi_{ij},$$
$$h_{lij}^- \le \xi_{ij}, \quad \forall (i,j) \in E, l \in N \tag{6f}$$

$$\max_{\mathbf{a}} \quad \sum\nolimits_{t\in T} p_t a_t \tag{6g}$$

$$\text{s.t.} \quad \sum\nolimits_{t\in T} \beta_{ts} a_t \le \sum\nolimits_{i\in N} \alpha_i\gamma_{is} m_{is} \quad s \in S \tag{6h}$$
$$\boldsymbol{\alpha} \in \{0,1\}^{|N|}, \boldsymbol{\xi} \in [0,1]^{|E|}, \mathbf{a} \in \mathbb{N}^{|T|}$$
$$\mathbf{h}^+ \ge \mathbf{0}, \mathbf{h}^- \ge \mathbf{0}, \mathbf{w} \ge \mathbf{0}. \tag{6i}$$

The key challenge of the BMILP is that the inner maximization program described in (6g) has integer variables. If we were to relax the integrality constraint, we could reformulate the bilevel program into a single mixed integer linear program, as we describe next.

**Linear Relaxation Approximation.** To reformulate the BMILP to an MILP, we first relax the integer program (6g) of computing the coalition's value in the lower level, such that the decision variable $\mathbf{a} \in \mathbb{N}^{|T|}$ is relaxed to $\tilde{\mathbf{a}} \ge \mathbf{0}$. For this relaxed linear program (RLP), let $\mathbf{u}$ be the dual variable associated with constraint (6h). A pair of feasible solutions $\tilde{\mathbf{a}}$ and $\mathbf{u}$ are optimal for the primal and dual RLPs if and only if the following *complementary slackness conditions* are satisfied [5]:

$$(\sum\nolimits_{s\in S} \beta_{ts} u_s - p_t) \cdot \tilde{a}_t = 0 \quad \forall t \in T \tag{7a}$$

$$(\sum\nolimits_{i\in N} \alpha_i\gamma_{is} m_{is} - \sum\nolimits_{t\in T} \beta_{ts}\tilde{a}_t) \cdot u_s = 0 \quad \forall s \in S \tag{7b}$$

Therefore, the RLP is equivalent with a set of constraints consisting of the primal and dual constraints restricting the feasibility of $\tilde{\mathbf{a}}$ and $\mathbf{u}$ and the complementary slackness conditions ensuring optimality, and the relaxed BMILP is reformulated as an MILP (8), with Eq.(8c) corresponding to the dual constraint of the RLP, and Eqs.(8d)–(8e) equivalent with the complementary slackness conditions (7). Our next results show that the solution quality of the linear relaxation approximation and, consequently, the resulting branch and price framework, provably achieves a competitive ratio which only depends on the number of skills $|S|$ (which we assume to be constant).

$$\min_{\substack{\boldsymbol{\alpha},\boldsymbol{\xi},\mathbf{w},\mathbf{u},\boldsymbol{\phi}\\ \mathbf{h}^+,\mathbf{h}^-,\tilde{\mathbf{a}},\boldsymbol{\varphi}}} \quad \sum\nolimits_{t\in T} p_t\tilde{a}_t - \mathbf{f}^{*\mathbf{T}}\boldsymbol{\alpha} - \mathbf{g}^{*\mathbf{T}}\boldsymbol{\xi} \tag{8a}$$

$$\text{s.t.} \quad \text{Eqs.(6b)–(6f), (6h)–(6i)} \tag{8b}$$

$$\sum\nolimits_{s\in S} \beta_{ts} u_s \geq p_t, \quad \forall t \in T \tag{8c}$$

$$\sum\nolimits_{i\in N} \alpha_i \gamma_{is} m_{is} - \sum\nolimits_{t\in T} \beta_{ts} \tilde{a}_t \leq M(1-\phi_s),$$
$$u_s \leq M\phi_s \quad \forall s \in S \tag{8d}$$

$$\sum\nolimits_{s\in S} \beta_{ts} u_s - p_t \leq M(1-\varphi_t),$$
$$\tilde{a}_t \leq M\varphi_t \quad \forall t \in T \tag{8e}$$

$$\tilde{\mathbf{a}} \geq \mathbf{0}, \mathbf{u} \geq \mathbf{0}$$
$$\boldsymbol{\phi} \in \{0,1\}^{|S|}, \boldsymbol{\varphi} \in \{0,1\}^{|T|}. \tag{8f}$$

LEMMA 4. *The coalition value $\tilde{v}(C)$ computed by LP relaxation of IP* (6g) *is bounded by $\tilde{v}(C) \leq (1+|S|)v(C)$.*

PROOF. We first prove the following fact: For the LP relaxation of IP (6g), if the optimal solution $\tilde{\mathbf{a}}^*$ has a fractional value $\tilde{a}_t^* > 0$ for some target type $t$, then $v(C) \geq p_t$. The idea is that since the coalition has integer capacities for all skills, if $\tilde{a}_t^* > 0$, the coalition's capacity for any skill $s \in S(t)$, which is necessary for attacking target of type $t$, should be larger than 0, i.e., at least 1. Therefore, the coalition can at least attack one target of type $t$ and $v(C) \geq p_t$.

With the above fact, there are two cases to take into consideration for proving this LEMMA.

**Case 1:** $v(C) = 0$. In this case, for any target type $t \in T$, there must exist (at least) a necessary skill $s^t \in S(t)$ such that $C$'s capacity of $s^t$ (i.e., $\sum_{i\in N} \alpha_i \gamma_{is} m_{is}$) is 0. For LP relaxation of IP (6g), any fractional solution $\tilde{a}$ with $\tilde{a}_t > 0$ is infeasible for violating the skill capacity constraint (6h) of skill $s^t$: $\sum_{t\in T} \beta_{ts^t} \tilde{a}_t \leq 0$. Thus $\tilde{v}(C) = v(C) = 0$.

**Case 2:** $v(C) > 0$. Let $\mathbf{a}^*$ be the optimal integer solution for IP (6g), and $\tilde{\mathbf{a}}^*$ be the optimal fractional integer solution for LP relaxation of IP (6g). According to the property of multidimensional knapsack problem [25], $\tilde{\mathbf{a}}^*$ can take fractional values in at most $|S|$ coordinates. Therefore, suppose $p_{max}$ be the largest value among all target types $t$ with $\tilde{a}_t^*$ fractional, we have: $\tilde{v}(C) = v(\tilde{\mathbf{a}}^*) = v(\lfloor \tilde{\mathbf{a}}^* \rfloor) + v(\tilde{\mathbf{a}}^{\mathbf{f}}) \leq v(\mathbf{a}^*) + |S| p_{max}$, where $\tilde{\mathbf{a}}^{\mathbf{f}} = \tilde{\mathbf{a}}^* - \lfloor \tilde{\mathbf{a}}^* \rfloor$ is the residual part of $\tilde{\mathbf{a}}^*$ after rounding. Since $v(C) = v(\mathbf{a}^*) \geq p_{max}$ according to the fact proved before, we have: $\tilde{v}(C) \leq (1+|S|)v(C)$.

To this end, we conclude that $\tilde{v}(C) \leq (1+|S|)v(C)$. □

THEOREM 3. *The branch and price approach, in which the slave problem of column generation is solved by the linear relaxation approximation, can obtain a defender strategy $\mathbf{B}'$ such that $U_d(\mathbf{B}') \geq (1+|S|)U_d(\mathbf{B}^*)$. Note that $U_d(\mathbf{B}) \leq 0$.*

PROOF. The branch and price approach, in which the slave problem of column generation is solved by linear relaxation approximation, can obtain the optimal solution $\mathbf{B}'$ for the following revised program of IP (3) in which $\tilde{v}_k$ is the value of $C_k$ computed by LP relaxation of IP (6g):

$$\min_{\mathbf{x},\mathbf{B}} \quad \sum\nolimits_{C_k\in\mathcal{C}} \tilde{v}_k x_k + \sum\nolimits_{(i,j)\in E} B_{ij}\lambda_{ij} \tag{9a}$$

$$\text{s.t.} \quad \text{Eqs. (3b)–(3d)} \tag{9b}$$

Since $\mathbf{B}'$ is optimal for IP (9), we have:

$$\begin{aligned} -U_d(\mathbf{B}') &= \sum\nolimits_{C\in CS^*(\mathbf{B}')} v(C) + \sum\nolimits_{(i,j)\in E} {B'}_{ij}\lambda_{ij} \\ &\leq \sum\nolimits_{C\in CS^*(\mathbf{B}')} \tilde{v}(C) + \sum\nolimits_{(i,j)\in E} {B'}_{ij}\lambda_{ij} \\ &\leq \sum\nolimits_{C\in CS^*(\mathbf{B}^*)} \tilde{v}(C) + \sum\nolimits_{(i,j)\in E} B^*_{ij}\lambda_{ij} \\ &\leq (1+|S|)(\sum\nolimits_{C\in CS^*(\mathbf{B}^*)} v(C) + \sum\nolimits_{(i,j)\in E} B^*_{ij}\lambda_{ij}). \end{aligned}$$

Therefore, we have $U_d(\mathbf{B}') \geq (1+|S|)U_d(\mathbf{B}^*)$. □

**Greedy Approximation.** Although the MILP of linear relaxation approximation can obtain an approximately-optimal coalition to add to $\mathcal{C}'$ of master problem, it is still relatively slow. Observe, however, that in each iteration of column generation we actually need not find a minimal reduced cost; rather, any reduced cost that improves solution quality would suffice. Consequently, a fast heuristic approach for generating columns would be satisfactory in most iterations, except when the heuristic is unable to find a solution improving column, at which point we can fall back on the MILP. To this end, we propose a Greedy with Multi-Start (GMS) heuristic (see Algorithm 1). This heuristic is fast, and has an important advantage of enabling generation of multiple coalitions in a single iteration of the column generation algorithm (hence, the name multi-start), significantly reducing the number of column generation iterations.

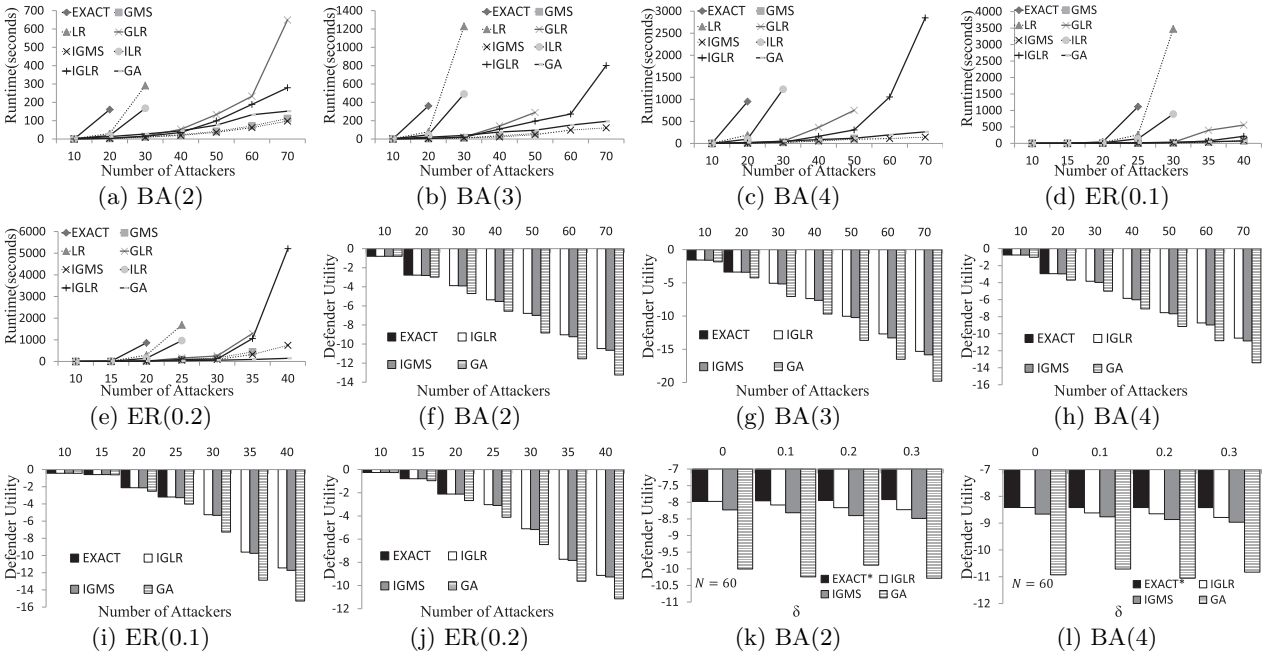
**Algorithm 1:** Greedy with Multi-Start (GMS)

**Input**: optimal dual solution $(\mathbf{f}^*, \mathbf{g}^*)$ of master problem, reduced cost function $r(C)$ defined in Eq.(5)
**Output**: a set of coalitions with negative reduced costs

```
C_GMS = ∅;
for i ∈ N do
    C = {i}, continue = true;
    while continue do
        î = arg min_{i∈N\C} r(C ∪ {i}) − r(C);
        if r(C ∪ {î}) − r(C) < 0 then C ← C ∪ {î};
        else continue = false;
    if r(C) < 0 then C_GMS ← C_GMS ∪ {C};
return C_GMS;
```

# 7. EXPERIMENTAL EVALUATION

We demonstrate the effectiveness of our algorithmic framework through extensive numerical evaluations. We use the CPLEX (version 12.6) to solve all linear programs. All computations were performed on a 64-bit PC with 16 GB RAM and a quad-core 3.4 GHz processor. All values are averaged over 40 instances unless otherwise specified. All scale-free graphs are generated by standard Barabási-Albert model [3] widely used for simulating networks with realistic topological properties, denoted by $BA(d)$, in which $d$ represents the average node degree. All random graphs are generated by standard Erdős-Rényi model denoted by $ER(p)$ where $p$ represents the probability of existence of an edge between any pair of nodes [9]. According to the different approaches adopted for the slave problem, the abbreviations of different branch and price algorithms are as follows: LR for *Linear Relaxation Approximation*, GLR for the combination of the GMS with LR such that LR is called only when GMS returns empty set; The optimal dual solution of the master problem is generated through *Interior Point Stabilization* (IPS) for IGMS, ILR, and IGLR. IP (3) is represented by EXACT. Our benchmark is a heuristic algorithm *Genetic Algorithm* (GA), and for the ease of reading, we put the details of GA in online Appendix C[1]. By default, the instances are parameterized as follows: $|T| = 10$ and $|S| = 20$, the necessary




(a) BA(2) (b) BA(3) (c) BA(4) (d) ER(0.1)
(e) ER(0.2) (f) BA(2) (g) BA(3) (h) BA(4)
(i) ER(0.1) (j) ER(0.2) (k) BA(2) (l) BA(4)

**Figure 3: Scalability: (a)–(e); Optimality: (f)–(j); Robustness: (k)–(l).**

skills for each target type and the ones owned by each attacker drawn randomly from $S$, $m_{is} \sim \{1, 2, 3, 4, 5\}$; $p_t \sim [0, 1.0]$ and $\lambda_{ij} \sim [0, r]$ where $r \in [0, 1]$ is chosen randomly.

## 7.1 Scalability and Optimality

**Runtime.** We test different algorithms on 5 types of networks, and the results are depicted in Figures 3(a)–3(e), from which we can observe the significant efficiency improvement provided by the branch and price framework. The IPS procedure and the GMS approach are shown to reduce the number of iterations of column generation drastically as they further improve the branch and price framework to scale up to realistic-sized problems. For example, Figure 1 contains 76 groups and the average degree is 3.4, which is well within the scalability of IGLR and IGMS.

**Defender Utility.** We compare the defender utility of IGLR and IGMS, with EXACT and GA as benchmarks. The results are illustrated in Figures 3(f)–3(j), from which we can see that IGLR can achieve an almost optimal solution which outperforms GA significantly, in accordance with the theoretical analysis of Theorem 3. The solutions obtained by IGMS, which also outperform GA, are near-optimal especially for networks with higher connectivity, such as large-scale networks of type ER(0.1) and ER(0.2). A tradeoff between runtime and solution quality is observed for IGMS and IGLR, as IGMS shows the better scalability while IGLR obtains the solution with higher quality, however, both of them outperform the alternatives significantly.

## 7.2 Robustness

In the real world, the defender's estimation of target value $p_t$ may not be perfect from attackers' perspective. Therefore, we analyze the performance of our algorithmic framework under the existence of noise of $p_t$. Let $\bar{p}_t$ be the defender's estimation of $p_t$ while $p_t$ is drawn uniformly within $\bar{p}_t \cdot [1 - \delta, 1 + \delta]$. We compare the defender utility of IGLR and IGMS under different degrees of uncertainty, with the near-optimal solution of IGLR, denoted by EXACT* (IP (3) cannot scale to $N = 60$), and heuristic solution of GA as comparison. The results are shown in Figures 3(k)–3(l), from which a decreasing efficiency is observed for IGLR and IGMS when $\delta$ increases. However, IGLR and IGMS solutions still outperform GA significantly under the existence of uncertainty of $p_t$, and IGLR can obtain an almost optimal solution even when $\delta = 0.3$, which shows the strong robustness of our algorithmic framework.

# 8. CONCLUSION

For the first time, this paper studies the problem of blocking attacker coalition through efficient allocation of security resources. This paper provides the following key contributions: 1) We formally define and model coalitional security games. 2) We prove the MAX SNP-hardness of defender's decision-making problem and NP-hardness of attackers' decision-making problem. 3) To address the MAX SNP-hardness, we propose an exact integer program and provide a branch and price algorithm, a linear relaxation based column generation with a constant factor approximation bound, an interior point stabilization procedure, and a greedy method to further improve the scalability. 4) Experiments demonstrate that our methods can scale up to realistic-sized instances and achieve near-optimal performance.

# Acknowledgments

This research is supported by the NSF (CNS-1238959), ONR (N00014-15-1-2621), AFRL (FA8750-14-2-0180) and the National Research Foundation, Prime Minister's Office, Singapore under its IDM Futures Funding Initiative and administered by the Interactive and Digital Media Programme Office.

## REFERENCES


[1] V. Asal and R. K. Rethemeyer. The nature of the beast: Terrorist organizational characteristics and organizational lethality. *Journal of Politics*, 70(2):437–449, 2008.

[2] Y. Bachrach and J. S. Rosenschein. Coalitional skill games. In *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1023–1030, 2008.

[3] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

[4] J. Bennett. Israeli killed as his commandos demolish west bank house. *The New York Times*, 2002.

[5] D. Bertsimas and J. N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, 1997.

[6] G. W. Bush. *National strategy for combating terrorism*. Wordclay, 2009.

[7] G. Chalkiadakis, E. Elkind, and M. Wooldridge. Computational aspects of cooperative game theory. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 5(6):1–168, 2011.

[8] E. Dahlhaus, D. S. Johnson, C. H. Papadimitriou, P. D. Seymour, and M. Yannakakis. The complexity of multiterminal cuts. *The SIAM Journal on Computing*, 23(4):864–894, 1994.

[9] P. Erdős and A. Rényi. On random graphs I. *Publ. Math. Debrecen*, 6:290–297, 1959.

[10] J. Gan, B. An, and Y. Vorobeychik. Security games with protection externalities. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI)*, pages 914–920, 2015.

[11] M. Jain, B. An, and M. Tambe. An overview of recent application trends at the aamas conference: Security, sustainability and safety. *AI Magazine*, 33(3):14, 2012.

[12] P. Klerks. The network paradigm applied to criminal organizations: Theoretical nitpicking or a relevant doctrine for investigators? Recent developments in the netherlands. *Connections*, 24(3):53–65, 2001.

[13] V. Krebs. Uncloaking terrorist networks. *First Monday*, 7(4), 2002.

[14] K. A. Kronstadt. Terrorist attacks in Mumbai, India, and implications for US interests. *Congressional Research Service*, 2008.

[15] J. Letchford and V. Conitzer. Solving security games on graphs via marginal probabilities. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence (AAAI)*, pages 591–597, 2013.

[16] M. Levitt. Untangling the terror web: Identifying and counteracting the phenomenon of crossover between terrorist groups. *SAIS Review*, 24(1):33–48, 2004.

[17] R. Lindelauf, H. Hamers, and B. Husslage. Cooperative game theoretic centrality analysis of terrorist networks: The cases of Jemaah Islamiyah and Al Qaeda. *European Journal of Operational Research*, 229(1):230–238, 2013.

[18] T. P. Michalak, T. Rahwan, N. R. Jennings, P. L. Szczepański, O. Skibski, R. Narayanam, and M. J. Wooldridge. Computational analysis of connectivity games with applications to the investigation of terrorist networks. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 293–301, 2013.

[19] A. Nekrassov. Chechen attack: 'terrorists get weapons from abroad, linked to mideast groups'. *Russia Today*, 2014.

[20] C. Papadimitriou and M. Yannakakis. Optimization, approximation, and complexity classes. In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing (ACM STOC)*, pages 229–234, 1988.

[21] M. B. Peterson. *Applications in Criminal Analysis: A Sourcebook*. Greenwood Press Westport, 1994.

[22] B. J. Phillips. *How Terrorist Organizations Survive: Cooperation and Competition in Terrorist Group Networks*. PhD thesis, University of Pittsburgh, 2012.

[23] J. S. Pistole. Identifying, Tracking and Dismantling the Financial Structure of Terrorist Organizations. 2003.

[24] C. L. Powell. Remarks to the United Nations Security Council, February 5, 2003.

[25] D. Pritchard. An LP with integrality gap $1+\epsilon$ for multidimensional knapsack. *arXiv preprint arXiv:1005.3324*, 2010.

[26] L. Rousseau, M. Gendreau, and D. Feillet. Interior point stabilization for column generation. *Operations Research Letters*, 35(5):660–668, 2007.

[27] T. Shanker and E. Schmitt. Three terrorist groups in Africa pose threat to US, American commander says. *The New York Times*, 2011.

[28] S. Shen, J. C. Smith, and R. Goli. Exact interdiction models and algorithms for disconnecting networks via node deletions. *Discrete Optimization*, 9(3):172–188, 2012.

[29] M. K. Sparrow. The application of network analysis to criminal intelligence: An assessment of the prospects. *Social networks*, 13(3):251–274, 1991.

[30] M. Tambe. *Security and Game Theory: Algorithms, Deployed Systems, Lessons Learned*. Cambridge University Press, 2011.

[31] Y. Vorobeychik, B. An, M. Tambe, and S. P. Singh. Computing solutions in infinite-horizon discounted adversarial patrolling games. In *Proceedings of the 24th International Conference on Automated Planning and Scheduling (ICAPS)*, pages 314–322, 2014.

[32] Z. Wang, Y. Yin, and B. An. Computing optimal monitoring strategy for detecting terrorist plots. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*, 2016.

[33] Y. Yin, B. An, and M. Jain. Game-theoretic resource allocation for protecting large public events. In *Proceedings of the 28th Conference on Artificial Intelligence (AAAI)*, pages 826–834, 2014.

[34] Y. Yin, H. Xu, J. Gan, B. An, and A. X. Jiang. Computing optimal mixed strategies for security games with dynamic payoffs. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 681–688, 2015.

[35] M. Zhao, B. An, and C. Kiekintveld. Optimizing personalized email filtering thresholds to mitigate sequential spear phishing attacks. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*, 2016.