NANYANG
TECHNOLOGICAL
UNIVERSITY

COMBATING ADVERSARIES IN NETWORK-STRUCTURED SECURITY DOMAINS

QINGYU GUO

NTU-UBC RESEARCH CENTER OF EXCELLENCE IN ACTIVE LIVING FOR THE ELDERLY

INTERDISCIPLINARY GRADUATE SCHOOL

2018

COMBATING ADVERSARIES IN NETWORK-STRUCTURED SECURITY DOMAINS

QINGYU GUO

NTU-UBC RESEARCH CENTER OF EXCELLENCE IN ACTIVE LIVING FOR THE ELDERLY
INTERDISCIPLINARY GRADUATE SCHOOL

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Doctor of Philosophy

2018

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. However, the network-structured security domains capturing a variety of realistic problems such as preventing the illegal good smuggling, are not paid enough attentions. Many critical challenges and issues exist in those scenarios which, however, remain to be solved, including the cooperation among multiple adversaries, preventing adversarial flows on the network, and the planning and learning in repeated games. Although existing work on target-structured security domains has considered part of those issues, their methodologies fail to deal with the network structure, as the adversarial activity on the network is much more complex.

This thesis provides the first models and approaches for several important network-structured security domains: 1) the interdiction of multiple cooperative adversaries connecting on the network from forming powerful coalitions, 2) the interdiction of an adversarial network flow, and 3) the online policy for combating the repeated attacks on the network.

First, 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.

Second, focusing on the challenging task of optimally interdicting an illegal network flow, we make several contributions. We propose a new Stackelberg game model for network flow interdiction. To solve this game, we provide a novel Column and Constraint Generation approach for computing the optimal defender strategy. We conduct extensive complexity analysis of the column generation subproblem. We propose compact convex nonlinear programs for solving the subproblems. Novel greedy and heuristic approaches for subproblems with good approximation guarantee are also incorporated. With these novel approaches, our approach can obtain a robust enough solution outperforming the existing methods and heuristic baselines significantly and scale up to realistic-sized problems according to extensive experimental evaluation.

Finally, we study repeated network interdiction games with no prior knowledge of the adversary and the environment. We provide the first defender strategy, that enjoys nice theoretical and practical performance guarantees, by applying the adversarial online learning approach. In particular, we model the repeated network interdiction game with no prior knowledge as an online linear optimization problem, for which a novel and efficient online learning algorithm, SBGA, is proposed, which exploits the unique semi-bandit feedback in network security domains. We prove that SBGA achieves sublinear regret against adaptive adversary, compared with both the best fixed strategy on hindsight and a near optimal adaptive strategy. Extensive experiments also show that SBGA significantly outperforms existing approaches with fast convergence rate.

Acknowledgements

Reaching the final line of this long journey, I want to express my tremendous gratitude to those who helped me during the past four years. I have enjoyed the life as a Ph.D. student and it has been a great privilege for me to share my experiences and relationships with a number of excellent people along the way.

First of all, I want to thank my supervisor Bo An. Without your endless help and instructions, this point can not be reached. Thank you for giving me this opportunity to pursue my Ph.D. degree in your group. Since the first day of joining in your research group, I was deeply impressed by your enthusiasm and love to research, and these become the things that I will strive for in my future career. When I was a fresh student with little knowledge and idea about research, your correction and guidance helped me go through puzzles and find my own way as a researcher. I cannot say more thanks to your inspirations on every project that I have participated in. Your commitment to working on real world problems and benefits to the society shapes my philosophy as a researcher. Your openness to listening to my opinions and concerns, critical but always helpful comments and patient attitude to long discussions mean a lot to me and I will always keep these in mind. Not only, you have set a lifelong example for me on how to do research, but you also teach me many lessons on how to be professional and responsible, how to properly deal with many challenging issues, and many others. I will be forever grateful to your supervision and the connections and relationships built in your research group will last forever and grow stronger over time.

I would like to thank my thesis advisory committee members: Chunyan Miao, Huaxiong Wang, and Zhiqi Shen. Those discussions and meetings help me a lot in improving my work, and build a better vision for me in doing research. Your guidance and kind help are so valuable for me and I will always be thankful for your efforts in advising me.

During the past four years in NTU, I am grateful to have the privilege to collaborate with so many excellent researchers: Jiarui Gan, Yevgeniy Vorobeychik, Long Tran-Thanh, Yair Zick, Branislav Bošanský, Christopher Kiekintveld, Milind Tambe and Fei Fang. I thank you for your guidance and insights in those projects we worked on together as well as the patience of mentoring.

It was also a great fortune for me to share my Ph.D. experience with a group of talented fellow students and postdoc researchers who all become my close friends: Haipeng Chen, Yanhai Xiong, Mengchen Zhao, Zhen Wang, Jiarui Gan, Yue Yin, Xinrun Wang, Jiuchuan Jiang, Martin Strobel, Youzhi Zhang, Wanyuan Wang, Lei Feng, Xu He and Aye Phyu. Thank you all for those moments of laughs, happiness and excitements, and your generous help whenever I needed. For me, it is a priceless memory of sharing the office, attending the conferences, fighting for the conference deadlines, having internal group meetings, going hiking and enjoying group treatments with you. Special thanks to Jiarui Gan for the excellent help and guidance in my first year.

I want to thank the LILY center for providing the office place and financial support.

Lastly but importantly, I want to thank my parents for your unconditional and endless love, understanding and supports. Thank you for always believing in me, and sharing with my good and bad times.

Contents

List of Figures

List of Tables

Chapter 1

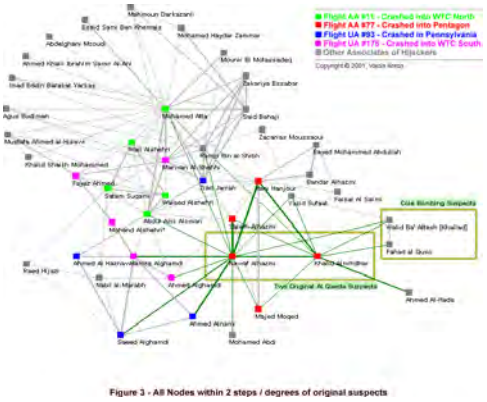
Introduction

Security is one of the longstanding and critical issues. The challenges of security include the protection of public infrastructure, airports and ports, patrolling the nature reserve and wildlife against poachers, interdiction of the illegal good smuggling, and so on. One major concern of the security agencies (i.e. defenders) in those domains is the insufficiency of the security resources, such as checkpoints, police officers, canines, inspection facilities. With limited security resources in hand, it is critical for the defender to allocate them efficiently and effectively so that the damage caused by the adversary's activities is minimized. This security resource allocation problem turns out to be complex due to the strategic behavior of the adversary. That is, the human adversaries are mostly sophisticated that they can conduct surveillance on the defender's resource allocation at a particular time, and the adversaries can learn the defensive pattern before launching an attack. In this case, a deterministic resource allocation policy would perform extremely worse. Several other concerns such as the human behavior and repeated interaction make the policy-making more complex.

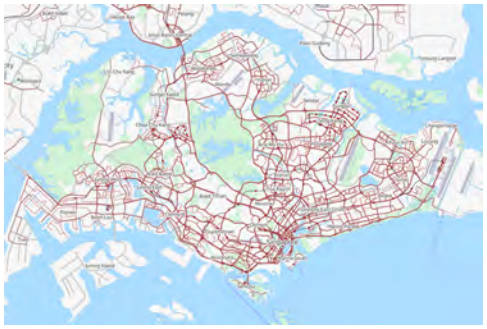
In the past decade, game theory has become a powerful paradigm in modelling complex competition and cooperation systems with multiple agents involved. In particular, the *Stackelberg security game* model has become one of the standard methodologies in complex security domains and several systems have been successfully deployed to protect airports, ports, transportation and wildlife [2–8]. In a Stackelberg security game, there are two players, a defender, also called leader, who protects the key targets with

limited resources, and an attacker, namely the follower, who can launch an attack on one of the target. The defender commits to a *mixed* strategy which is a probability distribution over all possible resource allocations, while the attacker, knowing the mixed strategy of the defender through extensive observation, launches an optimal attack. A vast amount of efforts are put into designing efficient algorithms to solve for the optimal mixed strategy for the defender by mitigating the exponentially explosive defender strategy space [9–11]. The human behavior modelling is also well studied and several stochastic behavior models incorporating the prospect theory and quantal response equilibrium are proposed and applied in protecting ports and wildlife [7, 12–18]. A recent branch of security games focuses on the planning and learning in repeated Stackelberg security games where the goal is to design an online policy with nice properties against unknown adversary [19–23].

(a) Cyber network



(b) Terrorist network



(c) Transportation

(d) Social network

FIGURE 1.1: Network-structured security domains.

The security domains that existing Stackelberg security game models target on can

be roughly classified into two categories. One is the target-structured security domain [2–8] where both players' actions are focusing on protecting or attacking the individual targets. The other one is the network-structured security domain [24–30] where the defender and the attacker are competing on a network, such as the transportation network. The actions of both players are also network-featured. For example, the defender's action could be an allocation of checkpoints or barriers on edges, while the attacker's decision is selecting one path to escape or a network flow to smuggle illegal goods. Many security issues in real world occur in the network-structured security domains, and Figure 1.1 lists several typical domains abstracted by the networks, such as the cyber network, the terrorist network, the transportation network and the social network.

While the target-structured security games research has progressed significantly recently, the scope of research on the network-structured security domains is limited, as most of the works in network security games[1], if not all, focus on the one-shot two-player games [24–32]. Given the significant applicability of network security games on many real-world domains, it is of huge interest to tackle the forementioned challenges, including multiple players, human behavior modelling, uncertainties, planning and learning in repeated games.

Along with the presentation of this thesis, we will find that, though some of these challenges, such as the uncertainties, and the planning and learning, have been considered in security games research, in particular, the target-structured game models, the network structure prevents us to apply their paradigms directly in network security games. For instance, the graph topology affects the interaction of agents; The adversarial behavior on the network is much more complex than that in target-structured domains. Novel approaches and paradigms, in such case, are necessary for handling the forementioned challenges. This thesis, for the first time, makes an attempt to address these challenges to an extend, in the network security games. The three major contributions towards enriching the network security games research are as follows.

[1]For the ease of reading, we will refer the games modelling the network-structured security domains to the network security games.

1.1 Combating Multiple Cooperative Adversaries

The first part of this thesis focuses on the problem of combating multiple cooperative adversaries. In many real-world scenarios, multiple adversaries exist and are cooperative with each other, sharing skills and making coordinated attacks. It is critical for the law enforcement to interdict such cooperations [33], which leads to our consideration of combating multiple cooperative adversaries.

Due to the geographical limits and the relationship restrictions, the cooperation of multiple adversaries is commonly represented by networks. One typical network is the terrorist network, which is a network connecting the terrorist groups and individuals. There are increasing evidences of the cooperation between terrorist groups. For example, three terrorist groups in Africa have been reported to share funds, training, and explosive materials with each other [34], and Chechen terrorists were reported to obtain weapons from terrorist organizations in the Middle East [35]. Various forms of cooperation have been spotted, including transferring funds, weapons support, training, and other sharing of critical skills and resources.

The cooperation of terrorist groups is commonly achieved through physical communication channels. For example, front companies and charities are used for transferring funds [36], and transportation hubs including roads, ports, and rivers are critical for the transfer of weapons, training and coordinate attacks. An important measure to break the ally of multiple groups is to cut off the communication channels, which is emphasized in the U.S. strategy for combating terrorists, quoted as *"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"* [33].

However, the existing research on terrorist networks neglects the effective interdiction of the cooperation networks. Instead, the scope of such research has been limited to either the social network analysis of terrorist groups [37–40], or using cooperative game theory to identify the key members in the network [41, 42]. The related work in security games, on the other hand, tends to model the attackers as independent agents [4, 43–45]. The cooperation of multiple attackers is thus ignored in the security games research.

In the first part of the thesis, we propose a novel *Coalitional Security Game (CSG)* model, an integration of cooperative game and Stackelberg security game, to capture the cooperation of multiple attackers and competition between the defender and attackers simultaneously. In CSG, there are several attackers, such as terrorist groups and individuals, connecting with each other on a network. The defender, on the other hand, due to the limited security resources, can cut off a certain number of connections of the network in order to reduce the threats of the cooperation among attackers. To measure the risk posed by cooperative attackers and analyze the coalition formation of them, we adapt the *coalitional skill game* model on the network structure. In this model, each attacker is associated with several skills, such as financial supports, weapons reservation, training, attacking. Those skills are critical in conducting the tasks, in particular, attacking important targets. A group of attackers can form a coalition if they can induce a connected subgraph on the network. The skills owned by members of the same coalition can be shared with each other and the value of a coalition is thus measured by the threat of conducting attacks on high-valued targets, once the required skills necessary for such attacks are in the skill deposits of members in this coalition.

To solve the CSG and compute the optimal interdiction strategy for the defender turn out to be an extremely challenging task. The main challenge comes from the computational complexity, as the strategy space for the defender and the set of possible coalitions for the attackers are both exponentially large. Besides, the coalition formation problem is a well-known challenging problem in cooperative game theory. The scalability challenge is formally verified by our theoretical analysis of the computational complexity, which shows that computing an optimal strategy for the defender is MAX SNP-hard, indicating no polynomial time approximation scheme unless P=NP, and evaluating the threat of a coalition is also NP-hard.

To overcome the computational challenges, we first formulate the decision-making problem for the defender as an integer program with all the possible coalitions enumerated. Then we develop a sophisticated branch and price algorithm, which is a combination of branch and bound, a standard paradigm for solving discrete optimization, and column generation for solving the large scale linear program. Several novel approaches are proposed to improve the scalability of the branch and price algorithm. Since

the slave problem of column generation is formulated as a bilevel mixed-integer linear program (BMILP), we adopt a linear relaxation approximation to reformulate it as a scalable single level MILP and a constant factor approximation guarantee is proved. Furthermore, we provide an interior-point stabilization method to generate an interior dual optimal solution of the master problem of column generation. The stabilized dual solution speeds up the convergence of column generation, and novel heuristics are integrated to produce multiple columns in each iteration of column generation.

To test the effectiveness of our proposed algorithmic framework, we conduct extensive experimental evaluation on both simulated and real-world terrorist networks. The results show that our algorithms provide orders of magnitude improvement in speed over the straightforward exact integer program. Furthermore, our algorithms are able to produce scalable high-quality approximation solutions of realistic problem instances and significantly outperform the classic heuristic search algorithms. Finally, our solutions are shown to be remarkably robust against uncertainties of attacker's payoffs.

1.2 Combating the Illegal Network Flow

Unlike the previous part which considers multiple cooperative adversaries, the rest parts of the thesis target on a single adversary, or more generally, multiple independent adversaries. Instead, the main focus of the rest content is the interdiction of an illegal network flow which captures various realistic adversarial measures, including the infectious disease propagation, enemy supply chains, and illegal goods smuggling. The second and the third parts tackle the network flow interdiction problem using different paradigms. In the second part, we adopt the minimax solution of the one shot two-player zero-sum games, and aim to remedy the existing network security games research which neglects the network flow interdiction problem. We also enrich the methodologies for solving minimax solutions by proposing the *Column and Constraint Generation (CCG)* algorithm, which can be regarded as the generalized version of the widely adopted *double oracle* method.

Though the minimax solution is robust and can be efficiently computed with C-CG algorithm, it fails to exploit the history information of the interaction between the defender and the adversary. Besides, it also poses a strong requirement of the prior knowledge of the environment of the defender. Therefore, the third part of the thesis, for the first time, addresses the planning and learning problem in repeated network security games. An online policy with low regret is proposed by adapting the online linear optimization methodology with semi-bandit information into consideration.

1.2.1 Network Flow Interdiction Games

Many illegal activities on the network can be represented as a flow, and one typical scenario is the illegal good smuggling. Preventing the illegal good flow on the transportation network is a longstanding problem for the law enforcements. For example, the U.S. government spends $1.4 billion and assigns thousands of agents to protect the U.S.-Mexico border from the illegal drug trafficking threats. The main efforts of the illegal drug interdiction are contributed by the checkpoints operated by the Border Patrol agency. However, due to the shortage of staffs, canines, and inspection facilities, only few of the checkpoints are actually operational [46]. *Randomized allocation strategies* of limited security resources are essential, especially knowing that the smugglers are sophisticated and have the ability to observe the operational status of checkpoints.

Unfortunately, computing the optimal randomized resource allocation for network flow interdiction is long neglected. Though the *network interdiction problem* is well studied, where the interdictor aims to prevent the objective of the competitor including the shortest path, the maximal network flow, via deletion of edges/nodes, or decreasing edge capacity [24–28, 47], none of them consider the randomized resource allocation strategy. While the randomized resource allocation is assumed to be not available for some domains, it is possible and essential in effective interdiction in the smuggling domain [46]. Besides, randomization does lead to a drastic increase in problem complexity. While in the deterministic interdiction model, an adversary's flow can be compactly represented, it is not the case in randomized setting.

Though we model the problem as two-player zero-sum game and adopt the minimax solution, which is widely studied and used in the network security games research [29, 31, 48], their methodologies based on double oracle framework cannot be directly applied to the network flow interdiction problem. The main reason is that, in their models, the adversary has finite pure strategies (s-t paths), while the possible network flows are infinite. With finite pure strategy space, the double oracle, based on von Neumann's minimax theorem, is applicable and ensures to converge to the minimax solution. However, such conclusion collapse with infinite pure strategies, and we generalize the double oracle framework with Sion's minimax theorem to solve the network flow interdiction problem.

The first contribution of this part is a novel model called *Network Flow Interdiction Game (NFIG)*, which captures the randomized interdiction on adversarial activities in network form. NFIG is a one shot two-player zero-sum game, where the defender allocates limited security resources to operate a limited subset of checkpoints, and the adversary plays a network flow satisfying the capacity constraints on edges. We assume the Stackelberg leader-follower manner for NFIG, that the defender commits to a mixed strategy first, while the adversary, knowing the defender's mixed strategy through extensive surveillance, makes his best response. With zero-sum nature, the Stackelberg equilibrium, Nash equilibrium and minimax solution coincide with each other.

Computing the minimax solution turns out to be extremely challenging due to the exponentially large number of the resource allocations for the defender, and infinite network flows for the adversary, which also makes the double oracle framework invalid for NFIG. Thus, the second contribution of this part is a novel framework called *Column and Constraint Generation (CCG)*, regarded as a generalization of the double oracle framework, which ensures the minimax solution to be obtained with finite iterations.

The third contribution of this part is the computational complexity analysis of the subproblem of the CSG framework, as well as the novel approaches to solve the subproblems. We formally show that the column generation subproblem is NP-hard for general networks by reducing the set cover problem to it, while polynomial-time solvable for the tree-like networks. We provide a convex integer program which solves for the optimal solution of the column generation subproblem, as well as a polynomial-time

greedy algorithm whose solution is proved to achieve competitive ratio. For the constraint generation subproblem, we also propose an exact convex integer program, and a competitive heuristic algorithm based on the Variable Neighborhood Search (VNS). Extensive experimental evaluation shows that our CCG framework with novel approaches for subproblems can scale up to realistic-sized NFIG instances and significantly outperform baseline approaches.

1.2.2 Repeated Network Interdiction Games

While minimax solution shares several nice properties, especially that it guarantees the defender the value of the game, it may not be the perfect solution for all network flow interdiction problems. In fact, two major shortcomings of the minimax solution drive us to propose an alternative solution. First, the effectiveness of the minimax solution strongly relies on the prior knowledge of the environment, such as the capacities and other parameters, and the perfectly rational model for the adversary's behavior. Although the value of the game is guaranteed regardless of the actual adversary's behavior model, the optimality of the minimax solution is weakened if the adversary deviates from the best response. Second, although the minimax solution seems to be fair to adopt for one shot game, many network flow interdiction problems involve the repeated interactions between the defender and the adversary. For example, the drug seizure happens almost daily and thousands tons of drugs are reported to be seized on land within the United States per year [49].

Adversarial online learning is a suitable paradigm for planning against an adversary based on historical information. While there are several existing works in target-based security games domain focusing on the planning in repeated games [19–23, 50], to the best of our knowledge, this thesis is the first to study the planning and learning in network security games. The most relevant work is the one by Xu et al. [23] applying online learning approaches to provide online policy for the defender in the repeated target-based Stackelberg security games. Though their work inspires our adoption of online learning paradigm in repeated network interdiction problem, this application is in fact non-trivial, since the problem is naturally formulated as an online submodular

maximization problem due to the problem structure. While the state-of-the-art learning algorithms for online submodular maximization problem fail to achieve convincing performance [51], we successfully transform the repeated network interdiction problem into an online linear optimization, for which low-regret solution is available. However, existing algorithms for online linear optimization fail to exploit the special property of the network interdiction domain [52, 53], that the defender has more information than they expect. We call such information the *semi-bandit* feedback, in particular, the amount of interdicted flow at each operated checkpoint.

Thus, in the third part of this thesis, we propose a new online learning algorithm, which exploits the *semi-bandit* feedback, called *Semi-Bandit style Geometric decision algorithm against an Adaptive adversary (SBGA)* to provide online decisions for the defender. We formally prove that SBGA achieves $\mathcal{O}(T^{2/3})$ regret against *adaptive* adversary who can adjust his actions according to the full information of the history interactions. Furthermore, we also prove a low regret bound of SBGA with a competitive near optimal adaptive defender strategy as the benchmark. Extensive experimental evaluation demonstrates that our approach significantly outperforms existing online learning algorithms and achieves much faster convergence rate.

1.3 Thesis Overview

The structure of this thesis is organized as follows: Chapter 2 discusses background material and related work to provide the context of this thesis. Chapter 3 considers security games with multiple cooperative adversaries on the network. Chapter 4 investigates network flow interdiction problems from the perspective of two-player zero-sum games. Chapter 5 explores the repeated interactions in network security games using online learning paradigm. Chapter 6 summarizes the thesis and presents possible directions for future work. The notations used in Chapters 2-5 are listed at the end of this thesis following an alphabet order.

Chapter 2

Background

This chapter reviews the background knowledge which supports this thesis with fundamental theorems, models and algorithms. The topics discussed in this thesis fall into the research area called security games which applies game theoretical reasoning, large-scale optimization and online learning to optimize the usage of limited security resources. The overview of the security games literature is given in Section 2.3, and before that, we introduce the fundamental models and theorems in game theory. We first present the normal form game model and the popular Nash equilibrium and the Stackelberg equilibrium for the normal form game in Section 2.1. We then discuss an important game philosophy called the zero-sum game, and the well known minimax theorem, which plays the key role in designing strategy generation algorithms for solving large-scale zero-sum games such as the double oracle and the column and constraint generation (Chapter 4). In Section 2.2, we illustrate the basic online convex and linear optimization models and algorithms for studying the repeated game plays. In particular, we review two widely used online optimization strategies, follow the perturbed leader and BGA, which inspire us to propose the SBGA algorithm for repeated network interdiction problem in Chapter 5.

	C_1	C_2
R_1	2,1	4,0
R_2	1,0	3,2

TABLE 2.1: A simple example of two-player strategic form game.

2.1 Game Theory

Game theory is the study of mathematical models for the confliction and cooperation among intelligent rational decision-makers. The widely adopted representation in game theory is the *normal form game*, also called the *strategic form game*.

Definition 2.1. A strategic form game is composed of

- a set of players N,
- a set of actions or pure strategies A_i for each player i,
- a payoff function $u_i : A \to \mathbb{R}$ for each player i.

Table 2.1 shows the payoff matrix of a simple two-player strategic form game, where the row player has two actions R_1 and R_2 while the column player has actions C_1 and C_2. In order to determine the outcome of the game, we must specify any additional information on the proceeding of the game and the players' model of the other players. In simultaneous-move and non-cooperative situations, each player will assume that the other players are rational and choose the actions to maximize their payoffs. In the example shown in table 2.1, the column player can reason that the row player will never play action R_2 as the payoff of playing R_1 is always higher regardless of the action chosen by the column player. Given that the row player will play R_1 for sure, the action maximizing the payoff of the column player is C_1. The outcome where row player plays R_1 and column player chooses C_1 is an equilibrium in the sense that no player has incentive to deviate individually.

Although the example shown in table 2.1 has an equilibrium with only one pure strategy for each player, this may not always be the case. For example, if we slightly modify the payoff matrix as shown in table 2.2, we can easily verify that at each outcome

	C_1	C_2
R_1	3,1	2,2
R_2	1,3	3,2

TABLE 2.2: An example of two-player strategic form game with no pure strategy equilibrium.

composed with pure strategies for both players, at least one player has incentive to deviate and play the other action for higher payoff. In general, we are interested in the randomized action selection of the player, called the *mixed strategy*. The mixed strategy σ_i of player i is a probability distribution over all possible actions A_i and $\sigma_i(a_i)$ denotes the probability of selecting action a_i. Denote by Δ^{A_i} the simplex set of all possible probability distributions over A_i. Let $\sigma = (\sigma_1, ..., \sigma_N)$ denote the strategy profile of all players, and $\sigma_{-i} = (\sigma_1, ..., \sigma_{i-1}, \sigma_{i+1}, ..., \sigma_N)$ be the strategy profile for players except i. The support set of a mixed strategy is the set of pure strategies with positive probability. The widely used solution concept in simultaneous-move and non-cooperative game is *Nash equilibrium*, an outcome where no player has incentive to deviate by playing a different (mixed) strategy. Referring the utility — the expected payoff — of each player i as $U_i(\sigma_i, \sigma_{-i})$ with respect to the mixed strategy profile σ, the Nash equilibrium is thus an outcome σ^* satisfying

$$U_i(\sigma_i^*, \sigma_{-i}^*) \geq U_i(\sigma_i, \sigma_{-i}^*) \quad \forall i \in N, \sigma_i \in \Delta^{A_i}. \tag{2.1}$$

2.1.1 Stackelberg Games

In many resource allocation problems in security domain, the Stackelberg game model, also called the leader-follower game, is adopted instead of the simultaneous-move game [54]. There are two players in the Stackelberg game, a leader and a follower. That is, $N = \{l, f\}$. The game is played in two stages. The leader first declares a mixed strategy σ_l, after which the follower, knowing the the strategy committed by the leader, chooses his strategy σ_f to play. While it seems harmful for the leader to expose her strategy to the follower, the Stackelberg commitment turns out to be an advantage of

the leader [55]. Take the game in table 2.1 as an example. Let the row player be the leader. Instead of following the simultaneous-move equilibrium and playing action R_1, the row player can declare that action R_2 is always played. Under such circumstance, the column player will select action C_2 since it is the best response action against R_2 for the column player. The commitment brings the row player a payoff of 3, higher than 2 in the Nash equilibrium.

The prevailing solution concept in the Stackelberg game literature is called the *Strong Stackelberg Equilibrium (SSE)*, which consists of a leader strategy and a follower response function $g : \Delta^{A_l} \rightarrow \Delta^{A_f}$. In SSE, the follower response function g always returns the best response strategy. When multiple best response strategies exist for the follower, the response function g in SSE will break the tie in favor of the leader. The formal definition of SSE is as follows.

Definition 2.2. $\langle \sigma_l^*, g(\sigma_l^*) \rangle$ forms an SSE if and only if

- $U_l(\sigma_l^*, g(\sigma_l^*)) \geq U_l(\sigma_l, g(\sigma_l)) \quad \forall \sigma_l \in \Delta^{A_l}$,
- $U_f(\sigma_l, g(\sigma_l)) \geq U_f(\sigma_l, \sigma_f) \quad \forall \sigma_l \in \Delta^{A_l}, \sigma_f \in \Delta^{A_f}$,
- $U_l(\sigma_l, g(\sigma_l)) \geq U_l(\sigma_l, \sigma_f) \quad \forall \sigma_l \in \Delta^{A_l}, \sigma_f \in BR(\sigma_l)$.

$BR(\sigma_l) = \arg \max_{\sigma_f \in \Delta^{A_f}} U_f(\sigma_l, \sigma_f)$ denotes the set of best response strategies for the follower.

2.1.2 Zero-sum Games

The two-player zero-sum game is closely related to this thesis. The players are still referred as $N = \{l, f\}$. In zero-sum game, the payoffs of both players always sum up to zero for any strategy profile. The zero-sum game is well studied and shares several critical properties from both the theoretical and computational standpoints, the significant one of which is that the Nash equilibrium, maximin and minimax all give the same solutions. Since the maximin formulation naturally returns the SSE solution, the SSE in the zero-sum game is equivalent with the Nash equilibrium in terms of the leader

Algorithm 1: Double Oracle Framework

Initialize A'_l with random leader actions;
Initialize A'_f with random follower actions;
repeat
  $(\sigma_l, \sigma_f) \leftarrow CoreLP(A'_l, A'_f)$;
  $A'_l \leftarrow A'_l \cup \{LeaderOracle(\sigma_f)\}$;
  $A'_f \leftarrow A'_f \cup \{FollowerOracle(\sigma_l)\}$;
until convergence;
return (σ_l, σ_f).

strategy. The leader's equilibrium strategy can thus be obtained by solving the maximin linear program in time polynomial in the size of the payoff matrix as follows.

$$\max_{\sigma_l} \quad U_l \tag{2.2a}$$

$$\text{s.t.} \quad U_l \leq \sum_{a_l \in A_l} \sigma_l(a_l) u_l(a_l, a_f) \quad \forall a_f \in A_f \tag{2.2b}$$

$$\sum_{a_l \in A_l} \sigma_l(a_l) = 1 \tag{2.2c}$$

$$\sigma_l \geq 0. \tag{2.2d}$$

2.1.2.1 Double Oracle

Although the maximin LP (2.2) can be solved in time polynomial in the size of the payoff matrix, in many complex problems, the strategy spaces A_l and A_f can be extremely large. For example, in the simplest security game model, the size of the strategy space for the defender is the combinatorial number C_n^m where m and n denote the numbers of resources and targets respectively. It is even impractical to enumerate all the strategies for large-scale instances. Double oracle algorithm is widely used to mitigate the scalability issue of solving large-scale zero-sum games, especially in security game literature. The double oracle algorithm takes the maximin LP (2.2) as a core, for which we denoted as *CoreLP*, and incrementally enlarges the strategy spaces of both players. Double oracle algorithm is useful in practice since the support sets of the equilibrium strategies for both players are often limited compared with the extremely large strategy spaces. The basic framework of double oracle is illustrated in Algorithm 1. The algorithm starts from a restricted game where only few random actions for both players

are considered. The CoreLP is solved on the restricted game with strategy spaces A'_l and A'_f for both players. The (Nash) equilibrium profile (σ_l, σ_f) is obtained for the restricted game, which may not be the equilibrium profile for the original game with strategy spaces A_l and A_f. Thus, two oracles are called to solve for the best responses for both players from A_l and A_f, denoted by *LeaderOracle* and *FollowerOracle*. The best responses are added to the restricted strategy spaces A'_l and A'_f, and the procedure is repeatedly executed, until the convergence criterion is met. The commonly used criterion is that both oracles return the best responses already existing in the restricted strategy spaces. In the worst case, the restricted strategy spaces will grow until they contain all the possible strategies in A_l and A_f, which are finite. Therefore, the algorithm will eventually terminate with finite iterations, and the returned strategy profile is proved to the equilibrium for the original game [56]. The computationally costly part of the double oracle algorithm is solving the two oracles which search the best responses from the entire strategy spaces A_l and A_f. The novelty of most adoption of the double oracle framework lies in designing efficient algorithms to solve these oracles.

2.1.2.2 Minimax Theorem

The guarantee of equilibrium to be obtained with double oracle algorithm is proved by the minimax theorem, in particular, the von Neumann's minimax theorem, which is widely acknowledged as the starting point of the game theory. The von Neumann's minimax theorem states that for two-player zero-sum game with finite pure strategies, the minimax value equals the maximin value. The formal statement is as follows.

Theorem 2.3 (von Neumann's minimax theorem)**.** *For two-player zero-sum game with $N = \{l, f\}$ and finite actions for each player, we have*

$$\max_{\sigma_l} \min_{\sigma_f} U_l(\sigma_l, \sigma_f) = \min_{\sigma_f} \max_{\sigma_l} U_l(\sigma_l, \sigma_f) = v. \tag{2.3}$$

The value v is also referred as the value of the game. In this thesis, however, we encounter a zero-sum game on the network where an action of the attacker (follower) is the flow on the network. Thus, the action space of the attacker is infinite, and the

convergence of the double oracle cannot be guaranteed. More importantly, the direct adoption of double oracle is extremely inefficient. In other words, the strategy space of the attacker is no longer a simplex, while a general convex set instead. Therefore, this thesis proposes a generalized version of double oracle framework which is based on the general minimax theorem, called Sion's minimax theorem [57].

Theorem 2.4 (Sion's minimax theorem)**.** *Let X be a compact convex subset of a linear topological space and Y a convex subset of a linear topological space. If f is a real-valued function on $X \times Y$ with*

- *$f(x, \cdot)$ upper semi-continuous and quasi-concave on Y, $\forall x \in X$, and*
- *$f(\cdot, y)$ lower semi-continuous and quasi-convex on X, $\forall y \in Y$*

then,

$$\min_{x \in X} \sup_{y \in Y} f(x, y) = \sup_{y \in Y} \min_{x \in X} f(x, y). \tag{2.4}$$

This thesis will adopt the following weak version of Sion's minimax theorem.

Theorem 2.5 (weak version of Sion's minimax theorem)**.** *Let $X \subset \mathbb{R}^n$ and $Y \subset \mathbb{R}^n$ be compact convex sets. If $f : X \times Y \to \mathbb{R}$ is a continuous function that is convex-concave, i.e.*

- *$f(\cdot, y) : X \to \mathbb{R}$ is convex for fixed y, and*
- *$f(x, \cdot) : Y \to \mathbb{R}$ is concave for fixed x*

then,

$$\min_{x \in X} \max_{y \in Y} f(x, y) = \max_{y \in Y} \min_{x \in X} f(x, y). \tag{2.5}$$

2.1.3 Coalitional Games

In the previous sections, we focus on illustrating the non-cooperative games. In the real world, it is common for agents to cooperate with others through binding agreements, such as contracts, about the distribution of payoffs or the choice of strategies. The

cooperative game theory studies the rational outcome in such content and has attracted significantly increasing attentions in recent years. The cooperative game is also called *coalitional game*, which is populated with non-empty set $N = \{1, ..., n\}$ of players. A *coalition* is a subset of players N, denoted as C. The *grand coalition* is the set N of all players. The widely adopted representation of the game in cooperative game theory is the characteristic function game, defined as follows.

Definition 2.6. A *characteristic function game* is featured with a pair (N, v), where $N = \{1, ..., n\}$ is a finite, non-empty set of players, and $v : 2^N \to \mathbb{R}$ is a *characteristic function*, which maps each coalition $C \subseteq N$ to a real number $v(C)$. The number $v(C)$ is usually referred as the *value* of the coalition C.

Given a coalitional game, we are interested in the possible outcomes that self-interested and rational agents agree with. An outcome of a characteristic function consists of two parts:

- a partition of players into disjoint coalitions, called a *coalition structure*; and
- a *payoff vector*, which distributes the value of each coalition among its members.

The formal definition of an outcome is as follows.

Definition 2.7. Given a characteristic function game $G = (N, v)$, a *coalition structure* over N is a collection of non-empty subsets $CS = \{C^1, ..., C^k\}$ such that

- $\bigcup_{j=1}^{k} C^j = N$, and
- $C^i \cap C^j = \emptyset$ for any $i, j \in \{1, ..., k\}$ such that $i \neq j$.

A vector $\mathbf{y} = (y_1, ..., y_n) \in \mathbb{R}^n$ is a *payoff vector* for a coalition structure $CS = \{C^1, ..., C^k\}$ over N if

- $y_i \geq 0$ for all $i \in N$, and
- $\sum_{i \in C^j} y_i = v(C^j)$ for any $j \in \{1, ..., k\}$.

The pair $(CS, \mathbf{y})$ form an outcome of the game.

One subclass of the characteristic function game that has nice properties is the *superadditive game*. The *superadditivity* is a strong property which makes it always profitable for any pair of coalitions to join forces.

Definition 2.8. A characteristic function game $G = (N, v)$ is said to be *superadditive* if it satisfies $v(C \cup D) \geq v(C) + v(D)$ for every pair of disjoint coalitions $C, D \subseteq N$.

Stability is a critical measurement of an outcome. A stable outcome means that no subset of players has an incentive to deviate. The solution concept for stability is the *core*, which is the set of stable outcomes.

Definition 2.9. The *core* $\mathcal{C}(G)$ of a characteristic function game $G = (N, v)$ is the set of all outcomes $(CS, \mathbf{y})$ such that $\sum_{i \in C} y_i \geq v(C)$ for every $C \subseteq N$.

2.1.3.1 Coalitional Skill Games

The coalitional game model proposed in this thesis is based on the *coalitional skill game* [58], which is composed of a set $N = \{1, ..., n\}$ of agents, a set T of tasks, and a set S of skills. Each agent i has a set of skills $S_i \subset S$, and each task $t \in T$ requires a set of skills $S(t) \subset S$. The set of skills a coalition C has is denoted by $S(C) = \cup_{i \in C} S_i$. A coalition C can conduct task t if $S(t) \subseteq S(C)$, and the set of tasks the coalition C can perform is $T(C) = \{t \in T | S(t) \subseteq S(C)\}$. There is a *task value function* $u : 2^T \rightarrow \mathbb{R}$ maps a subset of the tasks a coalition achieves into a real number. In the characteristic function game representation, the characteristic function of the coalitional skill game maps each coalition $C \subseteq N$ to a real number $v(C)$ such that $v(C) = u(T(C))$.

2.2 Online Convex Optimization

The online convex optimization and more broadly, the online learning, are domains closely related to the game theory, especially the repeated games. The online convex optimization can be casted into a process of repeated interactions between a *learner* and

an *adversary*. The online convex optimization is studied based on the following elegant model:

Online Convex Optimization (OCO)

input: A convex set $S \subset \mathbb{R}^n$

for $t = 1, 2, ...$

predict a vector $\mathbf{w}^t \in S$

receive a convex loss function $f^t : S \to \mathbb{R}$

suffer loss $f^t(\mathbf{w}^t)$

At iteration t, the learner picks a decision point $\mathbf{w}^t$ from a convex set S. After the learner has committed to this choice, a convex loss function $f^t \in \mathcal{F}$ is revealed by the adversary. Here $\mathcal{F}$ is the bounded family of loss functions available for the adversary. The learner thus suffers a loss $f^t(\mathbf{w}^t)$. In the *full information* setting, the learner will then know the loss function f^t chosen by the adversary, while in the *bandit* setting, the learner has no knowledge of the loss function f^t except the value $f^t(\mathbf{w}^t)$ on the chosen decision point. There are two types of adversary in the literature. The *oblivious* adversary is the one who decides the sequence of loss functions for $t = 1, ..., T$ at the very beginning of the process and cannot change them during the play. The *adaptive* adversary, on the other hand, determines the loss function f^t ahead of each iteration t with full knowledge of the learner's previous predictions. This thesis focuses on the bandit setting with adaptive adversary which is suitable for the domain studied here.

Let $\mathcal{H}_t$ denote the history information by time t (inclusive), i.e., $\mathcal{H}_t = (f^1, \mathbf{w}^1, ..., f^t, \mathbf{w}^t)$, and let $\mathcal{H}_t^l$ be the history information *available* for the learner by time t, i.e., $\mathcal{H}_t^l = (\mathbf{w}^1, f^1(\mathbf{w}^1), ..., \mathbf{w}^t, f^t(\mathbf{w}^t))$. Let $\mathcal{V}$ be an adaptive adversary model which returns a possibly randomized loss function $f^t \in \mathcal{F}$ given the history information $\mathcal{H}_{t-1}$. The goal in online convex optimization is to propose an algorithm $\mathcal{A}$ for the learner to make the sequential decisions, which provides a possibly randomized decision point $\mathbf{w}^t$ in set S given a certain game history $\mathcal{H}_{t-1}^l$ for the learner. The widely adopted metric for the performance of the algorithm $\mathcal{A}$ in online convex optimization is *regret*, which measures the difference between the loss the learner has incurred and

that of the best fixed decision in hindsight. Given an (adaptive) adversary model $\mathcal{V}$ and an online algorithm $\mathcal{A}$, for one trail of learning process with whole history information $f^1, \mathbf{w}^1, ..., f^T, \mathbf{w}^T$, the regret of the algorithm $\mathcal{A}$ for this trail is defined as

$$R_T(\mathcal{H}_T) = \sum_{t=1}^{T} f^t(\mathbf{w}^t) - \min_{\mathbf{w} \in S} \sum_{t=1}^{T} f^t(\mathbf{w}). \tag{2.6}$$

Due to the possible randomness inside the algorithm $\mathcal{A}$ and the adversary model $\mathcal{V}$, we are interested in minimizing the expected regret, defined as

$$R_T(\mathcal{V}, \mathcal{A}) = \mathop{\mathbb{E}}_{\mathcal{H} \sim \langle \mathcal{V}, \mathcal{A} \rangle} [R_T(\mathcal{H}_T)]. \tag{2.7}$$

Intuitively, an algorithm performs well if the expected regret is sublinear as a function of T against any adversary model, i.e., $R_T(\mathcal{V}, \mathcal{A}) = o(T)$, since this implies that on average the algorithm performs as well as the best fixed strategy in hindsight.

2.2.1 Online Linear Optimization

Online linear optimization is a subclass of online convex optimization where at each round t, the loss function $f^t(\mathbf{w}) = \mathbf{c}^t \cdot \mathbf{w}$ for some cost vector $\mathbf{c}^t$. Two algorithms for online linear optimization related to this thesis are presented here: *Follow the Perturbed Leader (FPL)* in the full information setting and *BGA*, standing for *Bandit-style Geometric decision algorithm against an Adaptive adversary* for the bandit setting.

2.2.1.1 Follow the Perturbed Leader

In full information setting, a straightforward policy is always picking the decision point which is the best fixed point so far, i.e., $\mathbf{w}^t \in \arg\min_{\mathbf{w} \in S} \sum_{k=1}^{t-1} \mathbf{c}^k \cdot \mathbf{w}$. This simple policy is termed as *Follow the Leader (FTL)*. However, such policy can perform extremely worse sometimes. Consider $S = [-1, 1]$, let $f^1(w) = \frac{1}{2}w$, and let f^t for $t = 2, ..., T$ alternate between $-w$ or w. It is easy to see that FTL solution will keep shifting between $w^t = -1$ and $w^t = 1$, always making the wrong choice. The intuition behind

Algorithm 2: Follow the Perturbed Leader

Parameter: ϵ, $S \subset \mathbb{R}^n$
for $t = 1, ..., T$ **do**
  Choose $\mathbf{z}^t \in \mathbb{R}^n$ at random from the cube $[0, \frac{1}{\epsilon}]^n$
  Return decision point $\mathbf{w}^t \in \arg\min_{\mathbf{w} \in S} (\sum_{k=1}^{t-1} \mathbf{c}^k + \mathbf{z}^t) \cdot \mathbf{w}$

the potential poor performance of FTL is that the solution is not stable. Various measures are available to mitigate this issue. By posing a regularization term on the loss functions, we obtain the *Regularized Follow the Leader (RFTL)* algorithm. In this thesis, we discuss another algorithm in detail, called *Follow the Perturbed Leader (FPL)*. FPL poses a randomized noise on the cumulative loss function and picks the best fixed point in hindsight according to the perturbed loss. This regularizes the strategy shifting between consecutive iterations. FPL is illustrated in Algorithm 2. With delicate setting of parameters, FPL can achieve $\mathcal{O}(\sqrt{T})$ regret as a function of T.

2.2.1.2 BGA

Algorithm 3: BGA

Parameter: γ and ϵ, where ϵ is a parameter of GEX
Fixed a basis $B = \{\mathbf{b}^1, ..., \mathbf{b}^n\} \subseteq S$
for $t = 1, ..., T$ **do**
  Let $\chi^t = 1$ with probability γ and $\chi^t = 0$ otherwise
  if $\chi^t = 0$ **then**
    Select $\mathbf{w}^t$ from the distribution GEX($\hat{\mathbf{c}}^1, ..., \hat{\mathbf{c}}^{t-1}$)
    Incur loss $l^t = \mathbf{c}^t \cdot \mathbf{w}^t$
    $\hat{\mathbf{c}}^t = \mathbf{0} \in \mathbb{R}^n$
  else
    Draw j uniformly at random from $\{1, ..., n\}$
    $\mathbf{w}^t = \mathbf{b}^j$
    Incur loss $l^t = \mathbf{c}^t \cdot \mathbf{w}^t$
    Define $\hat{\ell}^t$ by $\hat{\ell}^t_i = 0 \; \forall i \neq j$ and $\hat{\ell}^t_j = (n/\gamma) l^t$
    $\hat{\mathbf{c}}^t = (B^\dagger)^{-1} \hat{\ell}^t$

The major challenge in designing the low-regret algorithm for bandit setting is the well-known *exploration vs. exploitation* tradeoff. In bandit setting, learner usually keeps an estimation of the previous loss functions. BGA is an algorithm with separated exploration and exploitation processes. In BGA, the exploitation means applying the

full information algorithm such as FPL on the "imagined" online linear optimization process with estimated loss functions instead. The exploration, on the other hand, refers to randomized sampling of decision to obtain an accurate estimation of the loss function.

BGA is illustrated in detail in Algorithm 3. B is a basis in S. With probability γ, BGA randomly samples a base $\mathbf{b}^j$ from B, and incurs loss l^t. $\hat{\ell}^t$ is an estimator of ℓ^t which is defined as follows. $\ell_i^t = \mathbf{c}^t \cdot \mathbf{b}^i$ for $i = 1, ..., n$. Since BGA works on bandit setting, the learner only observes ℓ_j^t which is l^t. $\hat{\ell}^t$ is defined in the way shown in the algorithm. It is easy to verify that $\hat{\ell}^t$ is an unbiased estimator of ℓ^t since $\mathop{\mathbb{E}}_{\gamma,j}[\hat{\ell}_i^t] = P(\gamma = 1) \cdot P(j = i)\frac{n\mathbf{c}^t \cdot \mathbf{b}^i}{\gamma} = \ell_i^t$ for any $i \in \{1, ..., n\}$. Thus, $\hat{\mathbf{c}}^t$ is also an unbiased estimator of the cost $\mathbf{c}^t$. With probability $1 - \gamma$, BGA applies the GEX (*Geometric Experts algorithm*) for the full observation problem on the "imagined" learning process with estimated cost $\hat{\mathbf{c}}^t$ instead. In this thesis, we adopt the FPL as the realization of GEX. With delicate setting of parameters, BGA achieves $\mathcal{O}(T^{2/3})$ regret.

2.3 Security Games

One critical challenge in most security domains is the insufficiency of security resources, such as patrollers, inspection facilities, checkpoints and so on. The security game applies the game theoretical methodology to propose the solution for the security agency on the allocation of limited resources. The Stackelberg security game (SSG) is the most widely adopted model in many security domains. In the basic SSG model, there is a set of targets protected by the defender with limited resources against a sophisticated attacker. Such targets include public infrastructures, ports and air ports, wildlife and nature reserves, and so on. The pure strategy of the defender is thus a resource allocation and the attacker's pure strategy is simply a target to attack. The mixed strategies of both players follow the natural definition shown in Section 2.1. Note that this basic model can be generalized to many complex security problems, where the strategies of both players are more sophisticated, as we will show in the following chapters of this thesis. SSG adopts the Stackelberg game to model the interaction between a defender

(i.e. leader) and an attacker (i.e. follower), where the defender moves first and commits to a mixed strategy, while the attacker observes the defender's mixed strategy[1] and takes a best response. The solution concept adopted in SSG is the strong Stackelberg equilibrium (Definition 2.2).

The SSG model has been extensively studied and used by various security applications in the past decade. Inspired by Conitzer and Sandholm who study the computational complexity of the optimal commitment in Stackelberg games [59], the DOBSS algorithm is proposed to (implicitly) solve the strong Stackelberg equilibrium in Bayesian security games with a mixed-integer quadratic program [9]. DOBSS becomes the core of the ARMOR system which helps to schedule the allocation of checkpoints at the exits of the LAX airport against the adversary of unknown type [2]. The term *Stackelberg security game* is first adopted by Kiekintveld et al. who also explicitly use the strong Stackelberg equilibrium as the solution concept, which then becomes the standard solution in SSG [10]. Since then, a vast amount of efforts are put into designing scalable algorithms to address the exponentially explosive defender's strategy space, in order to solve the realistic-sized problems. To name a few, the ERASER solution algorithm exploits the special payoff structure of the security games to improve the efficiency [10], and the ASPEN algorithm applies the branch and price methodology to solve large scale security games with resource constraints [11]. Those approaches are widely adopted in many deployed systems, such as IRIS for dispatching air marshals for Federal Air Marshal Service (FAMS) [3], and GUARDS for national scale airport protection [5].

2.3.1 Robustness and Human Behavior Modelling in Security Games

Since the great initial success of SSG in protecting airports and flights [2, 3, 5], the uncertainties and human behavior modelling attract significant attentions in security games, for the aim of more realistic models. Various types of uncertainties may exist in

[1] The attacker's full observability on the defender's mixed strategy is commonly justified by the unlimited surveillance of pure strategy realizations [2, 10].

security games. To name a few, the attacker in standard SSG has access to the defender's mixed strategy, as the attacker can conduct enough observations of the realization of the mixed strategy. However, the attacker in reality only has limited surveillance capability and the observation uncertainty exists. Besides, the attacker in standard SSG is assumed to be perfectly rational, while in the real world, the attacker may deviate from his optimal response unless the marginal benefit of doing so is large enough. Pita et al. [14] make the first attempt in security games to address such uncertainties, and propose programs based on the maximin metric to solve for the robust solution. Nguyen et al. [15] propose the minimax regret, instead of maximin-based solution, as a metric of robustness and consider the uncertainty of the defender's evaluation of the payoff structure of the attacker. An et al. consider the cost of surveillance for the attacker and model the attacker's decision-making with a Markov Decision Process [60, 61]. Jiang et al. [7] study the uncertainty of the execution of the defending measure (interruptions) and model such uncertainty using the Markov Decision Processes. The corresponding system, called TRUST, is deployed in LA metro rail system for scheduling the patrols.

Human behavior modelling is an important technique to exploit the attacker's potential deviation from the best response for better defender strategy. The related works on human behavior modelling in security games are mostly based on two fundamental theories: Prospect Theory [12] and Quantal Response Equilibrium (QRE) [13]. Yang et al. are the first to introduce the Quantal Response (QR) model into the security game context [16]. Efficient algorithm to compute the QRE in security games is proposed in [17] and the PROTECT system based on QRE solution is deployed to protect the ports of the United States [6]. While the quantal response is based on the expected utility, Nguyen et al. integrate the so called subjective utility function (SU) into QR model and the resulting SUQR model shows superior predictive power over QR model as it captures the dependency of the human's stochastic behavior on the individual information, rather than the expected utility [62]. The SUQR model is applied in PAW system to improve the wildlife ranger patrols and protect the wildlife reservation [8].

2.3.2 Learning and Planning in Repeated Security Games

In recent years, the so called *green security games* [18] have drawn significant attentions in the protection of nature reservation and fishery and prevention of illegal poaching and fishing. The green security domains share several properties which differ them from the protection of public infrastructure, and one critical property of them is that the interactions between adversaries (i.e. poachers and illegal fishers) and the defender are frequent and repeated. Such repeated interactions make it possible for the defender to learn from history the adversary's behavior and to plan against the unknown adversary. Qian et al. consider the asymmetric information of the game that the protectors may have less information about the state such as the distribution and richness of the resources than extractors [19]. The adversary behavior model is fixed empirically according to the historical observations, and the planning for the defender is modelled as a POMDP. Kar et al. propose a model named SHARP which models the adaptive behavior of the adversary in repeated security games [50].

Learning Stackelberg strategy in repeated games has received increasing attentions recently. Marecki et al. study the repeated Bayesian Stackelberg game where the leader has a prior belief about the distribution of follower type and plays mixed strategy each time [20]. The Monte-Carlo tree search algorithm is adopted to balance the exploration and exploitation. Letchford et al. propose a learning problem in repeated Stackelberg games where the leader samples a probability distribution from the simplex and observes the follower's best response [21]. The goal of the learning problem is to minimizing the number of calls to the follower best response oracle necessary to finding the optimal Stackelberg mixed strategy for the leader. It is shown that polynomial number (in the representation of the game) of queries to the follower best response oracle are sufficient to learn a sufficiently near-optimal strategy [22]. Xu et al. apply the online learning framework, in particular, an FPL variant algorithm, to solve for a low-regret policy for the defender in repeated security games with no prior knowledge of the adversary's behavior pattern, by casting the repeated security games as an online combinatorial multi-armed bandit problem [23].

2.3.3 Network-structured Security Games

The previously discussed security domains can be abstracted as target-based domains, while a lot of others domains can be abstracted as networks, or graphs, such as preventing the illegal drug smuggling on the transportation network, interdicting the escaping path of a bank robber, stopping the spread of negative influence on the social network, and so on. The security game on such domain is played on a network $G = (V, E)$, where V is the vertex set and E represents a set of edges, either directed or undirected. Each edge may be associated with a per-unit flow cost and a capacity. The attacker tries to optimize its objective over the network, such as the shortest path, the maximum flow, and the minimum cost flow. The defender aims to alter the network to maximally impair the attacker's objective. For example, the defender can allocate the checkpoints on the edges to interdict probably a proportion of the network flow, or increase the cost on the edges. The network-structured security games are mostly zero-sum. The general minimax formulation of network-structured security games can be represented as $\min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} U_f(\mathbf{x}, \mathbf{y})$ where $\mathbf{x}$ and $\mathbf{y}$ represent the defender's strategy and attacker's strategy respectively. For example, in the case of allocating checkpoints on edges to interdict the attacker's maximum network flow, $\mathbf{x}$ can be an allocation and $\mathcal{X}$ is the set of all combinatorial allocations, or $\mathbf{x}$ can be a probability distribution over all possible allocations, i.e., a mixed strategy, and $\mathcal{X}$ thus becomes a simplex; $\mathbf{y}$ is a network flow and $\mathcal{Y}$ is the set of feasible network flows satisfying the capacity constraints. The realization of the general formulation is highly problem-dependent.

Network-structured security problems have been studied in a limited scope. In the *network interdiction problem* [47], we are given a weighted, directed or undirected graph, and our goal is minimizing the flow through the graph via deletion of edges/nodes (other goals — such as maximizing the shortest path, increasing detection probability — and other interdiction methods — such as decreasing edge capacity — have been studied as well [24–28]. Other works study *stochastic network interdiction* where the interdiction action is successful with some known probability p [63, 64]. Most of the network interdiction problem work, if not all, focus on the pure defender strategy or deterministic measure. The recent progress of Stackelberg security games inspires

computing the optimal mixed defender strategy in realistic-sized problem instances. Jain et al. model the urban security problem as a network security game where the defender allocates limited checkpoints on roads to interdict an escaping adversary who selects path to certain exit nodes [29]. Due to the exponentially large strategy spaces for both players, a double oracle framework (see Sec. 2.1.2.1) is applied to solve for the minimax mixed strategy for the defender. Efficient oracles are then proposed to scale up to realistic-sized urban security instances [48]. The double oracle framework then becomes a standard technique to solve for the optimal mixed strategy for the defender in network security games. For example, Wang et al. focus on the randomized monitoring of suspects to detect a potential terrorist plots [30]; Yin and An study the patrol of marine protected areas and build a transition graph to take into account the time factor [32].

2.4 Chapter Summary

This chapter reviews the background materials for understanding this thesis, which are composed by game theoretical reasoning, large-scale optimization and online convex optimization. We demonstrate the fundamental concepts in game theory, including the Nash equilibrium, the Stackelberg equilibrium, the minimax theorem and the coalitional games. When it comes to the repeated game plays, the online convex optimization techniques are introduced, especially the online linear optimization, which is simple to model various game structures. We illustrate two popular online learning algorithms, follow the perturbed leader and BGA, which will play an important role in Chapter 5, where we propose a BGA variant called SBGA by exploiting the semi-bandit feedback of a specific network security domain. Giving an overview of the security games literature, this thesis contributes to the security games community by significantly enlarging the scope the network security research and introducing various novel models and methodologies for several critical network security domains.

Chapter 3

Combating the Cooperation of Multiple Adversaries on the Network

This chapter addresses the problem of interdicting the cooperation of multiple attackers to alleviate the potential risk. There is 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 [65]. To name a few, three terrorist groups in Africa have been reported to share funds, training, and explosive materials with each other [34], and Chechen terrorists were reported to obtain weapons from terrorist organizations in the Middle East [35]. 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 means 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 [37–40], or using cooperative game theory as a means for identifying key members of terrorist networks [41, 42]. 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) [4, 43–45]. The ability and proclivity of attackers to form coalitions is, therefore, largely ignored within the security games research.

This chapter makes several contributions for combating cooperative adversaries. First, a formal Coalitional Security Game (CSG) model is introduced to address the problem of optimally inhibiting formation of attack coalitions. Second, it is shown 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. Third, to overcome the computational challenges, a sophisticated branch and price algorithm is developed, 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), the performance of the algorithm is further improved 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, an interior-point stabilization method is presented to improve convergence properties of column generation by generating an interior dual optimal solution of the master problem, associated with novel heuristics for generating multiple columns in each iteration to support the linear relaxation approximation and further speed up the column generation. Finally, extensive experimental results show that these 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

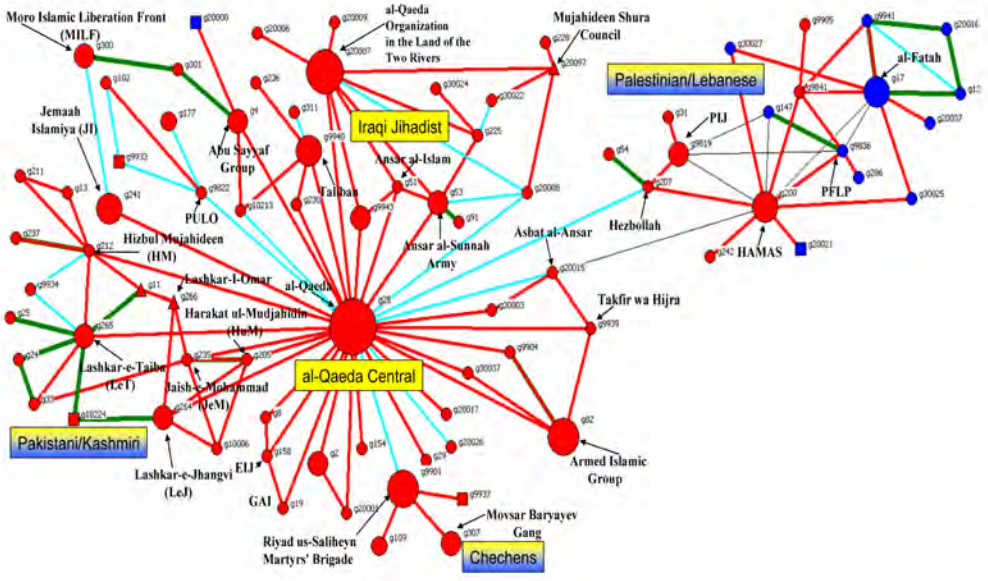


FIGURE 3.1: Mideast terrorist network [1].

heuristic search algorithms significantly; and 3) are remarkably robust against uncertainties of attackers' payoffs.

The rest of the chapter is organized as follows: Section 3.1 illustrates the motivating domain in this work. Section 3.2 presents the Coalitional Security Game (CSG) model. The complexity of CSG is analyzed in Section 3.3. Section 3.4 provides the branch and price framework to solve the defender's decision-making problem. The key component inside the framework— the column generation— is explained in detail in Section 3.5. Section 3.6 shows the experimental evaluation of proposed methods. Finally, the chapter is summarized in Section 3.7.

3.1 Cooperation Among Terrorist Groups and Organizations

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 [65]. 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 [36]. Levitt [36] 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 [66]. 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 [67].

Since cooperation among terrorist groups is achieved through communication channels, such as front companies, charities for transferring funds [36], 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 terror attacks abroad [68]. More broadly, 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"* [33].

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 among terrorist groups [36]. 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.

3.2 Coalitional Security Games

A *coalitional security game* (CSG) is modeled based on a *coalitional skill game* [58] (Sec. 2.1.3.1), and consists of T types of targets, each with a large number of copies.[1] 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 [42], 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 [69], 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

[1] This is a reasonable assumption, since there is a large pool of potential targets of terrorism around the world.

$\mathbf{a}$ is feasible if

$$\sum\nolimits_{t\in T,\, s\in S(t)} a_t \leq \sum\nolimits_{i\in C,\, s\in S_i} m_{is},\ \forall s \in S, \tag{3.1}$$

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. The value $v(C)$ of a coalition C is defined as the maximum achievable payoff, i.e.,

$$v(C) = \max\nolimits_{\mathbf{a}\text{: satisfying Eq.(3.1)}}\, u(\mathbf{a}) \tag{3.2}$$

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.

3.2.1 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 in this sense, we adopt the widely-used solution concept of ϵ-*core* [70].

Definition 1 (ϵ-**core**). The ϵ-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 $(CS, \mathbf{y})$ in ϵ-core with as lower ϵ value as possible, and the minimal value ϵ^* for which the ϵ-core is non-empty is called the *least-core value* of the game, with the corresponding ϵ^*-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 [70], given the coalition value function v defined in Eq.(3.2), the coalitional game turns out to be *superadditive* (LEMMA 3.1). 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 3.2).

Lemma 3.1. *Given the value function defined in Eq.(3.2), 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 3.2. *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. □

3.2.2 Defender Strategy

The defender's goal is to optimally cut off connections among 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, λ_{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

$$U_d(\mathbf{B}) = -\sum\nolimits_{C \in CS^*(\mathbf{B})} v(C) - \sum\nolimits_{(i,j) \in E} B_{ij}\lambda_{ij} \tag{3.3}$$

3.3 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.

3.3.1 Defender's Decision-making Problem

We reduce the *k-TERMINAL CUT* problem [71], 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) [72]. The following definitions will be useful to introduce our key result (THEOREM 3.3).

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

Proof. Let a and b be instances of 3-TERMINAL CUT and defender's decision-making problem respectively[1]. 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 λ_{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 $S(t_4) = \{s_1, s_2, s_3, s_4\}$; and $p_1 = p_2 = p_3 = M > W_{max} \cdot |E|$ and $p_4 = 2M$ in which $W_{max} = \max_{(i,j) \in E} \lambda_{ij}$.

Obviously, f reduces a to b in polynomial time. For a given defender strategy $\mathbf{B}$, the cost[2] 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 two terminal attackers

[1] Our procedure of reduction from 3-TERMINAL CUT problem can be easily extended to the reduction from any k-TERMINAL CUT problem with fixed $k \geq 3$.

[2] The cost here is the defender's total loss, i.e., $-U_d(\mathbf{B})$.

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 α 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 the optimal solution of 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 $cost(\mathbf{B}^*) - cost(\mathbf{B}) = \sum_{(i,j)\in E} w_{ij}(B^*_{ij} - B_{ij}) \leq 0$, which means that $\mathbf{B}^*$ is optimal. Otherwise, $\widetilde{F}_{\mathbf{B}} \neq 0$, we have $\widetilde{F}_{\mathbf{B}} \geq M > W_{max} \cdot |E|$. It follows that $cost(\mathbf{B}^*) - cost(\mathbf{B}) < \sum_{(i,j)\in E} w_{ij}B^*_{ij} - W_{max} \cdot |E| \leq 0$, which contradicts that $\mathbf{B}$ is optimal. We conclude that $\mathbf{B}^*$ is optimal to b. Therefore, $opt(b) = opt(a) \Rightarrow \alpha = 1$ is a satisfying constant.

Finally, when $\beta = 1$, $|cost(E') - opt(a)| \leq |cost(E') + \widetilde{F}_{\mathbf{B}} - opt(a)| = \beta|cost(\mathbf{B}) - opt(b)|$ 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 3.3 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.

3.3.2 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 3.4. *Computing 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 $u(\mathbf{a}) \geq |Q|$ if and only if 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 $u(\mathbf{a}) \geq |Q|$ 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 $u(\mathbf{a}) \geq |Q|$ only if EXACT-COVER has a "yes" solution. According to Eq.(3.1), 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.

3.4 Solution Approach

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

3.4.1 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 $x_k = 1$ if $C_k \in CS$ and $x_k = 0$ otherwise. If we suppose that all coalitions in $C_k \in \mathcal{C}$, as well as their associated values v_k, have been precomputed, the defender's optimization problem can be formulated as the following integer program, where the objective is minimizing the defender's loss while Eqs.(3.4b)–(3.4c) ensure that the stable coalition structure is formed:

$$\min_{\mathbf{x},\mathbf{B}} \quad \sum\nolimits_{C_k \in \mathcal{C}} v_k x_k + \sum\nolimits_{(i,j)\in E} B_{ij}\lambda_{ij} \tag{3.4a}$$

$$\text{s.t.} \quad \sum\nolimits_{C_k \in \mathcal{C}} \alpha_{ki} x_k = 1 \qquad i \in N \tag{3.4b}$$

$$\sum\nolimits_{C_k \in \mathcal{C}} \alpha_{ki}\alpha_{kj} x_k \geq 1 - B_{ij} \quad (i,j) \in E \tag{3.4c}$$

$$\mathbf{x} \in \{0,1\}^{|\mathcal{C}|}, \mathbf{B} \in \{0,1\}^{|N|\times|N|} \tag{3.4d}$$

Eq.(3.4b) ensures that each attacker is in exactly one coalition, so that CS denoted by $\mathbf{x}$ is a partition of N. Eq.(3.4c) 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.(3.4b)–(3.4c), 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 3.2.

Although IP (3.4) 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*.

3.4.2 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 (3.4) 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 3.5. *The formulation* (3.4) *is equivalent to its relaxed formulation where* $\mathbf{x}$ *is continuous.*

An auxiliary Proposition is provided for proving Lemma 3.5 as follows:

Proposition 1. For any feasible solution $\mathbf{x}$ of MILP (3.4), x_k is non-zero only if the induced subgraph of C_k is a connected component itself or consists of several connected components of graph $G(\mathbf{B})$ (i.e., the remaining graph after defender's blocking).

Proof of Proposition 1. To prove Proposition 1, we show that for a feasible solution $\mathbf{x}$ with an $x_{\hat{k}} > 0$ corresponding to coalition $C_{\hat{k}}$, if a vertex $i \in C_{\hat{k}}$, then all vertices connected to i in $G(\mathbf{B})$ is also in $C_{\hat{k}}$. Considering the contrary, given a solution $\mathbf{x}$ with $x_{\hat{k}} > 0$ for $C_{\hat{k}}$, if there exists an edge $(\hat{i}, \hat{j})$ not blocked by the defender (i.e., $B_{\hat{i}\hat{j}} = 0$), such that $\hat{i} \in C_{\hat{k}}$ while $\hat{j} \notin C_{\hat{k}}$, i.e., $\alpha_{\hat{k}\hat{i}} = 1$ and $\alpha_{\hat{k}\hat{j}} = 0$, then according to Eq.(3.4c), we have: $\sum_k \alpha_{k\hat{i}}\alpha_{k\hat{j}}x_k = \sum_{k:k\neq\hat{k}} \alpha_{k\hat{i}}\alpha_{k\hat{j}}x_k \geq 1$ since $\alpha_{\hat{k}\hat{j}} = 0$. Therefore, we have $\sum_k \alpha_{ki}x_k = \sum_{k:k\neq\hat{k}} \alpha_{ki}x_k + x_{\hat{k}} \geq \sum_{k:k\neq\hat{k}} \alpha_{k\hat{i}}\alpha_{k\hat{j}}x_k + x_{\hat{k}} \geq 1 + x_{\hat{k}} > 1$, which is a contradiction since $\sum_k \alpha_{ki}x_k = 1$ according to Eq.(3.4b). □

Proof of Lemma 3.5. We will prove Lemma 3.5 by showing that for any binary solution $\mathbf{B}$, the optimal solution $\mathbf{x}$ for the formulation (3.4) is always binary.

Let $\mathcal{K}$ be the set of connected components of $G(\mathbf{B})$, and let $\mathcal{C}'$ be the set of coalitions whose induced subgraphs are connected components themselves or consist of several connected components of $G(\mathbf{B})$. Let K_l denote the l-th connected component in $\mathcal{K}$, and for a coalition $C \in \mathcal{C}'$, let $\beta_l^C = 1$ if the induced subgraph of C is K_l itself or consists of K_l, and $\beta_l^C = 0$ otherwise. According to Proposition 1, a feasible $\mathbf{x}$ can only take positive values on coalitions in $\mathcal{C}'$. Therefore, we can treat the connected components as the basic elements of forming coalitions, and reformulate the solution $\mathbf{x}$ as $\tilde{\mathbf{x}}$ defined on $\mathcal{C}'$. Thus, we have that every connected component K_l has a total coverage of 1,

$$\sum_{C \in \mathcal{C}'} \beta_l^C \tilde{x}_C = 1 \quad \forall K_l \in \mathcal{K} \tag{3.5}$$

and the minimized objective becomes: $\sum_{C\in\mathcal{C}'} v(C)\tilde{x}_C + \sum_{(i,j)\in E} B_{ij}\lambda_{ij}$. For any fixed defender strategy $\mathbf{B}$, since the coalition's value is superadditive, we have that $v(C) \geq \sum_{K_l\in\mathcal{K}} \beta_l^C \tilde{v}(K_l)$ where $\tilde{v}(K_l)$ is the value of coalition whose induced subgraph is K_l. In this case, for any $\mathbf{x}$, we have: $\sum_{C\in\mathcal{C}'} v(C)\tilde{x}_C \geq \sum_{K_l\in\mathcal{K}} \tilde{v}(K_l) \sum_{C\in\mathcal{C}'} \beta_l^C \tilde{x}_C = \sum_{K_l\in\mathcal{K}} \tilde{v}(K_l)$ according to Eq.(3.5), which means that the binary solution $\mathbf{x}$, which takes value of 1 for those coalitions whose induced subgraphs are connected components of $G(\mathbf{B})$ and 0 otherwise, is optimal to relaxed formulation (3.4). □

Figure 3.2 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 (3.4) (with $\mathbf{x}$ being relaxed), where the root node in the tree corresponds to the IP (3.4) and it keeps an *upper bound* (**UB**) on its optimal objective, which can be the objective value of any feasible solution of IP (3.4). 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 (3.4).

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 3.2. 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 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.

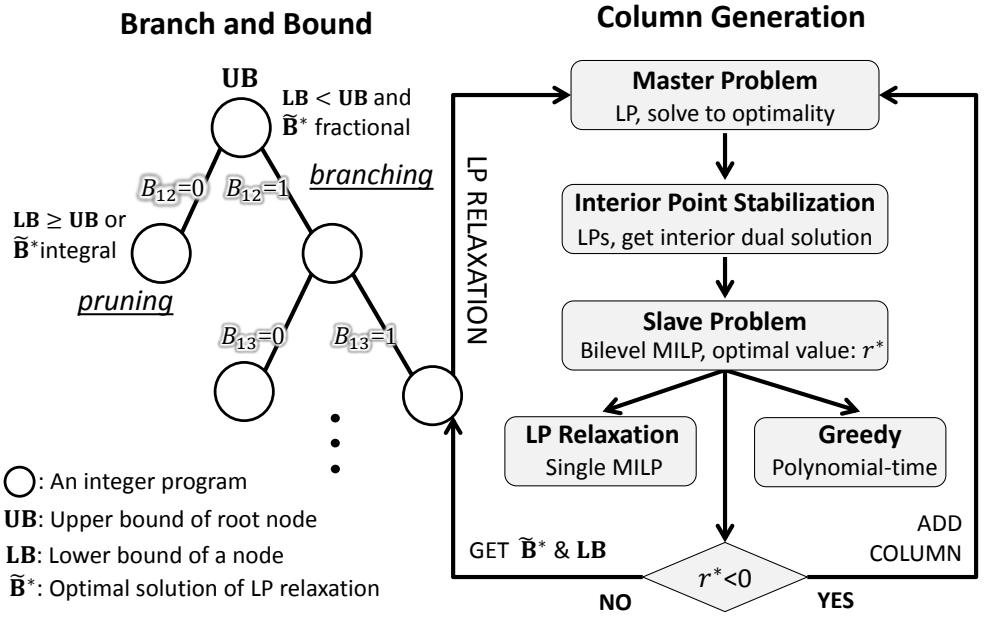


FIGURE 3.2: Branch and price framework.

3.5 Column Generation

The LP relaxation at each node in Figure 3.2 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.

3.5.1 Master Problem

The master problem (3.6) starts with a small set of coalitions $\mathcal{C}' \subset \mathcal{C}$ and solves for the optimal relaxed defender strategy $\tilde{\mathbf{B}}^*$. Note that Eqs.(3.6a)-(3.6c) have ensured $\mathbf{x} \leq \mathbf{1}$ and $\tilde{\mathbf{B}} \leq \mathbf{1}$.

$$\min_{\mathbf{x},\tilde{\mathbf{B}}} \quad \sum\nolimits_{C_k \in \mathcal{C}'} v_k x_k + \sum\nolimits_{(i,j)\in E} \tilde{B}_{ij}\lambda_{ij} \tag{3.6a}$$

$$\text{s.t.} \quad \sum\nolimits_{C_k \in \mathcal{C}'} \alpha_{ki} x_k = 1 \qquad i \in N \tag{3.6b}$$

$$\sum\nolimits_{C_k \in \mathcal{C}'} \alpha_{ki}\alpha_{kj} x_k \geq 1 - \tilde{B}_{ij} \quad (i,j) \in E \tag{3.6c}$$

$$\mathbf{x} \geq \mathbf{0}, \tilde{\mathbf{B}} \geq \mathbf{0} \tag{3.6d}$$

Let $\mathbf{f}$ and $\mathbf{g}$ be dual variables of master constraints (3.6b) and (3.6c) 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.

3.5.1.1 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 [73] 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 the optimal dual 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 [74]. 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. The pseudo-code of IPS is given by Algorithm 4.

Algorithm 4: Interior Point Stabilization (IPS)

Input: M
Output: an interior point $(\mathbf{f}^{int}, \mathbf{g}^{int})$ of $\mathcal{D}$

$Counter = 0$, $\mathbf{f}^{int} = \mathbf{0}$, $\mathbf{g}^{int} = \mathbf{0}$;
while $Counter < M$ **do**
  $Counter = Counter + 1$;
  randomly generate objective coefficient $(\boldsymbol{\mu}, \boldsymbol{\omega})$;
  solve the $(\mathcal{D}^{\boldsymbol{\mu},\boldsymbol{\omega}})$ and $(\mathcal{D}^{-\boldsymbol{\mu},-\boldsymbol{\omega}})$ to get two extreme points $(\mathbf{f}^*_{\boldsymbol{\mu},\boldsymbol{\omega}}, \mathbf{g}^*_{\boldsymbol{\mu},\boldsymbol{\omega}})$ and $(\mathbf{f}^*_{-\boldsymbol{\mu},-\boldsymbol{\omega}}, \mathbf{g}^*_{-\boldsymbol{\mu},-\boldsymbol{\omega}})$;
  $\mathbf{f}^{int} = \mathbf{f}^{int} + (\mathbf{f}^*_{\boldsymbol{\mu},\boldsymbol{\omega}} + \mathbf{f}^*_{-\boldsymbol{\mu},-\boldsymbol{\omega}})/2M$;
  $\mathbf{g}^{int} = \mathbf{g}^{int} + (\mathbf{g}^*_{\boldsymbol{\mu},\boldsymbol{\omega}} + \mathbf{g}^*_{-\boldsymbol{\mu},-\boldsymbol{\omega}})/2M$;
return $(\mathbf{f}^{int}, \mathbf{g}^{int})$;

Once the master problem is solved to optimality, let $\tilde{\mathcal{C}}$ and $\tilde{E}$ be defined as the set of coalitions and edges for which $x_k > 0$ and $B_{ij} > 0$ (i.e., some of the basic columns), and let $\hat{E}$ be the set of edges for which the constraint (3.6c) is not tight. Using complementary slackness conditions, the optimal dual polyhedron $\mathcal{D}$ of master

problem containing all optimal values for $\mathbf{f}$ and $\mathbf{g}$ is defined by

$$\sum_{i \in C_k} f_i + \sum_{(i,j) \in E : i,j \in C_k} g_{ij} \leq v_k \quad \forall C_k \in \mathcal{C}' \backslash \tilde{\mathcal{C}} \tag{3.7a}$$

$$\sum_{i \in C_k} f_i + \sum_{(i,j) \in E : i,j \in C_k} g_{ij} = v_k \quad \forall C_k \in \tilde{\mathcal{C}} \tag{3.7b}$$

$$g_{ij} \leq \lambda_{ij} \qquad \forall (i,j) \in E \backslash \tilde{E} \tag{3.7c}$$

$$g_{ij} = \lambda_{ij} \qquad \forall (i,j) \in \tilde{E} \tag{3.7d}$$

$$g_{ij} \geq 0 \qquad \forall (i,j) \in E \backslash \hat{E} \tag{3.7e}$$

$$g_{ij} = 0 \qquad \forall (i,j) \in \hat{E} \tag{3.7f}$$

$$f_i \in \mathbb{R} \qquad \forall i \in N \tag{3.7g}$$

To obtain an extreme point of $\mathcal{D}$, we can 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 following LP, denoted by ($\mathcal{D}^{\boldsymbol{\mu},\boldsymbol{\omega}}$), with a simplex based method:

$$\min_{\mathbf{f},\mathbf{g}} \quad \sum_{i \in N} \mu_i f_i + \sum_{(i,j) \in E} \omega_{ij} g_{ij} \tag{3.8a}$$

$$\text{s.t.} \quad \text{Eqs.(3.7a)–(3.7g)} \tag{3.8b}$$

Different instances of ($\mathcal{D}^{\boldsymbol{\mu},\boldsymbol{\omega}}$) can be generated by defining multiple objective coefficients $(\boldsymbol{\mu}, \boldsymbol{\omega})$ and thus several extreme points of $\mathcal{D}$ can be obtained. Since $\mathcal{D}$ is a convex set, any convex combination of its extreme points will lie within it. In particular, if we take the average of all obtained extreme points, we should obtain an interior point of $\mathcal{D}$ that gives much more centered dual values. Noticed that in Algorithm 4, for each objective coefficient $(\boldsymbol{\mu}, \boldsymbol{\omega})$ generated randomly, both problems ($\mathcal{D}^{\boldsymbol{\mu},\boldsymbol{\omega}}$) and ($\mathcal{D}^{-\boldsymbol{\mu},-\boldsymbol{\omega}}$) are solved in order to favor the identification of distant extreme points.

3.5.2 Slave Problem

The slave problem finds the best column (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 [74]. The reduced cost r_k of coalition C_k, associated with variable x_k, is given in Eq.(3.9), 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.

$$r_k = v_k - \sum\nolimits_{i \in N} \alpha_{ki} f_i^* - \sum\nolimits_{(i,j) \in E} \alpha_{ki} \alpha_{kj} g_{ij}^* \tag{3.9}$$

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) (3.10), where $(\mathbf{f}^*, \mathbf{g}^*)$ is the optimal dual solution of the master problem (3.6). 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 (3.10), 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.(3.10b) ensures that $\xi_{ij} = \alpha_i \alpha_j$ for edge $(i, j) \in E$. Eqs.(3.10c)–(3.10f) enforce $\boldsymbol{\alpha}$ to induce a connected subgraph of G using flow balance techniques [75]: According to constraints (3.10c) and (3.10d), $w_i = 1$ 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.(3.10f) 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.(3.10e) 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 (3.10g) computes the coalition's value $v(C)$ where Eq.(3.10h) restricts C to have enough skill capacities to conduct the attacking plan $\mathbf{a}$ according to Eq.(3.1).

$$\min_{\substack{\boldsymbol{\alpha},\boldsymbol{\xi},\mathbf{w}\\ \mathbf{h}^+,\mathbf{h}^-}} \quad \sum\nolimits_{t\in T} p_t a_t - \mathbf{f}^{*\mathbf{T}}\boldsymbol{\alpha} - \mathbf{g}^{*\mathbf{T}}\boldsymbol{\xi} \tag{3.10a}$$

$$\text{s.t.} \quad \xi_{ij} \le \alpha_i, \xi_{ij} \le \alpha_j,$$

$$\xi_{ij} \ge \alpha_i + \alpha_j - 1 \quad \forall (i,j) \in E \tag{3.10b}$$

$$w_i \le \alpha_i,$$

$$w_i \le 1 - \alpha_j \quad \forall i,j \in N : j < i \tag{3.10c}$$

$$\sum\nolimits_{i\in N} w_i = 1 \tag{3.10d}$$

$$\sum\nolimits_{j\in\mathcal{N}(i)} h^+_{lij} - \sum\nolimits_{j\in\mathcal{N}(i)} h^-_{lij} \ge w_i - (1-\alpha_l), \quad \forall i,l \in N : i \ne l \tag{3.10e}$$

$$h^+_{lij} \le \xi_{ij},$$

$$h^-_{lij} \le \xi_{ij}, \quad \forall (i,j) \in E, l \in N \tag{3.10f}$$

$$\max_{\mathbf{a}} \quad \sum\nolimits_{t\in T} p_t a_t \tag{3.10g}$$

$$\text{s.t.} \quad \sum\nolimits_{t\in T} \beta_{ts} a_t \le \sum\nolimits_{i\in N} \alpha_i \gamma_{is} m_{is} \quad s \in S \tag{3.10h}$$

$$\boldsymbol{\alpha} \in \{0,1\}^{|N|}, \boldsymbol{\xi} \in [0,1]^{|E|}, \mathbf{a} \in \mathbb{N}^{|T|}$$

$$\mathbf{h}^+ \ge \mathbf{0}, \mathbf{h}^- \ge \mathbf{0}, \mathbf{w} \ge \mathbf{0}. \tag{3.10i}$$

The key challenge of the BMILP is that the inner maximization program described in (3.10g) 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.

3.5.2.1 Linear Relaxation Approximation

To reformulate the BMILP to an MILP, we first relax the integer program (3.10g) of computing the coalition's value in the lower level, such that the decision variable $\mathbf{a} \in \mathbb{N}^{|T|}$ is relaxed to $\tilde{\mathbf{a}} \geq \mathbf{0}$. For this relaxed linear program (RLP), let $\mathbf{u}$ be the dual variable associated with constraint (3.10h). 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 [74]:

$$(\sum\nolimits_{s\in S} \beta_{ts} u_s - p_t) \cdot \tilde{a}_t = 0 \quad \forall t \in T \tag{3.11a}$$

$$(\sum\nolimits_{i\in N} \alpha_i \gamma_{is} m_{is} - \sum\nolimits_{t\in T} \beta_{ts} \tilde{a}_t) \cdot u_s = 0 \quad \forall s \in S \tag{3.11b}$$

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 (3.12), with Eq.(3.12c) corresponding to the dual constraint of the RLP, and Eqs.(3.12d)–(3.12e) equivalent with the complementary slackness conditions (3.11). 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).

$$\min_{\substack{\boldsymbol{\alpha},\boldsymbol{\xi},\mathbf{w},\mathbf{u},\boldsymbol{\phi}\\ \mathbf{h}^+,\mathbf{h}^-,\tilde{\mathbf{a}},\boldsymbol{\varphi}}} \quad \sum\nolimits_{t\in T} p_t \tilde{a}_t - \mathbf{f}^{*\mathbf{T}}\boldsymbol{\alpha} - \mathbf{g}^{*\mathbf{T}}\boldsymbol{\xi} \tag{3.12a}$$

$$\text{s.t.} \quad \text{Eqs.(3.10b)–(3.10f), (3.10h)–(3.10i)} \tag{3.12b}$$

$$\sum\nolimits_{s\in S} \beta_{ts} u_s \geq p_t, \quad \forall t \in T \tag{3.12c}$$

$$\sum\nolimits_{i\in N} \alpha_i \gamma_{is} m_{is} - \sum\nolimits_{t\in T} \beta_{ts} \tilde{a}_t \leq M(1-\phi_s),$$
$$u_s \leq M\phi_s \quad \forall s \in S \tag{3.12d}$$

$$\sum\nolimits_{s\in S} \beta_{ts} u_s - p_t \leq M(1-\varphi_t),$$
$$\tilde{a}_t \leq M\varphi_t \quad \forall t \in T \tag{3.12e}$$

$$\tilde{\mathbf{a}} \geq \mathbf{0}, \mathbf{u} \geq \mathbf{0}$$

$$\boldsymbol{\phi} \in \{0,1\}^{|S|}, \boldsymbol{\varphi} \in \{0,1\}^{|T|}. \tag{3.12f}$$

Lemma 1. The coalition value $\tilde{v}(C)$ computed by LP relaxation of IP (3.10g) is bounded by $\tilde{v}(C) \leq (1+|S|)v(C)$.

Proof. We first prove the following fact: For the LP relaxation of IP (3.10g), 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.

1. $v(C) = 0$. In this case, the coalition C cannot attack any target due to the limitation of skill capacities, i.e., for any target type $t \in T$, there exists 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 (3.10g), any fractional solution $\tilde{a}$ with $\tilde{a}_t > 0$ is infeasible for violating the skill capacity constraint (3.10h) of skill s^t: $\sum_{t \in T} \beta_{ts^t} \tilde{a}_t \leq 0$, so that $\tilde{v}(C) = v(C) = 0$.

2. $v(C) > 0$. Let $\mathbf{a}^*$ be the optimal integer solution for IP (3.10g), and $\tilde{\mathbf{a}}^*$ be the optimal fractional integer solution for LP relaxation of IP (3.10g). According to the property of multidimensional knapsack problem [76], $\tilde{\mathbf{a}}^*$ can take fractional values in at most $|S|$ coordinates, which means that, 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 $\tilde{v}(C)$ computed by LP relaxation of IP (3.10g) is bounded by $\tilde{v}(C) \leq (1+|S|)v(C)$. □

Theorem 3.6. *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 (3.4) in which $\tilde{v}_k$ is the value of C_k computed by LP relaxation of IP (3.10g):

$$\min_{\mathbf{x},\mathbf{B}} \quad \sum\nolimits_{C_k \in \mathcal{C}} \tilde{v}_k x_k + \sum\nolimits_{(i,j)\in E} B_{ij}\lambda_{ij} \tag{3.13a}$$

$$\text{s.t.} \quad \text{Eqs. (3.4b)–(3.4d)} \tag{3.13b}$$

Since $\mathbf{B}'$ is optimal for IP (3.13) and $v(C) \leq \tilde{v}(C) \leq (1+|S|)v(C)$, we have:$-U_d(\mathbf{B}') = \sum_{C\in CS^*(\mathbf{B}')} v(C) + \sum_{(i,j)\in E} {B'}_{ij}\lambda_{ij} \leq \sum_{C\in CS^*(\mathbf{B}')} \tilde{v}(C) + \sum_{(i,j)\in E} {B'}_{ij}\lambda_{ij} \leq \sum_{C\in CS^*(\mathbf{B}^*)} \tilde{v}(C) + \sum_{(i,j)\in E} B^*_{ij}\lambda_{ij} \leq (1+|S|)\sum_{C\in CS^*(\mathbf{B}^*)} v(C) + \sum_{(i,j)\in E} B^*_{ij}\lambda_{ij} \leq (1+|S|)(\sum_{C\in CS^*(\mathbf{B}^*)} v(C) + \sum_{(i,j)\in E} B^*_{ij}\lambda_{ij}) = -(1+|S|)U_d(\mathbf{B}^*)$. Therefore, we have $U_d(\mathbf{B}') \geq (1+|S|)U_d(\mathbf{B}^*)$. □

3.5.2.2 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 5). This heuristic is fast, and has an important advantage of enabling generation of multiple coalitions (columns) in a single iteration of the column generation algorithm (hence, the name multi-start), which significantly reduces the number of column generation iterations.

Algorithm 5: Greedy with Multi-Start (GMS)

Input: optimal dual solution $(\mathbf{f}^*, \mathbf{g}^*)$ of master problem, reduced cost function $r(C)$ defined in Eq.(3.9)
Output: a set $\mathcal{C}_{GMS}$ 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;
```

3.6 Experimental Evaluation

We demonstrate the effectiveness of our algorithmic framework through extensive numerical evaluations. We use 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 [77] 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 [78]. 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 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 (3.4) is represented by EXACT.

Our benchmark is a heuristic algorithm *Genetic Algorithm* (GA), which is a popular probabilistic search algorithm applied in complex optimization problem [79]. In our implementation of GA, each defender strategy is encoded into a binary string of length $|E|$, where 1 represents "blocking" and 0 represent "not blocking". The fitness of each solution is the defender utility $U_d(\mathbf{B})$. Each generation consists of a fixed number g of defender strategies. There are four genetic operators: selection, crossover, mutation and reproduction. During the selection processes, since the fitness is negative, we use

the tournament selection framework [80] instead of fitness proportionate selection. 2-point crossover is used to recombine the selected two defender strategies and produce the "offspring" strategies. A bitwise mutation procedure with mutation probability q is applied after the crossover is finished. The new generation is reproduced by copying g strategies with the highest utilities among parent and child generations. The selection-crossover-mutation-reproduction cycle is repeated until the terminal criterion is met: the gap of average fitness between parent generation and child generation is smaller than a fixed constant δ.

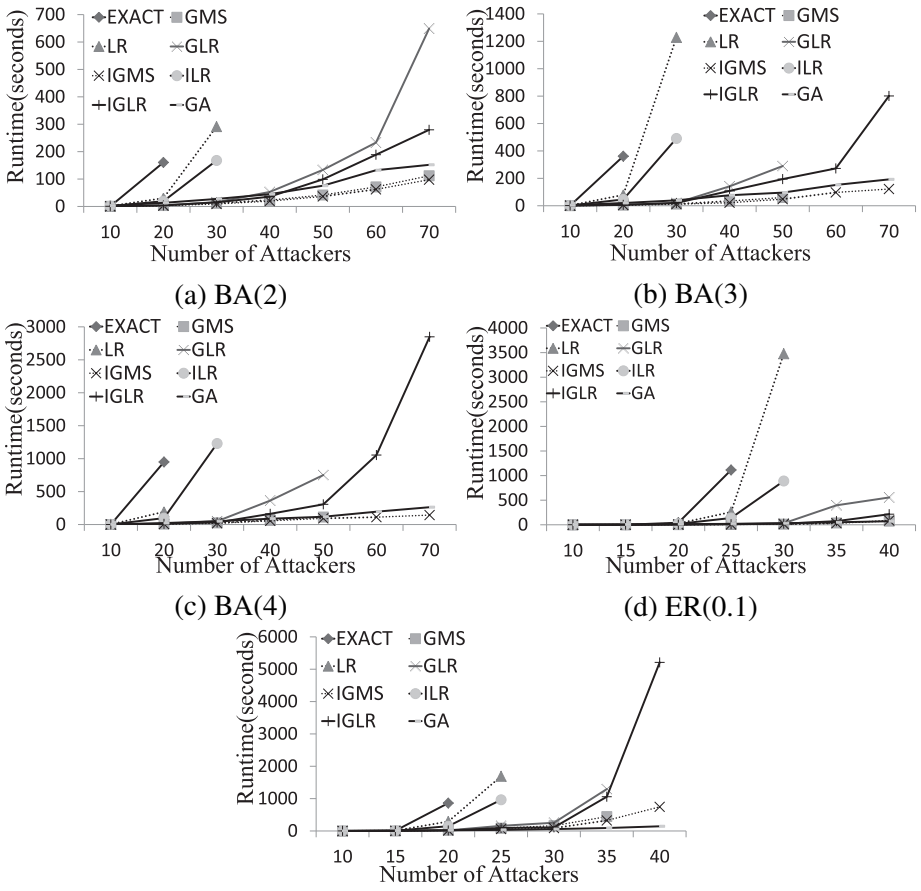
By default, the instances are parameterized as follows: 10 target types and 20 skills, the necessary skills for each target type and the ones owned by each attacker drawn randomly from S, and $m_{is} \sim \{1, 2, 3, 4, 5\}$; $p_t \sim [0, 1.0]$ and $\lambda_{ij} \sim [0, r]$ in which $r \in [0, 1]$ is chosen randomly. The parameters of GA are set as follows: the size of each generation is 100, and the mutation probability $q = 1/|E|$. Once the difference of fitness between the child generation and current generation is smaller than $\delta = 10^{-6}$, GA terminates.

3.6.1 Scalability and Optimality

We test different algorithms on 5 types of networks generated by BA(2), BA(3), BA(4), ER(0.1) and ER(0.2), and the results are shown in Figures 3.3–3.4.

3.6.1.1 Runtime

From Figure 3.3, we can observe the significant efficiency improvement provided by the branch and price framework. The interior point stabilization 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 3.1 contains 76 groups and the average degree is 3.4, which is well within the scalability of IGLR and IGMS.



(a) BA(2) (b) BA(3)

(c) BA(4) (d) ER(0.1)

(e) ER(0.2)

FIGURE 3.3: Runtime of algorithms for solving CSGs.

3.6.1.2 Defender Utility

We compare the defender utility of IGLR and IGMS, with EXACT and GA as benchmarks. Figure 3.4 shows that IGLR can achieve an almost optimal solution which outperforms GA significantly, in accordance with the theoretical analysis of Theorem 3.6. 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.

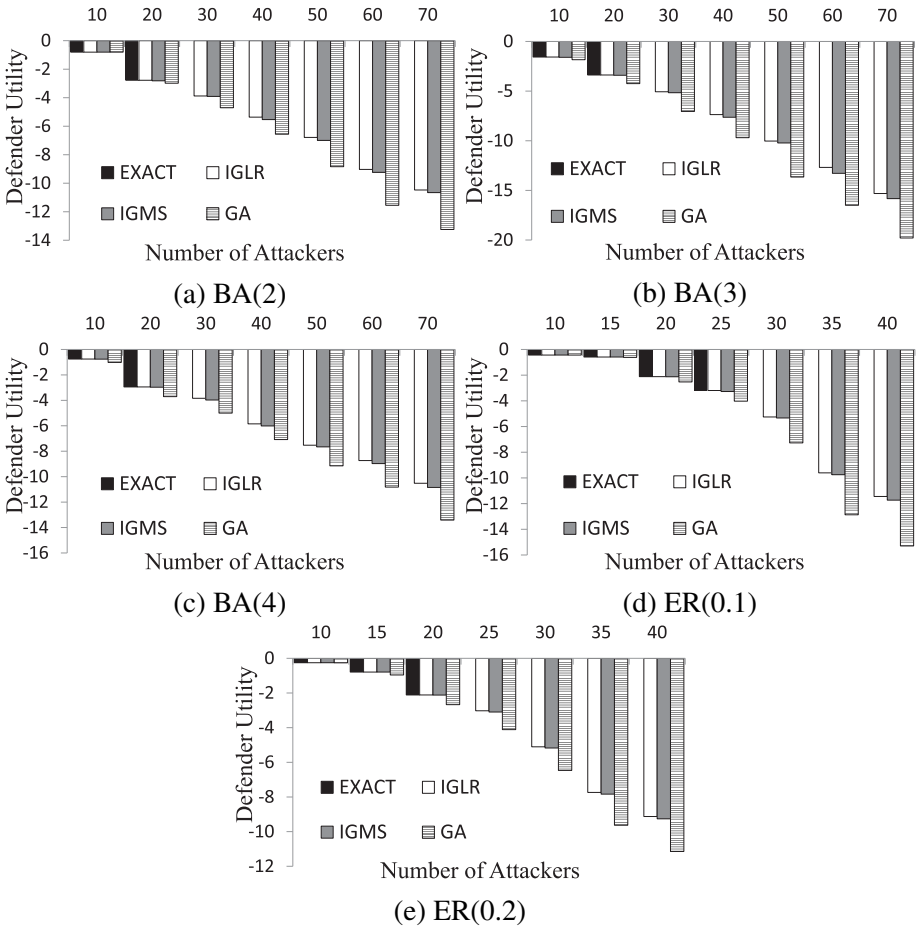
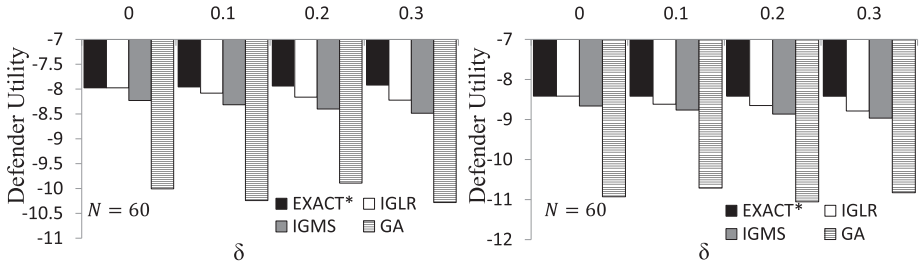


FIGURE 3.4: Solution qualities of algorithms for solving CSGs.

3.6.2 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 optimal solution, denoted by EXACT*[1], and heuristic solution of GA as comparison. The results are shown in Figure 3.5, from which a decreasing efficiency is observed for IGLR and IGMS when δ 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.

[1]Since IP (3.4) is not available for large-scale games, we use the solution of IGLR as an approximation.

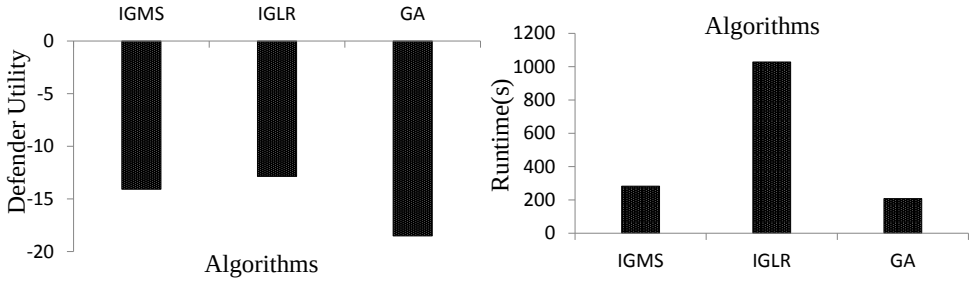


(a) BA(2) (b) BA(4)

FIGURE 3.5: Robustness analysis of algorithms for solving CSGs.

3.6.3 Realistic Terrorist Network

We conducted experimental evaluation of algorithms IGMS, IGLR, and GA on the Mideast terrorist network, as shown in Figure 3.1, with 76 groups and the average degree of 3.4, keeping other settings such as blocking costs, target values, skill sets the same as the previous trails, and the results averaged on 10 instances are depicted in Figure 3.6. As we can see from the figure, the solution qualities of our algorithms IGMS and IGLR outperform the heuristic method significantly, with acceptable runtime, which shows that our algorithmic framework captures the structure of realistic terrorist network as well.



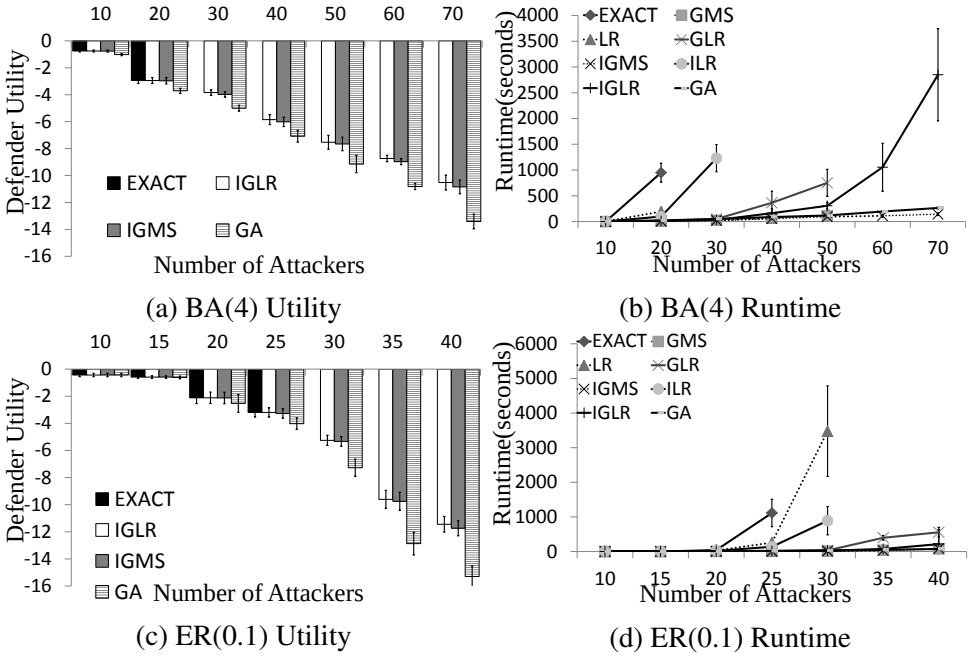
(a) Utility (b) Runtime

FIGURE 3.6: Simulation on Mideast terrorist network.

3.6.4 Statistic Analysis

We also conduct statistic analysis on instances where graphs are generated by BA(4) and ER(0.1), and the results are depicted in Figure 3.7 with error bars showing the 95% confidence interval. The figure shows that our results are statistically significant, and

the conclusion that our algorithmic framework outperforms the heuristic method in the solution qualities significantly with reasonable runtime is statistically convincing.



(a) BA(4) Utility (b) BA(4) Runtime

(c) ER(0.1) Utility (d) ER(0.1) Runtime

FIGURE 3.7: Statistic analysis on graphs generated by BA(4) and ER(0.1).

3.7 Chapter Summary

For the first time, this chapter studies the problem of blocking attacker coalition through efficient allocation of security resources. This chapter 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.

Chapter 4

Combating the Adversarial Flow on the Network

This chapter and the following one focus on stopping undesirable behavior abstracted by a flow on a network, which pertains to many security domains: disruption of enemy supply chains, infectious disease control, and the interception of illegal goods such as drugs or weapons. Physical networks are typically defended via *checkpoints*: physical roadblocks or inspection points positioned at various points in the network. In all network interdiction scenarios, the objective of those defending the network is to minimize the amount of flow through the network. From a resource optimization perspective, this is an extremely challenging task. There are several factors interrelating here: first, defender resources are limited; second, placing a guard at some point on the network does not guarantee that any smuggler passing through will be caught - it merely increases the likelihood of a successful capture; finally, it is natural to assume that smugglers have prior knowledge of defenders' positioning via intelligence gathering. The defender is thus placed at a disadvantage: not only does she have limited capability to stop an attack, her agents' moves are also monitored; thus, smugglers may successfully evade checkpoints if these are constantly positioned in a predictable manner. Indeed, randomized allocation strategies are imperatively needed [46]. This chapter and chapter 5 address the following challenge:

> *Devise a formal methodology for guarding a large, complex network against undesirable flow, considering action observability and limited resources.*

This chapter tackles such challenge from the one-shot game perspective and computes the minimax solution (Sec. 2.1.2), which is robust and widely adopted in game theory. The following chapter focuses on the repeated interactions between the adversary and the defender, which exist in many network-structured security domains, and proposes an efficient online policy to exploit the history information of the adversarial behavior to solve the more general network interdiction problem.

In this chapter, we introduce a novel Network Flow Interdiction Game (*NFIG*) model, where the defender allocates a fixed number of security resources on the network, while the adversary commits to a feasible network flow. To compute the optimal defender strategy, we first provide a standard minimax bilevel formulation and reformulate it as a linear program (*LP*). However, due to the huge number of constraints and variables caused by exponentially large numbers of defender strategies and network paths, the *LP* is hard to solve. To overcome the computational challenge, we propose a *Column and Constraint Generation (CCG)* algorithm, with the following key contributions: i) we show that the algorithm converges to the Stackelberg equilibrium with finite iterations; ii) we show the NP-hardness of the *Column Generation* subproblem, and provide several novel algorithms to solve it, including an exact compact convex nonlinear program and a greedy algorithm with constant-factor approximation guarantee; iii) we provide an exact compact convex nonlinear program for the *Constraint Generation* subproblem as well as a fast local search based heuristic algorithm; iv) extensive experimental evaluation shows that our *CCG* framework can scale up to realistic-sized *NFIG* instances and significantly outperform existing approaches.

The *network interdiction problem* is well studied [47]. The goal of the network interdiction is minimizing the flow through the graph via deletion of edges/nodes, maximizing the shortest path, or increasing detection probability through decreasing edge capacity [24–28]. Other works study *stochastic network interdiction* where the interdiction action is successful with some known probability p [63, 64]. However, none of them

consider the randomized resource allocation strategy, i.e., the mixed strategy, due to the impossibility to change the resource allocation frequently, especially for interdiction actions such as destroying arcs. Moreover, randomized resource allocation is known to be necessary to improve the interdiction efficiency; for example, in the smuggling domain, checkpoint operation status can actually change daily [46]. However, randomization does lead to a drastic increase in problem complexity: while in the deterministic network interdiction model an adversary's strategy has a compact representation (as a network flow), it is not the case in our setting.

We model NFIG as a *Stackelberg security game*; security games try and find the optimal resource allocation to defend some valuable facility, based on game-theoretic methodologies [4, 81]. However, in standard security game models, attacker strategies are typically compactly representable (e.g. a distribution over possible targets) [61, 82, 83]. Some recent work proposes network security games [29, 31, 48]; here, the defender allocates security resources on a (transportation) network and the attacker chooses a reliable path to escape, and both players have exponentially large strategy spaces; our model differs from these works in several important ways. Most importantly, our goal is minimizing the flow through a network, rather than preventing a single attacker from escaping; thus, the methodology proposed in existing works cannot be directly applied to our setting.

The rest of the Chapter is organized as follows: Section 4.1 presents a motivating scenario of the network flow interdiction problem. Section 4.2 illustrates our network flow interdiction game model. The algorithmic framework *Column and Constraint Generation (CCG)* is elaborated in Section 4.3. Section 4.4 investigates the column generation subproblem of CCG. Section 4.5 overviews the constraint generation subproblem of CCG. Section 4.6 shows the results on the experimental evaluation. The Chapter is wrapped up at Section 4.7.

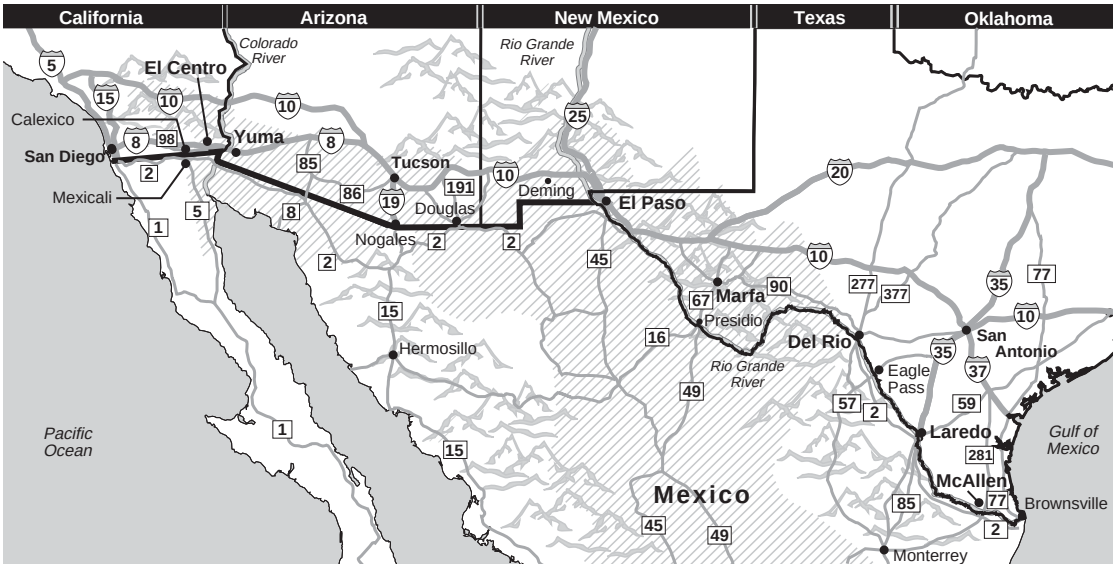


FIGURE 4.1: Topography and road systems in the southern border of U.S..

4.1 Drug Smuggling in Mexico–United States border

Our model is quite general, and can easily apply to various network security domains; to illustrate the underlying issues and motivation for our model choice, we now briefly present the problem in the context of stemming the drug flow through the US-Mexico border patrol. Figure 4.1 shows the major roads in the southern border area [46]. Drug smuggling in the southern border is dominated by four large well-organized and independent cartels: Sinaloa, Juarez, Gulf and Los Zetas, each of them controls the drug trafficking over 1000-mile border and has its own *major area of influence* in the United States [84, 85]. These organizations typically use commercial, private and rental vehicles to smuggle illegal goods via *land points of entry (sources)*. The goods travel along different paths on the road network to different *destination cities (sinks)* [85, 86]. Traffickers typically choose multiple transport paths, in order to avoid raising obvious suspicion by the inspection stations, and to mitigate losses in case of capture [87].

In 2009, the Border Patrol operated 39 *tactical checkpoints* with flexible and daily changeable operation status in the southern border, which are generally operated at fixed locations. However, due to the shortage of staff, canines and basic facilities (i.e. *security resources*), only few of them are actually operational [46]. With this limitation for tactical checkpoints, *randomized allocation strategies* of limited security resources are essential; smugglers have the ability to observe the operational status of checkpoints,

and make their best decision on trafficking activities, such as the amount of drugs to move through different POEs and paths. Indeed, a testimony before Congress by the Arizona Attorney General, revealed the sophisticated surveillance and communication technologies used to monitor security vulnerabilities and adjust plans to increase the chance of success [88].

4.2 Network Flow Interdiction Games

Our proposed *Network Flow Interdiction Game (NFIG)* models an attacker and a defender who take actions on a capacitated graph $G = (V, E)$, with nodes set V and edges set E, and a capacity vector $\mathbf{c}$, where capacity c_e represents the maximum amount of adversary flow passing through edge e without arousing obvious suspicion by inspection facilities (permanent checkpoints, sensors, etc). We assume that the graph has a unique *source node* (POE) $s \in V$ and a unique *sink node* $t \in V$ (drug distribution city in smuggling scenario). The unique source/sink assumption is no loss of generality: a graph with multiple source nodes and sink nodes can be transformed into a single-source-sink graph by adding two new nodes s and t as the new unique source and sink nodes respectively, connecting s to each source node and t to each sink node with proper capacitated edges. Let $\mathcal{P}$ denote the set of all s-t paths in G. For a path $p \in \mathcal{P}$, we say node $v \in p$ (edge $e \in p$) if p passes through v (e). Let I denote the set of all *inspection stations*, such as the tactical checkpoints in the border patrol scenario, and an inspection station is operated if the defender assigns a security resource on it, such as staffs, canines, and inspection facilities. Each station $i \in I$ is characterized by a location, either a node or an edge in the graph (with a bit abuse of notation, we use i interchangeable with its location, such that $i \in V \cup E$), and a constant parameter $\tau_i \in [0, 1]$ denoting the proportion of adversary flow interdicted at i when operated, i.e., *inspection probability*. For example, in the border patrol scenario, the officers at checkpoint i can conduct regular inspections on passing vehicles and an illegal trafficker will be caught with some probability τ_i depending on the officers' experience, which can be treated as a constant factor. Let $k < |I|$ be the number of security resources owned by the defender.

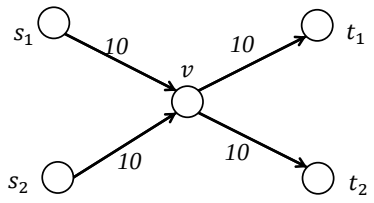
4.2.1 Strategies

A defender pure strategy $S = \langle S_i \rangle$ is an allocation of k security resources to k inspection stations, i.e., $\sum_{i \in I} S_i = k$, where $S_i \in \{0, 1\}$ and $S_i = 1$ indicates that the inspection station i is operated. The defender pure strategy space is denoted by $\mathcal{S}$. A mixed defender strategy $\mathbf{x} = \langle x_S \rangle$ is a probability distribution over all pure strategies where x_S denotes the probability that S is played. Let $\mathcal{X}$ denote the defender mixed strategy space.

An attacker strategy is a network flow $\mathbf{f}$ with f_p representing the amount of adversary flow passing along path $p \in \mathcal{P}$. In order to avoid raising obvious suspicion, a feasible attacker strategy must satisfy the capacity constraint on each edge. Let $\mathcal{F}$ denote the set of all feasible attacker strategies, i.e.,

$$\mathcal{F} = \{\mathbf{f} \geq \mathbf{0} : \sum\nolimits_{p \in \mathcal{P}: e \in p} f_p \leq c_e \quad \forall e \in E\}. \tag{4.1}$$

Note that the commonly adopted compact representation of network flow in an edge-based manner, i.e., $\mathbf{f} = \langle f_e \rangle$ is infeasible in *NFIG*. A simple example demonstrating the infeasibility of compact representation for the attacker's flow is shown in the figure on the left with 2 sources and two sinks. With compact representation, the flow $\mathbf{f}$ is defined over edges E such that f_e denotes the amount of flow passing through edge e. In this example, $f_e = 10$ for all edges.



Now, suppose the defender allocates two resources on edges (s_1, v) and (v, t_1) with inspection probabilities 1. In this case, we cannot compute the exact amount the flow interdicted by the defender. It can be 10 units passing through $s_1 - v - t_2$ and 10 units passing through $s_2 - v - t_1$, and the attacker utility is 0 in this case; It may also be 10 units passing through $s_1 - v - t_1$ and 10 units passing through $s_2 - v - t_2$, and the attacker utility is 10 now. The reason for this kind of infeasibility comes from the fact that each compact representation can be realized by numerous possible flows defined over paths, and they can have different utilities

against a given resource allocation. Thus, we adopt the path-based representation of the network flow in *NFIG*.

4.2.2 Utilities

Since the attacker aims at maximizing his successful drug flow while the defender wants to minimize it, we assume *NFIG* a zero-sum game. Given a pure defender strategy S and an attacker feasible flow $\mathbf{f}$, the attacker's utility is the sum of successful flows on all paths, i.e., $U_a(S, \mathbf{f}) = \sum_{p \in \mathcal{P}} \Phi(S, p) f_p$, where

$$\Phi(S, p) = \prod\nolimits_{i \in I : i \in p} (1 - \tau_i)^{S_i} \tag{4.2}$$

and $\Phi(S, p)$ represents the proportion of adversary flow on path p not interdicted by the operated inspection stations given S. The defender utility is $U_d(S, \mathbf{f}) = -U_a(S, \mathbf{f})$.

Given a defender mixed strategy $\mathbf{x}$ and an attacker feasible flow $\mathbf{f}$, the attacker's and defender's expected utilities are: $U_a(\mathbf{x}, \mathbf{f}) = \sum_{S \in \mathcal{S}} x_S U_a(S, \mathbf{f})$ and $U_d(\mathbf{x}, \mathbf{f}) = -U_a(\mathbf{x}, \mathbf{f})$.

4.2.3 Equilibrium

NFIG is a leader-follower game, for which the widely adopted solution concept is the Stackelberg equilibrium (*SE*). Let $y(\mathbf{x})$ denote the best response flow against defender mixed strategy $\mathbf{x}$. A strategy profile $\langle \mathbf{x}^*, \mathbf{f}^* \rangle$ forms an *SE*, if: i) $\mathbf{f}^* = y(\mathbf{x}^*)$, and ii) $U_d(\mathbf{x}^*, \mathbf{f}^*) \geq U_d(\mathbf{x}, y(\mathbf{x}))$ for all $\mathbf{x} \in \mathcal{X}$. With the zero-sum assumption, the *SE* can be obtained by the following minimax formulation:

$$\min_{\mathbf{x} \in \mathcal{X}} \max_{\mathbf{f} \in \mathcal{F}} U_a(\mathbf{x}, \mathbf{f}). \tag{4.3}$$

Since $\mathcal{X}, \mathcal{F}$ are convex sets, and $U_a(\mathbf{x}, \mathbf{f})$ is a linear function in $\mathbf{x}$ when fixing $\mathbf{f}$ and vice versa, according to the weak version of Sion's minimax theorem [57] (Sec. 2.1.2.2), we have:

$$\min_{\mathbf{x} \in \mathcal{X}} \max_{\mathbf{f} \in \mathcal{F}} U_a(\mathbf{x}, \mathbf{f}) = \max_{\mathbf{f} \in \mathcal{F}} \min_{\mathbf{x} \in \mathcal{X}} U_a(\mathbf{x}, \mathbf{f}) \tag{4.4}$$

A direct consequence of the minmax theorem, critical for our key solution approach (Section), is the equivalence of *SE* and the Nash equilibrium (*NE*), where $\mathbf{x}^*$ is the best response against $\mathbf{f}^*$ and $\mathbf{f}^*$ is also the best response against $\mathbf{x}^*$.

4.3 Algorithms

In this section, we first provide an LP formulation to compute the equilibrium based on the standard minimax formulation. Since this LP has exponentially large number of variables and constraints, we then propose a novel Column and Constraint Generation (*CCG*) to further improve the scalability.

4.3.1 LP Formulation

We start from the standard minimax formulation for *SE*:

$$\min_{\mathbf{x}} \max_{\mathbf{f}} \quad \sum\nolimits_{S\in\mathcal{S},p\in\mathcal{P}} \Phi(S,p) x_S f_p \tag{4.5a}$$

$$\text{s.t.} \quad \sum\nolimits_{S\in\mathcal{S}} x_S = 1 \tag{4.5b}$$

$$\sum\nolimits_{p\in\mathcal{P}:e\in p} f_p \le c_e \quad \forall e \in E \tag{4.5c}$$

$$\mathbf{x} \ge \mathbf{0}, \mathbf{f} \ge \mathbf{0}. \tag{4.5d}$$

The objective in Eq.(4.5a) is the attacker's expected utility $U_a(\mathbf{x}, \mathbf{f})$. The bilevel minimax program (4.5) can be reformulated as a linear program by replacing the inner program with its dual, and the resulting linear program (*LP*) is as follows:

$$\min_{\mathbf{x},\mathbf{u}} \quad \sum\nolimits_{e\in E} c_e u_e \tag{4.6a}$$

$$\text{s.t.} \quad \sum\nolimits_{S\in\mathcal{S}} x_S = 1 \tag{4.6b}$$

$$\sum\nolimits_{S\in\mathcal{S}} x_S \Phi(S,p) \le \sum\nolimits_{e\in p} u_e \quad \forall p \in \mathcal{P} \tag{4.6c}$$

$$\mathbf{x} \ge \mathbf{0}, \mathbf{u} \ge \mathbf{0}. \tag{4.6d}$$

The dual solution with respect to inequality (4.6c) provides the equilibrium attacker flow $\mathbf{f}$, defined over all paths in $\mathcal{P}$.

4.3.2 Column and Constraint Generation

The linear program (4.6) is challenging to solve due to the exponentially large number of variables and constraints ($|\mathcal{S}|, |\mathcal{P}|$). To address this challenge, we propose a novel algorithm called *Column and Constraint Generation (CCG)*, and prove the correctness of the *CCG* algorithm based on the property that the *SE* of NFIG is the same as the *NE* (Section 4.2.3).

The algorithm *CCG* is sketched in Algorithm 6. Initially, a small space $\langle \mathcal{S}', \mathcal{P}' \rangle$ is generated with arbitrary candidate defender strategies and s-t paths. Then *CCG* solves a restricted version of *NFIG* with $LP(\mathcal{S}', \mathcal{P}')$, where the defender pure strategy space is $\mathcal{S}'$ and the attacker flow is restricted to paths only in $\mathcal{P}'$, i.e., Eqs.(4.6a)–(4.6d) with $\langle \mathcal{S}, \mathcal{P} \rangle$ replaced by $\langle \mathcal{S}', \mathcal{P}' \rangle$, and obtains primal and dual solutions $(\mathbf{x}, \mathbf{u}, \mathbf{f})$. This restricted *NFIG* can be solved efficiently since the game is very small. Obviously, the obtained primal solution $\mathbf{x}$ and dual solution $\mathbf{f}$ form an *SE* of the restricted *NFIG*, but not necessarily the original *NFIG*. Therefore, two subproblems *ColG* (Column Generation) and *ConG* (Constraint Generation) are proposed to generate useful defender pure strategies and s-t paths into $\langle \mathcal{S}', \mathcal{P}' \rangle$ in order to guide the *SE* of the restricted *NFIG* to the one of the original *NFIG*. Respectively, *ColG* generates a defender strategy $S \in \mathcal{S}$ which is a best response against $\mathbf{f}$, and *ConG* generates an s-t path $p \in \mathcal{P}$ with maximal *reduced cost* which measures the degree of violating inequality (4.6c).

The key of the *CCG* algorithm is that once *ColG* generates the pure strategy S already in $\mathcal{S}'$, the current solution $\mathbf{x}$ of restricted *NFIG* is the defender best response against the adversary flow $\mathbf{f}$ in the original *NFIG* (PROPOSITION 4.1), and similarly, if *ConG* generates an s-t path p with non-positive reduced cost, the current equilibrium flow $\mathbf{f}$ of restricted *NFIG* is the attacker best response flow against $\mathbf{x}$ in the original *NFIG* (PROPOSITION 4.7). Therefore, the *NE* of the original *NFIG* is achieved when *ColG* generates a pure strategy S already in $\mathcal{S}'$ and *ConG* generates an s-t path p with non-positive reduced cost. Since the *NE* is the same as the *SE*, the current solution pair $(\mathbf{x}, \mathbf{f})$ forms the *SE* of the original *NFIG*.

Algorithm 6: CCG Algorithm for NFIG

Initialize $\mathcal{S}'$ by generating arbitrary defender strategies;
Initialize $\mathcal{P}'$ by generating arbitrary s-t paths;
repeat
  $(\mathbf{x}, \mathbf{u}, \mathbf{f}) \leftarrow LP(\mathcal{S}', \mathcal{P}')$;
  $S \leftarrow ColG(\mathbf{f})$;
  $\mathcal{S}' \leftarrow \mathcal{S}' \cup \{S\}$;
  $p \leftarrow ConG(\mathbf{x}, \mathbf{u})$;
  $\mathcal{P}' \leftarrow \mathcal{P}' \cup \{p\}$;
until convergence;
return $(\mathbf{x}, \mathbf{f})$.

4.3.3 Comparison with Double Oracle

The *double oracle (DO)* algorithm (Sec. 2.1.2.1), first proposed by McMahan *et al.* [56], is a standard method for solving two-player zero-sum games with large scale strategy spaces, and is widely adopted by security game research [29, 30, 48, 89]. In *DO*, both players commit to mixed strategies. However, in NFIG, the attacker is not playing a mixed strategy. That is, the attacker strategy space is no longer a simplex where each decision point corresponds to one probability distribution. Instead, an attacker strategy is a capacity-constrained network flow and thus, the attacker strategy space is a convex hull represented by the flow constraints. The convergence to the minimax solution of *DO* algorithm is proved by the von Neumann's minimax theorem working on the simplex strategy spaces (Sec. 2.1.2.2). On the other hand, the convergence result of *CCG* is guaranteed by the generalized version of minimax theorem, in particular, weak version of Sion's minimax theorem (Sec. 2.1.2.2). In implementation, each oracle in *DO* generates one pure strategy best response for the corresponding player, which would be a network flow for the adversary considering the attacker oracle. In *CCG*, instead, the subproblem *ConG* solves an extreme point of the attacker strategy space, in this case, an s-t path.

The central challenge and novelty of *CCG* is how to efficiently generate the pure defender strategies and s-t paths to add to $\mathcal{S}'$ and $\mathcal{P}'$. Thus, we present the corresponding *ColG* and *ConG* algorithms in the following sections.

4.4 Column Generation (ColG)

This section concerns the *ColG* subproblem of Algorithm 6, which can be stated as follows: given an attacker feasible flow $\mathbf{f}$ defined over $\mathcal{P}'$, generate the defender pure strategy $S \in \mathcal{S}$ blocking the largest amount of flow, i.e., $S = \arg\max_{S' \in \mathcal{S}} U_d(S', \mathbf{f})$. PROPOSITION 4.1 shows that once ColG generates the pure strategy S already in $\mathcal{S}'$, the current equilibrium mixed strategy $\mathbf{x}$ of restricted *NFIG* is the best response against $\mathbf{f}$ of the original *NFIG*.

Proposition 4.1. *If* ColG *generates the pure strategy S already in $\mathcal{S}'$, the current equilibrium mixed strategy $\mathbf{x}$ of restricted* NFIG *is the best response against $\mathbf{f}$ of the original* NFIG.

Proof. To prove this Proposition, it is sufficient to show that $U_d(\mathbf{x}, \mathbf{f}) \geq U_d(\mathbf{x}', \mathbf{f})$ for all $\mathbf{x}' \in \mathcal{X}$. Since $\langle \mathbf{x}, \mathbf{f} \rangle$ forms the Nash (Stackelberg) equilibrium of the restricted *NFIG* with restricted sets $\mathcal{S}'$ and $\mathcal{P}'$, we have:

$$U_d(\mathbf{x}, \mathbf{f}) \geq U_d(S', \mathbf{f}) \quad \forall S' \in \mathcal{S}'$$

Therefore, if $S \in \mathcal{S}'$, we have: $U_d(\mathbf{x}, \mathbf{f}) \geq U_d(S, \mathbf{f})$. For any mixed strategy $\mathbf{x}' \in \mathcal{X}$, we have:

$$U_d(\mathbf{x}', \mathbf{f}) = \sum_{S' \in \mathcal{S}} x_{S'} U_d(S', \mathbf{f}) \leq U_d(S, \mathbf{f})$$

since S is the best response pure strategy against $\mathbf{f}$ and $\sum_{S' \in \mathcal{S}} x_{S'} = 1$.

Thus, $U_d(\mathbf{x}, \mathbf{f}) \geq U_d(\mathbf{x}', \mathbf{f})$ for all $\mathbf{x}' \in \mathcal{X}$. □

We conduct thorough analysis of *ColG*'s computational complexity, and show that it is NP-hard for general graphs, while polynomial-time solvable for the tree-like networks.

Theorem 4.2. *The* ColG *subproblem is NP-hard.*

Proof. We show the NP-hardness of the Column Generation problem by reducing the set cover problem to it. **The Set-Cover problem:** Given a set U, a collection $\mathcal{Q}$ of

subsets of U (that is, $\mathcal{Q} \subseteq 2^U$), and an integer k. The question is whether there is a cover $\mathcal{C} \subseteq \mathcal{Q}$ of size k or less, that is, $\bigcup_{q \in \mathcal{C}} q = U$ and $|\mathcal{C}| \leq k$.

Given an instance $\langle U, \mathcal{Q}, k \rangle$ of set cover problem, we reduce it to a *ColG* instance $\langle G, \mathcal{P}', I, \mathbf{f} \rangle$ as follows: Except the two unique source and sink nodes s, t, for each subset $q \in \mathcal{Q}$, we assign a vertex v_q in G and an inspection station i_q on vertex v_q with inspection probability $\tau_{i_q} = 1$. Now, for each element $u \in U$, we construct a path $p_u \in \mathcal{P}'$ as follows: p_u starts from s, passing through vertices v_q such that $u \in q$, and end at t; The attacker flow $\mathbf{f}$ is defined on paths of $\mathcal{P}'$ such that $f_p = 1$ for all $p \in \mathcal{P}'$. Obviously, this reduction can be done in polynomial time. Next, we show that the set cover problem instance $\langle U, \mathcal{Q}, k \rangle$ has a cover $\mathcal{C}$ of size k or less if and only if the Column Generation problem has a solution S such that all the flow in $\mathbf{f}$ is intervened, i.e., $U_a(S, \mathbf{f}) = 0$.

The "if" direction: If S intervenes all the flow in $\mathbf{f}$, which means each path p_u in $\mathcal{P}'$ passes through some inspection station i_q in S, then the collection $\mathcal{C} = \{q \in \mathcal{Q} : i_q \in S\}$ is a cover of size k or less.

The "only if" direction: If there exists a cover $\mathcal{C}$ of size k or less for the set cover problem, the defender pure strategy $S = \{v_q : q \in \mathcal{C}\}$ intervenes all the paths $p_u \in \mathcal{P}'$ and hence all the flow in $\mathbf{f}$ is blocked. □

Theorem 4.3. *If the network G is tree-like, i.e., $G - \{t\}$ is a tree, then there exists a dynamic programming algorithm able to solve the* ColG *subproblem in polynomial time.*

Proof. We provide a dynamic programming algorithm for solving the *ColG* subproblem on tree-like graph G with source node s in polynomial time. First, we transform G into a tree G' in following way: for sink node t, supposing its incoming edges are $(v_1, t), ..., (v_m, t)$, we replace t with m new sinks $t_1, ..., t_m$ and replace incoming edges with m new edges $(v_1, t_1), ..., (v_m, t_m)$. Figure 4.2 shows an example of the tree G' transformed from tree-like graph G. Let s be the root of tree G'. Given a vertex v, let T_v be the subtree rooted at v, C_v be the set of child nodes of v, c_v^i be the i-th child node of v, and T_v^j be the subtree rooted at v with first j branches.

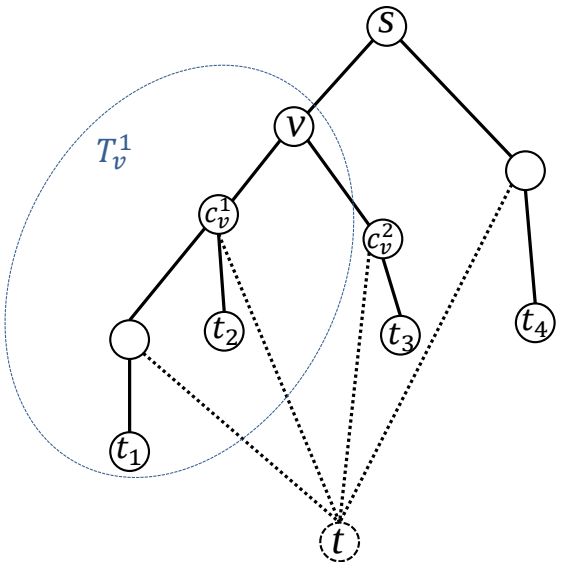


FIGURE 4.2: Dynamic programming for tree-like graph.

Let $S_v(r, j)$ denote the optimal allocation of r resources on T_v^j, and $S_v(r) = S_v(r, |C_v|)$ be the optimal allocation of r resources on T_v. A key property here is that since all flows passing through T_v^j come from the unique path from s to v (G' is a tree), the resource allocation outside T_v^j will decrease f_p with the same proportion for all paths p from s to $v \in T_v^j$. Thus, the optimal resource allocation $S_v(r, j)$ is independent with allocation of resources on the complementary part $G' - T_v^j$.

The above property leads to the following recursion:

$$S_v(r, j+1) = \arg\min_{r'<r} \min \begin{cases} U(S_v(r-r', z)) + U(S_{v'}(r')) \\ U(S_v(r-r'), z) + k_v^{v'} U(S_{v'}(r'-1)) \end{cases} \tag{4.7}$$

where $U(S_v(r, j))$ measures the amount of flow passing through T_v^j with resource allocation $S_v(r, j)$ on T_v^j and no resource put outside T_v^j, $v' = c_v^{j+1}$, i.e., the $j+1$-th child node of v, and $k_v^{v'}$ be the inspection probability of the checkpoint on edge (v, v'). The idea of this recursion is that allocating r resources on T_v^{j+1} can be done in two possible ways: i) putting $r - r'$ resources on T_v^j and the rest on $T_{v'}$; ii) putting $r - r'$ resources on T_v^j, $r' - 1$ resources on $T_{v'}$, and one resource on (v, v'). The best way minimizing the amount of flow passing through T_v^j gives $S_v(r, j+1)$. Here, we assume that checkpoints are all on edges, which is reasonable because in the tree graph, each node v has a unique parent node $\tilde{v}$. Thus, we can always insert a new node v' at the middle of edge

$(\tilde{v}, v)$ and move the checkpoint on v to the new edge (v', v), while keeping the game unchanged.

With the above recursion, we can compute $S_v(r, j)$ for every nodes following direction from sink nodes to s with r from 1 to the budget k and j from 1 to $|C_v|$. Thus, the algorithm computes the optimal solution $S^* = S_s(k)$ for *ColG* in $O(k|V|^2)$ runtime.

□

4.4.1 Convex Integer Nonlinear Formulation

Disregarding the NP-hardness of *ColG* subproblem, we provide a compact integer program, with nonlinear objective function $U_a(S, \mathbf{f})$, to solve it exactly. The program is proved to be convex (PROPOSITION 4.4) and hence can be solved to optimality with many commercial solvers, like KNITRO.

$$\min_{S} \quad \sum\nolimits_{p\in\mathcal{P}'} \prod\nolimits_{i\in I: i\in p} (1-\tau_i)^{S_i} f_p \tag{4.8a}$$

$$\text{s.t.} \quad \sum\nolimits_{i\in I} S_i \leq k \tag{4.8b}$$

$$S \in \{0,1\}^{|I|}. \tag{4.8c}$$

Proposition 4.4. *Given an attacker flow* $\mathbf{f} \in \mathcal{F}$*, the objective function* (4.8a) *is convex over decision variable* $S \in [0,1]^{|I|}$.

Proof. To prove the convexity of $U_a(S, \mathbf{f})$, it is enough to prove that $\Phi(S, p)$ is convex. To do so, we provide an equivalent representation of $\Phi(S, p)$ defined in the following equation:

$$\Phi(S, p) = \prod_{i\in I} (1-\tau_i)^{\alpha_{pi} S_i},$$

where $\alpha_{pi} = 1$ if path p passes through inspection station i and $\alpha_{pi} = 0$ otherwise. The Hessian matrix of $\Phi(S, p)$ is denoted by $\mathbf{H}$, such that:

$$H_{ij} = \frac{\partial^2 \Phi(S,p)}{\partial S_i \partial S_j} = \alpha_{pi}\alpha_{pj} ln(1-\tau_i) ln(1-\tau_j) \Phi(S,p)$$

Algorithm 7: Greedy Algorithm for Column Generation

Initialize $S' = \emptyset$;
while $|S'| < k$ **do**
  $i^* = \arg\max_{i \notin S': S' \cup \{i\} \in \mathcal{S}} U_d(S' \cup \{i\}, \mathbf{f})$;
  $S' = S' \cup \{i^*\}$;
return S'.

It is easy to verify that $\mathbf{H}$ can be represented as:

$$\mathbf{H} = \mathbf{y}\mathbf{y}^{\mathbf{T}}\Phi(S, p)$$

where $y_i = \alpha_{pi} ln(1 - \tau_i)$. Therefore, $\mathbf{H}$ is positive semidefinite, and $U_a(S, \mathbf{f})$ is convex function over $S \in [0, 1]^{|I|}$. □

4.4.2 Greedy Algorithm

We also propose a polynomial-time greedy algorithm of computing the approximate best response defender pure strategy, as shown in Algorithm 7 (we slightly abuse notation and let S represent the set of operated inspection stations). Starting from an empty set $S = \emptyset$, we iteratively assign a security resource on an inspection station i^* which brings the maximal marginal utility $U_d(S \cup \{i\}, \mathbf{f}) - U_d(S, \mathbf{f})$, until all k resources are assigned. The approximate pure strategy S' of Algorithm 7 and the approximate mixed strategy $\mathbf{x}'$ computed by *CCG* with *ColG* solved by Algorithm 7 are proved to achieve competitive ratios (THEOREMS 4.5 & 4.6).

Theorem 4.5. *The utility of S' obtained by Algorithm 7 is bounded by $U_d(S', \mathbf{f}) - U_d(\emptyset, \mathbf{f}) \geq (1 - \frac{1}{e})(U_d(S^*, \mathbf{f}) - U_d(\emptyset, \mathbf{f}))$, where S^* is the optimal solution for* ColG.

Proof. The defender utilities are always negative, therefore an approximation guarantee in terms of utility is meaningless. To facilitate analysis, we first define a non-negative function h for each defender allocation S such that $|S| \leq k$:

$$h_{\mathbf{f}}(S) = U_d(S, \mathbf{f}) - U_d(\emptyset, \mathbf{f}) = \sum_{p \in \mathcal{P}} (1 - \Phi(S, p)) f_p$$

Note h denotes the marginal benefit of allocation S over the non-defending defender strategy.

Given the adversarial flow $\mathbf{f}$, we first prove that $h_{\mathbf{f}}(S)$ is a monotone submodular function, i.e., i) $h_{\mathbf{f}}(S_1) \leq h_{\mathbf{f}}(S_2)$, and ii) $h_{\mathbf{f}}(S_1 \cup \{i\}) - h_{\mathbf{f}}(S_1) \geq h_{\mathbf{f}}(S_2 \cup \{i\}) - h_{\mathbf{f}}(S_2)$, for each $S_1 \subseteq S_2 \subseteq I, i \in I \setminus S_2$.

Monotone Remember $\Phi(S, p)$ represents the proportion of adversary flow on path p not interdicted by the operated inspection stations given S. Given $S_1 \subseteq S_2$, we have $\Phi(S_1, p) \geq \Phi(S_2, p)$ for all $p \in \mathcal{P}$. Thus, $h_{\mathbf{f}}(S_1) \leq h_{\mathbf{f}}(S_2)$.

Submodulariry For each $S \subseteq I$ and $i \in I \setminus S$, we have:

$$h_{\mathbf{f}}(S \cup \{i\}) - h_{\mathbf{f}}(S) = \sum_{p \in \mathcal{P}} (\Phi(S, p) - \Phi(S \cup \{i\}, p)) f_p$$

According to Eq.(4.2):

$$\Phi(S, p) = \prod_{i \in I : i \in p} (1 - \tau_i)^{S_i}$$

We have:

$$h_{\mathbf{f}}(S \cup \{i\}) - h_{\mathbf{f}}(S) = \sum_{p \in \mathcal{P} : i \in p} \tau_i \cdot \Phi(S, p) f_p$$

Since $\Phi(S_1, p) \geq \Phi(S_2, p)$ for each $S \subseteq I$ and $i \in I \setminus S$ and $p \in \mathcal{P}$, we get: $h_{\mathbf{f}}(S_1 \cup \{i\}) - h_{\mathbf{f}}(S_1) \geq h_{\mathbf{f}}(S_2 \cup \{i\}) - h_{\mathbf{f}}(S_2)$.

Therefore, $h_{\mathbf{f}}(S)$ is a non-negative monotone submodular function. According to [90], the greedy algorithm for submodular function maximization can achieve a $1 - \frac{1}{e}$ approximation ratio. Hence, $U_d(S', \mathbf{f}) - U_d(\emptyset, \mathbf{f}) \geq (1 - \frac{1}{e})(U_d(S^*, \mathbf{f}) - U_d(\emptyset, \mathbf{f}))$. □

Theorem 4.6. *Let* $(\mathbf{x}^*, \mathbf{f}^*)$ *be the equilibrium solution of an* NFIG*, and let* $(\mathbf{x}', \mathbf{f}')$ *be the solution computed by* CCG *algorithm with* ColG *subproblem solved by greedy Algorithm 7. Then* $U_d(\mathbf{x}', \mathbf{f}') - U_d(\emptyset, \mathbf{f}') \geq (1 - \frac{1}{e})(U_d(\mathbf{x}^*, \mathbf{f}^*) - U_d(\emptyset, \mathbf{f}'))$.

Proof. Let S' be the greedy solution of Algorithm 7 against attacker's network flow $\mathbf{f}'$, and S^* be the defender best response pure strategy against $\mathbf{f}'$, i.e.,

$$S^* = \arg\max_{S \in \mathcal{S}} U_d(S, \mathbf{f}') \tag{4.9}$$

According to Theorem 4.5, we have:

$$U_d(S', \mathbf{f}') - U_d(\emptyset, \mathbf{f}') \geq (1 - \frac{1}{e})(U_d(S^*, \mathbf{f}') - U_d(\emptyset, \mathbf{f}'))$$

Since $\mathbf{x}'$ is the defender best response strategy against $\mathbf{f}'$ in the restricted *NFIG* defined on restricted defender pure strategy space $\mathcal{S}'$, and $S' \in \mathcal{S}'$ according to the convergence criteria of *CCG* algorithm, we have:

$$\begin{aligned} & U_d(\mathbf{x}', \mathbf{f}') - U_d(\emptyset, \mathbf{f}') \\ & \geq U_d(S', \mathbf{f}') - U_d(\emptyset, \mathbf{f}') \\ & \geq (1 - \frac{1}{e})(U_d(S^*, \mathbf{f}') - U_d(\emptyset, \mathbf{f}')) \end{aligned} \tag{4.10}$$

According to Eq.(4.9), we have:

$$U_d(\mathbf{x}^*, \mathbf{f}') = \sum_{S \in \mathcal{S}} x_S^* U_d(S, \mathbf{f}') \leq U_d(S^*, \mathbf{f}')$$

Since $\mathbf{f}^*$ is the attacker best response flow against $\mathbf{x}^*$ in the original *NFIG*, we have:

$$U_d(\mathbf{x}^*, \mathbf{f}^*) \leq U_d(\mathbf{x}^*, \mathbf{f}') \leq U_d(S^*, \mathbf{f}') \tag{4.11}$$

Finally, substitute Eq.(4.10) with Eq.(4.11):

$$\begin{aligned} & U_d(\mathbf{x}', \mathbf{f}') - U_d(\emptyset, \mathbf{f}') \\ & \geq (1 - \frac{1}{e})(U_d(\mathbf{x}^*, \mathbf{f}^*) - U_d(\emptyset, \mathbf{f}')) \end{aligned}$$

□

4.5 Constraint Generation (ConG)

Given the optimal primal solution $(\mathbf{x}, \mathbf{u})$ of $LP(\mathcal{S}', \mathcal{P}')$, the *ConG* subproblem generates an s-t path $p \in \mathcal{P}$ that has maximal reduced cost defined as: $R(p) = \sum_{S \in \mathcal{S}'} x_S \Phi(S, p) - \sum_{e \in p} u_e$, i.e., $p = \max_{p' \in \mathcal{P}} R(p')$. Note that $R(p) \leq 0$ if and only if inequality (4.6c) associated with p is satisfied by $(\mathbf{x}, \mathbf{u})$. We show that if *ConG* generates an s-t path p such that $R(p) \leq 0$, the current equilibrium flow $\mathbf{f}$ of the restricted *NFIG* is the best response against $\mathbf{x}$ of the original *NFIG*.

Proposition 4.7. *If the* ConG *generates an s-t path p with non-positive maximal reduced cost, the current equilibrium flow $\mathbf{f}$ of the restricted* NFIG *is the attacker best response pure strategy against $\mathbf{x}$ of the original* NFIG.

Proof. Supposing the optimal solution of $CoreLP(\mathcal{S}', \mathcal{P}')$ is $(\mathbf{x}, \mathbf{u})$, then the current equilibrium flow $\mathbf{f}$ over $\mathcal{P}'$, which is the dual solution with respect to inequality (4.6c) on paths in $\mathcal{P}'$, satisfies

$$U_a(\mathbf{x}, \mathbf{f}) = \sum_{e \in E} c_e u_e$$

based on the dual theory [74]. If the ConG generates an s-t path p with maximal reduced cost such that $R(p) \leq 0$, we have

$$R(p') \leq R(p) \leq 0 \quad \forall p' \in \mathcal{P}$$

In other words, $(\mathbf{x}, \mathbf{u})$ is also the optimal solution of $CoreLP(\mathcal{S}', \mathcal{P})$ as it satisfies inequality (4.6c) with respect to all $p \in \mathcal{P}$. Notice that $CoreLP(\mathcal{S}', \mathcal{P})$ computes the attacker best response flow $\mathbf{f}^*$ against defender mixed strategy $\mathbf{x}$ over pure strategy set $\mathcal{S}'$ also satisfying

$$U_a(\mathbf{x}, \mathbf{f}^*) = \sum_{e \in E} c_e u_e$$

according to the dual theory. Thus, $U_a(\mathbf{x}, \mathbf{f}) = U_a(\mathbf{x}, \mathbf{f}^*)$, and the current equilibrium flow $\mathbf{f}$ is an attacker best response pure strategy against $\mathbf{x}$. □

4.5.1 Convex Integer Nonlinear Formulation

We first propose a compact convex integer program to solve the *ConG* problem exactly. Let binary variable $\boldsymbol{\gamma} \in \{0,1\}^{|V|+|E|}$ denote an s-t path p, such that for every node $v \in V$ and edge $e \in E$, $\gamma_v = 1$ if v is on path p and $\gamma_e = 1$ if e is on path p. We say node $v \in e$ if edge e is adjacent with v. The convex formulation is as follows:

$$\max_{\boldsymbol{\gamma}} \quad \sum\nolimits_{S\in\mathcal{S}'} x_S \prod\nolimits_{i\in S}(1-\tau_i)^{\gamma_i} - \sum\nolimits_{e\in E} u_e\gamma_e \tag{4.12a}$$

$$\text{s.t.} \quad \sum\nolimits_{e:s\in e} \gamma_e = 1, \sum\nolimits_{e:t\in e} \gamma_e = 1 \tag{4.12b}$$

$$\sum\nolimits_{e:v\in e} \gamma_e = 2\gamma_v \quad \forall v \in V, v \neq s,t \tag{4.12c}$$

$$\gamma_e \leq \gamma_v, \gamma_e \leq \gamma_{v'} \quad \forall e \in E : e = (v, v') \tag{4.12d}$$

$$\gamma_s = \gamma_t = 1 \tag{4.12e}$$

$$\boldsymbol{\gamma} \in \{0,1\}^{|V|+|E|}. \tag{4.12f}$$

The convexity of the objective (4.12a) can be easily verified in the same way of proving PROPOSITION 4.4. Eqs.(4.12b)–(4.12e) ensure that $\boldsymbol{\gamma}$ denotes a feasible s-t path p in the sense that: i) s and t are on p (4.12e); ii) exactly one edge adjacent with s (t) is on p (4.12b); iii) if node v, except s and t, is on p, then exactly two edges adjacent with v are on p (4.12c); and iv) if node v is not on p, then no edge adjacent with v is on p (4.12c).

Notice that the program (4.12) without constraint (4.12d) can also solve the *ConG* subproblem exactly. The integer programs are usually solved with the popular *branch and bound (B&B)* framework and the efficiency of *B&B* depends on the quality of the upper bound obtained by solving the linear relaxation of the program (4.12). Thus, inequality (4.12d) is proposed to tighten the upper bound by cutting off some fractional solutions with too high objective values, such an example is $\boldsymbol{\gamma}$ with: i) $\gamma_e = 1$ for $e' = (s, v')$, $e'' = (v'', t)$; ii) $\gamma_v = 0.5$ for v' and v''; iii) $\gamma_v = 0$, $\gamma_e = 0$ for all other nodes and edges.

Algorithm 8: *VNS* for *ConG*

1 Generate a random $s-t$ path p;
2 $p \leftarrow LocalSearch(p)$;
3 **while** *no termination condition is met* **do**
4 $\quad p' \leftarrow$ randomly pick one from $\mathcal{N}^l(p)$;
5 $\quad p' \leftarrow LocalSearch(p')$;
6 $\quad$ **if** $R(p') > R(p)$ **then** $p \leftarrow p'$;
7 **if** $R(p) > 0$ **then return** p

Algorithm 9: *LocalSearch* (p)

1 **Repeat**
2 $\quad p^* \leftarrow \arg\max_{p' \in \mathcal{N}^l(p)} R(p')$;
3 $\quad$ **if** $R(p^*) > R(p)$ **then** $p \leftarrow p^*$;
4 $\quad$ **else return** p;

4.5.2 Variable Neighborhood Search (VNS)

Although the convex program (4.12) can solve the *ConG* subproblem exactly, it still cannot scale up to solve large scale networks. Observe, however, in each iteration of *CCG* algorithm, we actually need not find the path with maximal reduced cost; rather, any path with a large enough (positive) reduced cost would suffice. Thus, a fast heuristic search for generating such paths is satisfactory for most iterations, except when the heuristic is unable to find a path with positive reduced cost. In this case we can call the convex program (4.12). To this end, we propose a heuristic search algorithm (Algorithm 8) based on *Variable Neighborhood Search (VNS)* framework [91].

Two key components of *VNS* are: i) $\mathcal{N}^l(p)$, denoting the set of neighbor paths of p within distance l; and ii) *LocalSearch* (p) to find a local optimum starting from p (Algorithm 9). We say that path p' is a neighbor path within distance l of p if p' can be obtained by replacing a v'-v'' sub-path of p with another v'-v'' path of length no larger than l. The *LocalSearch* (p) iteratively searches for a neighbor path p^* with maximal reduced cost (Line 2) and updates current incumbent p if p^* is a better solution (Line 3), until a local optimum is obtained whose reduced cost is higher than all its neighbor paths.

After initialized with an arbitrary local optimum (Lines 1–2), the *VNS* randomly picks a neighbor path p' of p and applies *LocalSearch* (p') to find a local optimum

(Lines 4–5). Afterwards, the incumbent p is updated. This loop is repeated until a termination condition is met: i) current incumbent p has a positive reduced cost (Line 7); or ii) for c_{max} consecutive iterations, the incumbent p is not updated.

4.6 Experimental Evaluation

We evaluate the performance of our approach through extensive experiments. We use CPLEX (version 12.6) to solve linear programs and KNITRO (version 9.0.0) to solve nonlinear 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 random planar graphs are generated by the Waxman geographical model (*WG*) suitable for modeling highway networks [92]. In the *WG* model, $|N|$ nodes (cities) are placed in a rectangular domain uniformly, and each pair of nodes at Euclidean distance d is jointed by an edge with probability $p = e^{-\lambda d}$, where λ is a constant adjusted to achieve the desirable average degree D. The two nodes with maximal Euclidean distance are set as the *source* and *sink* nodes of graph G. By default, the instances are parameterized as follows: the number of inspection stations $|I| = \lfloor \alpha(|V| + |E|) \rfloor$ and the number of resources $k = \lfloor \beta(|V| + |E|) \rfloor$, where α and β are tunable parameters. All inspection stations are randomly placed on the graph. The edge capacity c_e is randomly chosen in $[0, 10]$, and inspection probability $\tau_i \sim [0.4, 0.6]$. The l and c_{max} are set as 4 and $|V|$ respectively in *VNS*.

We compare the scalability of three versions of our algorithms: i) *CCG*: Algorithm 6 with *ColG* and *ConG* solved by programs (4.8) and (4.12); ii) *CCG**: Algorithm 6, where *ColG* and *ConG* are first solved by greedy methods (Algorithms 7 and 8), and then call the programs (4.8) and (4.12) when greedy methods return a pure strategy S already in $\mathcal{S}'$ or path p with $R(p) \leq 0$; iii) *LP*: the linear program (4.6), with the benchmark *double oracle (DO)* where two oracles solve the best response pure strategies (S, $\mathbf{f}$) for both players [29].

To evaluate the solution quality of our approach, we implement three heuristic strategies as baselines. The quality of a solution $\mathbf{x}$ is measured by $U_d(\mathbf{x}, y(\mathbf{x}))$. The baseline

mixed strategies are: i) *DMU* with marginal coverage probability of each inspection station equal to $\frac{k}{|I|}$; ii) *DMT* where marginal coverage probability of each inspection station is normalized inspection probability; iii) *DMW* where marginal coverage probability of each inspection station is normalized inspection weight. The inspection weight w_i of station i is defined as the maximum amount of flow that i can interdict, i.e., the production of τ_i and the maximum amount of flow passing through that edge or node; Given the marginal coverage probabilities, the defender mixed strategies are generated by the *Combo Sampling* algorithm [31].

4.6.1 Scalability Analysis

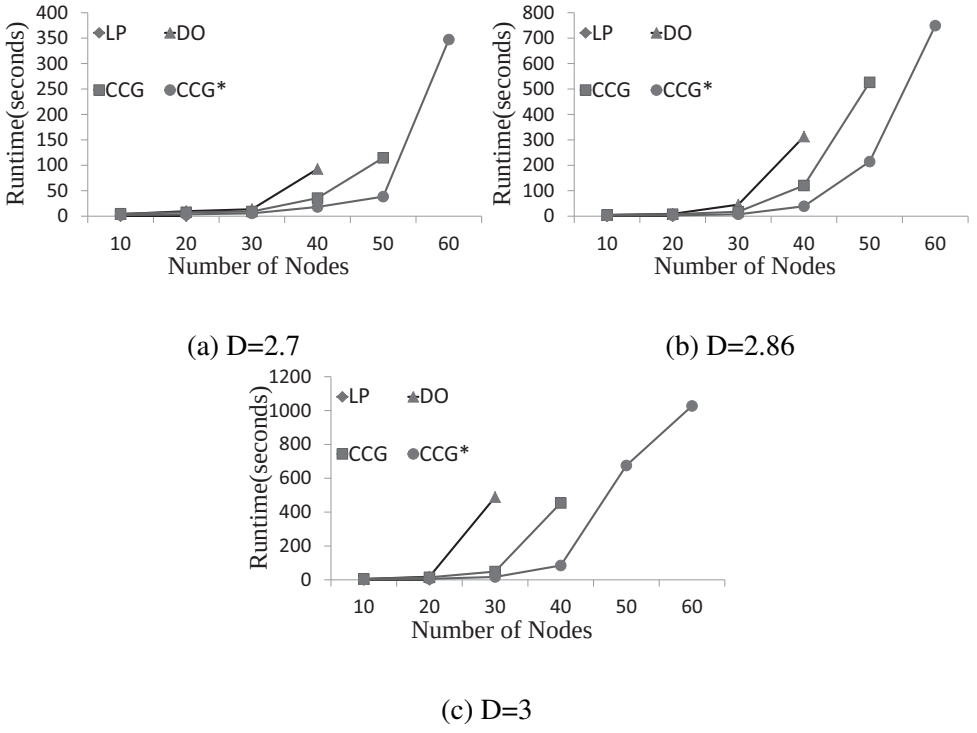
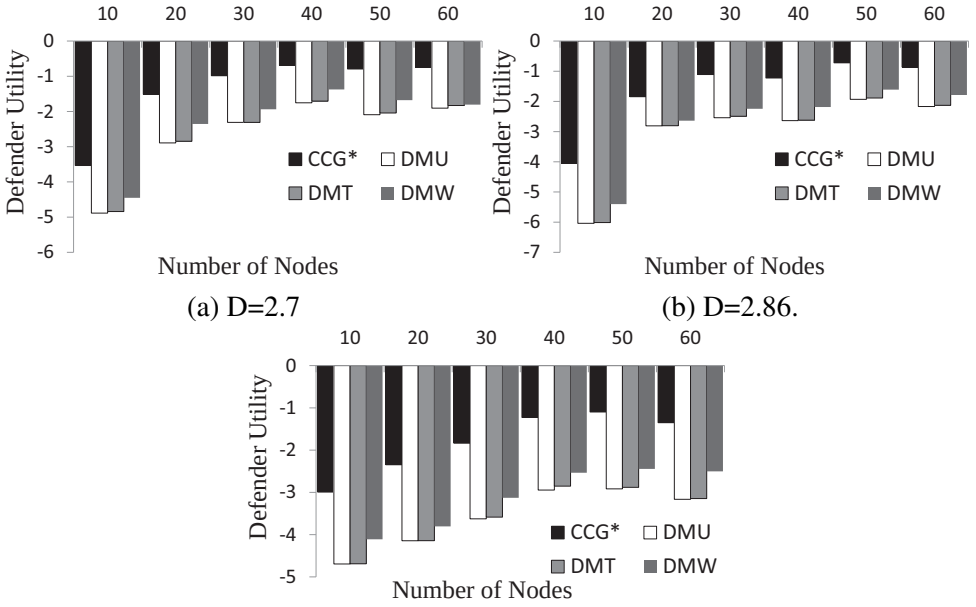


FIGURE 4.3: Scalability of algorithms for solving NFIGs.

We compare the scalability of our algorithms on *WG* graphs with varying degrees: $D = 2.86$ which is the mean degree of road network [93], and $D \in \{2.7, 3.0\}$. $\alpha = 0.3$ and $\beta = 0.15$. The results are depicted in Figure 4.3, where *DO* cannot scale up to 50 nodes for $D = 2.7$ and $D = 2.86$ and 40 nodes for $D = 3.0$ with runtime cap of 1800 seconds, while *LP* cannot scale up to 30 nodes with memory cap of 8GB. The results show that although our approaches (*CCG*,*CCG**) involve nonlinear programs (4.8)

and (4.12), they still outperform *LP* and *DO* significantly due to the compact representation. Besides, *CCG** also outperforms *CCG* a lot which shows that the greedy methods (Algorithms 7 and 8) can obtain good enough solutions. Especially *CCG** can scale up to realistic-sized networks with 60 nodes and over *40* inspection stations and *20* resources, while the number of tactical checkpoints operated in the Southern border is 39 in 2009 [46].

4.6.2 Solution Quality Analysis

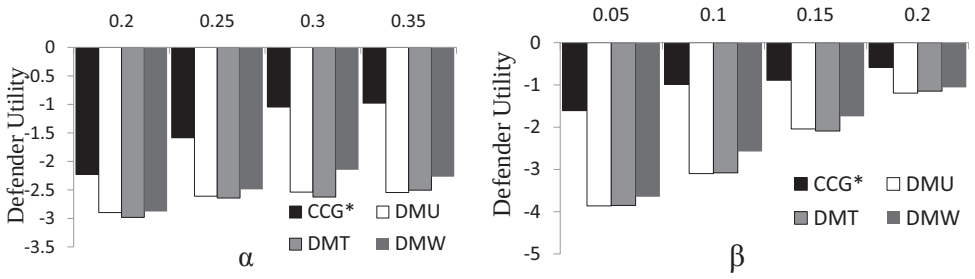


(a) D=2.7 (b) D=2.86.

(c) D=3.0.

FIGURE 4.4: Solution quality of CCG (CCG*) for solving NFIGs.

We also compare the solution qualities of our algorithms with three baseline strategies on *WG* graphs with varying degrees ($\alpha = 0.3$ and $\beta = 0.15$), demonstrated by Figure 4.4. It is clear that *CCG** outperforms these baselines significantly. Besides, the baseline *DMW* works best among the three baselines since the inspection weights well measure the capability of inspection stations. Note that the numbers of inspection stations and resources are proportional with the network size, and hence there exists no monotone relationship between utility and network size.



(a) Varying α ($\beta = 0.15$)

(b) Varying β ($\alpha = 0.3$)

FIGURE 4.5: Varying α and β.

4.6.2.1 Varying Values of α and β

Figure 4.5 shows the solution qualities of *CCG** and three baselines on *WG* graphs ($|V| = 40$, $D = 2.86$) with varying values of α and β, from which we can see: i) with fixed β, increasing α improves the defender utility for more available inspection stations; ii) with fixed α, increasing β also improves the defender utility for more available resources. Moreover, *CCG** outperforms these baselines significantly for all tested values of α and β.

4.6.2.2 Robustness

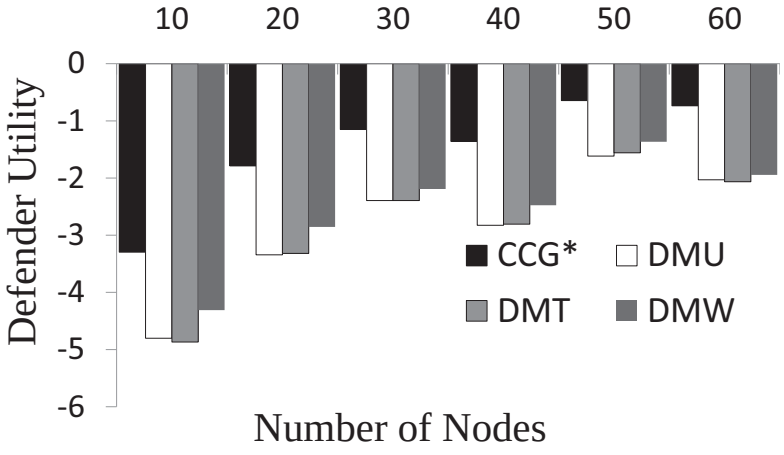
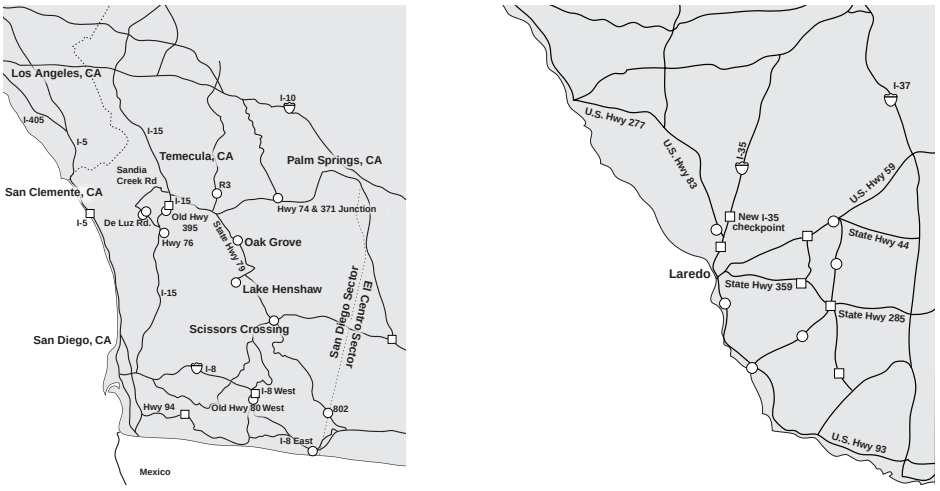


FIGURE 4.6: Robustness (20%).

In reality, the attacker might not have perfect observation on the defender strategy. Thus, we compare the solution qualities of *CCG** and baselines on *WG* graphs ($|V| = 40$, $D = 2.86$, $\alpha = 0.3$ and $\beta = 0.15$) with the attacker's estimation $\mathbf{x}'$ of the defender strategy $\mathbf{x}$ randomly generated by: $\mathbf{x}' = \tilde{\mathbf{x}}/|\tilde{\mathbf{x}}|$ where $\tilde{x}_S \sim x_S \cdot [1-\delta, 1+\delta]$. The

solution quality of $\mathbf{x}$ is measured by $U_d(\mathbf{x}, \mathbf{f}')$ with $\mathbf{f}'$ being the attacker best response against $\mathbf{x}'$. The result is depicted in Figure 4.6 with $\delta = 20\%$, and the result shows that our approach *CCG** is robust enough and outperforms baselines significantly under a high level of observation uncertainty.

4.6.2.3 Application on Southern Border Network



(a) San Diego Sector (b) Laredo Sector

FIGURE 4.7: San Diego and Laredo sectors (circles are tactical checkpoints).

We conduct experiments on two sectors in Southern Border Patrol [94]: i) San Diego sector with 29 nodes and degree of 2.89, two source nodes (San Ysidro and Tecate POEs), one sink node (Los Angeles), and 12 checkpoints (Figure 4.7a); ii) Laredo sector with 20 nodes and degree of 3, one source node (Laredo POE), one sink node (San

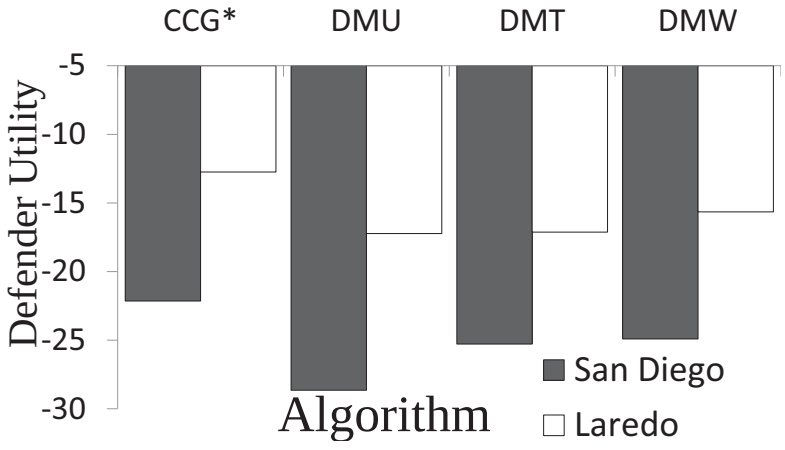


FIGURE 4.8: Border patrol sectors.

Antonio), and 6 checkpoints (Figure 4.7b). The roads can be classified into four levels: Interstate, U.S. Hwy, State Hwy, and Hwy. The capacity of a road is randomly drawn from: [0, 35] (Interstate), [0, 30] (U.S. Hwy), [0, 25] (State Hwy), and [0, 20] (Hwy). $k = 6$ for San Diego sector and $k = 3$ for Laredo sector. The results are depicted in Figure 4.8, showing that our approach significantly outperforms the baseline methods for the realistic networks.

4.7 Chapter Summary

This chapter studies the problem of optimally interdicting an illegal network flow. We introduce a novel Stackelberg game model called *NFIG*, and propose a *Column and Constraint Generation (CCG)* algorithm to solve it. The computational complexity of its *ColG* subproblem is analyzed. Several novel algorithms are provided to solve the two subproblems including convex optimization, $1-\frac{1}{e}$ approximation method and a heuristic search algorithm. Experimental evaluation shows that *CCG* scales up to realistic-sized networks and significantly outperforms existing methods and heuristics.

Chapter 5

Combating Repeated Attacks on the Network

Last chapter as well as earlier works model network interdiction as a one-shot leader-follower game, where the defender randomly allocates the resources first and the adversary, knowing the defender strategy with extensive surveillance, chooses an optimal action to respond. Though the minimax solution is robust and guarantees the defender the value of the game, such models are not ideal to capture many realistic scenarios, such as illegal drug interdiction, due to several unreasonable assumptions. First, it is well understood that complete rationality, unlimited observation and high level computational ability are not ideal for modeling human adversaries [61, 95]. Second, the defender's prior knowledge of the adversary and the environment is extremely limited in practice, while existing works in network interdiction assume full information on the defender side. Finally, the one-shot game model neglects the repeated interactions between the defender and the adversary, which are common in many network interdiction domains. For example, it is reported that over one thousand tons of drugs are seized on land in 2009 within the United States [49], and the U.S. border patrol agency has to alter the checkpoint operation policy frequently against unknown and fickle smugglers [46].

Existing works typically deal with part of these challenges, either the bounded rationality of the attacker or the repeated interactions. Most of them in the security game literature typically focus on bounded rationality of the attacker by following the concept of

Quantal Response [96], in order to predict the future move of the attacker. Such models include, but are not limited to, e.g., Subjective Utility Quantal Response (SUQR) [62], Bayesian SUQR [8] and Robust SUQR [97]. However, those models suffer from a number of limitations [50]. In particular, most of them failed to capture the attacker's adaptive behavior of changing the attacking strategy based on defender's past moves. Besides, all these approaches only focus on simple target protection scenarios.

Another line of related work applies online learning approaches on repeated security games [22, 23]. Unfortunately, the scope of all these works is limited to the target protection scenarios and cannot be applied to the network security domain, where the defender utility is a complex submodular function (we will show later). The straightforward approaches for online submodular maximization have poor performance in network security domain. In particular, the state-of-the-art algorithm for the online submodular maximization problem is an online version of the greedy algorithm [51], which only achieves a low $(1 - \frac{1}{e})$-regret and the performance guarantee only holds for the *oblivious* adversary.

We are the first to study the network interdiction from a perspective that aims to address all these challenges. In particular, inspired by the work of Xu *et al.* [23], we apply the adversarial online learning approaches to tackle the network interdiction problem. However, we have to emphasize that this application is in fact non-trivial, since the repeated network interdiction problem is formulated as an online submodular maximization problem due to the network structure, for which state-of-the-art learning algorithms fail due to their poor performance. To overcome this issue, we linearize our learning problem by using a novel decision space transformation. The cost of this linearization transformation, however, is the exponentially increased size of the decision space. As such, a direct application of existing online linear learning methods, such as geometric hedge [98], FPL [99] and Exp3 [100], will fail, due to the large decision space. Thus, we propose a new online linear learning algorithm, which exploits the semi-bandit style feedback in the network interdiction domain, called *Semi-Bandit style Geometric decision algorithm against an Adaptive adversary (SBGA)* to provide online decisions for the defender. We formally show that SBGA achieves $\mathcal{O}(T^{2/3})$ regret bound against *adaptive* adversary compared with the best fixed strategy on hindsight, and a low regret

compared with the near optimal adaptive defender strategy. Finally, extensive numerical evaluations are conducted to demonstrate that our approach outperforms existing online learning approaches significantly and achieves much faster convergence rate. In particular, SBGA requires significantly fewer rounds (typically 50 rounds) to obtain satisfactory low average regret solution against various adversarial behavior models, and can efficiently scale up to realistic sized game instances.

Although we provide a novel transformation from online submodular maximization to online linear optimization, existing approaches for online linear optimization cannot address all the brutal challenges that we face: *scalability*, *limited feedback* and *adaptive adversary*. Most works related to online linear optimization typically focus on the *oblivious* adversary [101–103]. Bartlett *et al.* [98] provided the Geometric Hedge algorithm with $\mathcal{O}(\sqrt{T})$ regret against adaptive adversary with high probability. However, their algorithm needs to store and compute on the entire exponentially large decision space. The most relevant approach is *Bandit-style Geometric decision algorithm against an Adaptive adversary (BGA)* (Sec. 2.2.1.2) which achieves sublinear regret against adaptive adversary with limited feedback [52, 53]. However, the empirical evaluation shows that its regret convergence rate is very low in practice since it fails to exploit the unique semi-bandit feedback in our problem, and thus the variation of the unbiased estimation could be significantly high.

The remaining content of the Chapter is organized as follows: Section 5.1 presents the model of repeated network interdiction problems. Section 5.2 transforms the online submodular maximization problem into an online linear optimization problem. Section 5.3 illustrates our SBGA algorithm to solve the online linear optimization problem with semi-bandit feedback. Section 5.4 shows the theoretical analysis of SBGA algorithm. Section 5.5 presents the experimental results on the performance of our SBGA algorithm. Section 5.6 presents an implementation of the methodologies proposed in this chapter to solve a game theoretical online promotion of exergames for the elderly. Section 5.7 summarizes this Chapter.

5.1 Repeated Network Interdiction Games

The repeated *Network Interdiction Games (NIG)* model is the repeated version of network interdiction models, such as NFIG in the previous chapter, and the escaping path interdiction model [29, 31]. At each iteration, the game shares the same settings as NFIG (see Sec. 4.2), except that $\mathcal{P}$ denotes the set of candidate s-t paths in G for the attacker, such as the eight drug transportation corridors in the United States [49], instead of all possible paths[1]. We denote by m the number of candidate paths, i.e., $m = |\mathcal{P}|$. The repeated NIG is illustrated as follows.

Repeated Network Interdiction Game

for $t = 1, 2, ...T$

1. The defender chooses an allocation $S^t \in \mathcal{S}$ to play.
2. The attacker chooses a network flow $\mathbf{f}^t \in \mathcal{F}$ to play.
3. The defender observes the amount of interdicted flow at each operated checkpoint in S^t and receives the utility $U_d(S^t, \mathbf{f}^t)$.

5.1.1 Semi-bandit Feedback

Unlike existing works which assume plenty of available information for both players, the defender in repeated NIG has no prior knowledge of the adversarial behavior. The information available for the defender only includes the network structure G, candidate paths for the adversary $\mathcal{P}$, the inspection probability τ_i on each checkpoint i and an upper bound U of the total amount of all the feasible flow, i.e., $U \geq \arg\max_{\mathbf{f} \in \mathcal{F}} |\mathbf{f}|_1$. The knowledge of U can be interpreted as the fact that the defender, although may not know the exact capacity on each edge, can easily derive an upper bound of such capacity according to the domain knowledge and the past experience. W.l.o.g., we set U to 1.

[1]The selected candidate paths contribute the majority of the illegal network flow, so that the complexity of the model is reduced without a significant loss of optimality.

Regarding the feedback of past plays, a natural assumption for the repeated NIG is that the defender only knows the amount of flow interdicted by each operated checkpoint. Such information on each operated checkpoint is called the *semi-bandit feedback*. On the other side, the *only* requirement for the attacker is that he cannot observe the real-time defense S^t before round t starts since the game is simultaneously played at each round. Furthermore, we assume that the defender is an expected utility maximizer, while no behavior model for the attacker is required. In other words, the attacker could be an expected utility maximizer or irrational to any extent; could be fully adversarial, or a random player.

The above repeated NIG model is general for many realistic network security domains. For example, in the illegal drug trafficking scenario, the drug smuggling activity on the transportation network can be formulated as an adversarial network flow. The adversarial behavior, on the other hand, is hard to capture as the defender is facing several sophisticated drug trafficking cartels who hire people with unknown backgrounds to smuggle illegal drug [84]. Besides, as pointed out by the government report, the checkpoint operation policy almost changes daily to deal with the fickle smugglers [46], which is captured by the repeated manner of our model. Another example is the escaping path interdiction problem [29], where the escaping path of the attacker can be modeled by a unit flow on the path. The roadblock placed by the defender can be treated as a checkpoint with interdiction probability 1.

5.1.2 Regret

Let $\mathcal{H}_t$ denote the history information of the game by time t (inclusive), and $\mathcal{H}_0$ denote no history information at all. Given a sequence of adversarial flows $\mathbf{f}^1, ..., \mathbf{f}^T$, where $\mathbf{f}^t$ may be adaptive depending on $\mathcal{H}_t$, we are interested in designing an *online* policy $S^1(\mathcal{H}_0), ..., S^T(\mathcal{H}_{T-1})$ (possibly randomized) that maximizes the defender's expected utility $\mathbb{E}[\sum_{t=1}^{T} U_d(S^t, \mathbf{f}^t)]$, where the randomization is taken over the randomness of the policy and the environment. Alternatively, we aim at minimizing the defender's *regret*:

$$R_T = \max_{S \in \mathcal{S}} \sum_{t=1}^{T} U_d(S, \mathbf{f}^t) - \mathbb{E}[\sum_{t=1}^{T} U_d(S^t, \mathbf{f}^t)].$$

The first term $\max_{S\in\mathcal{S}} \sum_{t=1}^{T} U_d(S, \mathbf{f}^t)$ is the utility of the best hindsight allocation, which serves as the benchmark. Without any prior knowledge of the adversarial behavior, we take the worst case analysis and focus on the largest regret against all possible adaptive adversaries. This type of regret notion is common in the online learning theory literature [104].

5.1.3 Online Submodular Maximization

As shown by Proposition 5.1, the defender utility function is submodular with allocation S. At the first glance, we may relate the repeated NIG with the online submodular maximization problem where the decision-maker faces a sequence of submodular reward function and chooses a fix sized subset to play at each round. However, existing approaches for online submodular maximization cannot perform well in repeated NIG. As such, we propose a novel approach to solve the repeated NIG efficiently with low regret solution. In particular, we first provide a non-trivial transformation from the repeated NIG as an online submodular maximization problem to an online linear optimization problem. To solve the transformed online linear optimization problem, we propose a novel algorithm called SBGA which exploits the semi-bandit feedback on operated checkpoints and obtains low regret solutions.

Proposition 5.1. *The defender utility function $U_d(S, \mathbf{f})$ is submodular with allocation S.*

Proof. For each $S \subseteq \mathcal{I}$ and $i \in \mathcal{I} \setminus S$, we have:

$$U_d(S \cup \{i\}, \mathbf{f}) - U_d(S, \mathbf{f}) = \sum\nolimits_{p\in\mathcal{P}:i\in p} \tau_i \cdot \Phi(S, p) f_p.$$

Since $\Phi(S', p) \geq \Phi(S'', p)$ for any pair of $S' \subseteq S''$, $U_d(S, \mathbf{f})$ is submodular with allocation S when $\mathbf{f}$ is fixed. □

5.2 From Online Submodular Maximization to Online Linear Optimization

In this section, we show a transformation of repeated NIG to an online linear optimization problem for which efficient low regret solution is possible. The intuition is as follows: if we look at the defender utility function $U_d(S, \mathbf{f})$ in details, we can find that the only parameter unknown is the adversarial flow $\mathbf{f}$. Fortunately, $U_d(S, \mathbf{f})$ is actually linear with $\mathbf{f}$. In other words, we can write down $U_d(S, \mathbf{f})$ as $\mathbf{w} \cdot \mathbf{f}$ where the coefficient $\mathbf{w}$, depending on the allocation S, is actually known to the defender since the interdiction probability can be estimated. As such, we define a mapping $\phi : \mathcal{S} \rightarrow \mathbb{R}^m$ such that for each allocation $S \in \mathcal{S}$, $\phi(S) = \mathbf{w} = \langle w_p \rangle$ with $w_p = 1 - \Phi(S, p)$, $\forall p \in \mathcal{P}$. Given an allocation S and $\mathbf{w} = \phi(S)$, we have: $U_d(S, \mathbf{f}) = \mathbf{w} \cdot \mathbf{f}$.

Therefore, instead of regarding the repeated NIG as allocating resources for a sequence of unknown submodular functions, we can treat it as selecting decision points $\mathbf{w}$ for the coming unknown linear functions. The later one is an online linear optimization problem, shown as follows.

Online Linear Optimization Problem

for $t = 1, 2, ...T$

1. The defender chooses a point $\mathbf{w}^t \in \phi(\mathcal{S})$.
2. The attacker chooses a network flow $\mathbf{f}^t \in \mathcal{F}$ to play.
3. The defender receives the utility $\mathbf{w}^t \cdot \mathbf{f}^t$.

$\phi(\mathcal{S}) = \{\mathbf{w} | \exists S \in \mathcal{S} : \mathbf{w} = \phi(S)\}$ denotes the set of points mapped from $\mathcal{S}$ with ϕ. However, the most relevant existing approach BGA fails to solve such an online linear optimization problem due to the slow regret convergence rate as shown in experimental evaluation. The main drawback of BGA is that once an allocation S is played, only the total amount of interdicted flow of S is utilized to learn the unknown adversarial flow. In this case, BGA has to put a lot of effort in exploration of estimating the adversarial flow. However, as we will show later, by further exploiting the semi-bandit feedback on

each operated checkpoint of S, the exploration efficiency can be improved significantly. Based on this idea, we propose the SBGA algorithm with nice and provable guarantees and satisfactory practical performance.

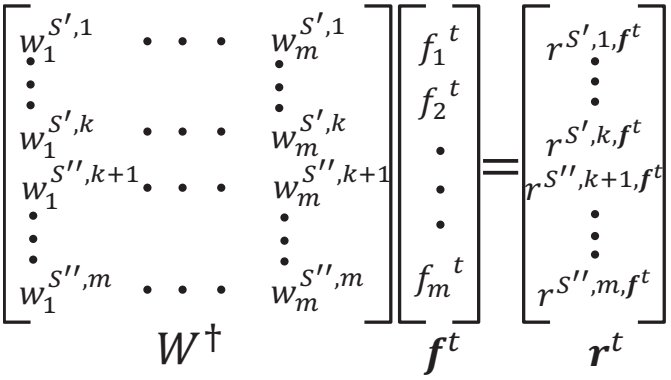
5.3 SBGA

As depicted in Algorithm 10, the intuition of SBGA is as follows: suppose that the full information of the adversarial flows of past plays can be accessed in repeated NIG, then an efficient online algorithm with low regret against adaptive adversary is available, denoted as *Geometric EXperts algorithm (GEX)* (Algorithm 11). Unfortunately, we only have semi-bandit feedback in repeated NIG. As such, SBGA samples from an *exploration basis* $\mathcal{S}^{eb}$, exploits the received semi-bandit feedback on operated checkpoints, and builds an "imaginary" full information game where the unknown adversarial flows (of past plays) are replaced by their unbiased estimators $\hat{\mathbf{f}}$. The algorithm GEX is then applied on the "imaginary" repeated NIG to provide online decisions. We now describe in details each components of SBGA.

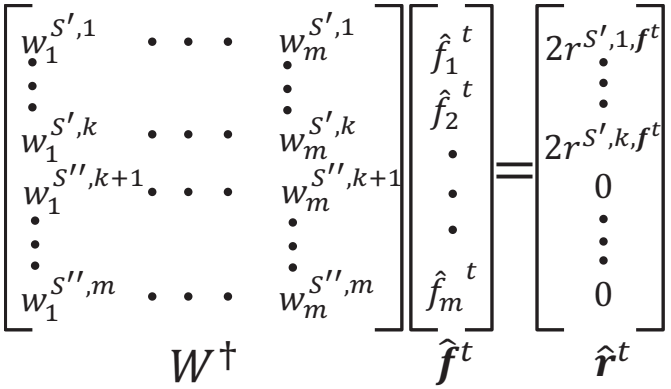
5.3.1 GEX

Suppose that the adversarial flows $\mathbf{f}^1, ..., \mathbf{f}^{t-1}$ of previous rounds are known. We adopt the algorithm proposed by Kalai and Vempala [105] to serve as GEX subroutine, shown in Algorithm 11. In particular, GEX returns the optimal decision $\mathbf{w}$ in $\phi(\mathcal{S})$ against the cumulative flow of previous rounds perturbed by a random noise vector $\mathbf{z}$. The scale of $\mathbf{z}$ is controlled by a learning parameter ϵ, which balances the tradeoff between *exploitation* of playing the best hindsight decision so far, and *exploration* of adding perturbations to make the algorithm less predictable, especially for the adaptive adversary. Such tradeoff between exploitation and exploration is essential in the online learning theory, and GEX successfully addresses it by setting ϵ to a delicate value and achieves sublinear $\mathcal{O}(\sqrt{T})$ regret [105].

5.3.2 Exploration and Unbiased Estimator



(a) True adversarial flow



(b) Exploration with $\mathcal{S}^{eb} = \{S', S''\}$

FIGURE 5.1: Exploration with $\mathcal{S}^{eb} = \{S', S''\}$

As we mentioned before, at round t, SBGA creates an unbiased estimator $\hat{\mathbf{f}}^t$ of the unknown adversarial flow $\mathbf{f}^t$ by sampling from an exploration basis (with some probability), and hence an "imaginary" game, for which GEX is applicable, is constructed. To do so, we first show that the semi-bandit feedback on each operated checkpoint is linear with the unknown adversarial flow. Specifically, for each allocation S, let $r^{S,i,\mathbf{f}}$ denote the amount of interdicted flow on operated checkpoint $i \in S$ against adversarial flow $\mathbf{f}$ when S is played. We have:

$$r^{S,i,\mathbf{f}} = \sum\nolimits_{p \in \mathcal{P}} \prod\nolimits_{j \in S(p,i)} (1 - \tau_j)\tau_i \alpha_{p,i} \cdot f_p \quad \forall i \in S$$

where $S(p, i)$ denotes the set of checkpoints in S such that path p passes through before checkpoint i, and $\alpha_{p,i}$ is the indicator which takes value 1 if i is on path p and 0

otherwise. Let $\mathbf{w}^{S,i}$ denote the vector such that:

$$w_p^{S,i} = \prod_{j \in S(p,i)} (1 - \tau_j)\tau_i \cdot \alpha_{p,i} \quad \forall p \in \mathcal{P}. \tag{5.1}$$

We have: $r^{S,i,\mathbf{f}} = \mathbf{w}^{S,i} \cdot \mathbf{f}$ for each checkpoint $i \in S$, and $\mathbf{w}^{S,i}$ is known to the defender. Once S is played, $r^{S,i,\mathbf{f}}$ is revealed for each $i \in S$. Let $m \times k$ matrix $W^S = \bigcup_{i \in S} \mathbf{w}^{S,i}$.

To illustrate the exploration in SBGA, consider an example where $m = 2k$ and let $\mathcal{S}^{eb} = \{S', S''\}$, as shown in Figure 5.1. To simplify the explanation, suppose that the checkpoints set $\mathcal{I} = \{1, ..., n\}$ and $n \geq m$. Allocation S' operates the first k checkpoints in $\mathcal{I}$ and S'' operates checkpoints $\{k+1, ..., m\}$. Let $m \times m$ matrix $W = [W^{S'}, W^{S''}]$, whose transpose $W^\dagger$ is illustrated in the figure. Each element of $W^\dagger$ is defined in (5.1). Assume that W is full rank. At round t, let $\mathbf{r}^t$ be the vector of semi-bandit feedbacks as shown in Figure 5.1a, and the true adversarial flow satisfies $W^\dagger \mathbf{f}^t = \mathbf{r}^t$. However, $\mathbf{r}^t$ is not available since only one allocation is played at round t, which only reveals half of $\mathbf{r}^t$. Thus, we randomly pick one allocation in $\mathcal{S}^{eb}$ to play and estimate $\mathbf{r}^t$ with $\hat{\mathbf{r}}^t$, as shown in Figure 5.1b where S' is drawn and semi-bandit feedbacks of S' are doubled in $\hat{\mathbf{r}}^t$. We estimate the flow with $\hat{\mathbf{f}}^t = (W^\dagger)^{-1}\hat{\mathbf{r}}^t$ and it is easy to see that $\mathbb{E}[\hat{\mathbf{f}}^t] = \mathbf{f}^t$ since $\mathbb{E}[\hat{\mathbf{r}}^t] = \mathbf{r}^t$.

In general, assume that m is dividable by k and let $\bar{m} = \frac{m}{k}$. Let $\mathcal{S}^{eb}$ be an *exploration basis* of $\bar{m}$ allocations, and let $m \times m$ matrix $W = \bigcup_{S \in \mathcal{S}^{eb}} W^S = [\mathbf{b}^1, ..., \mathbf{b}^m]$ where the l-th column is denoted as $\mathbf{b}^l$. W.l.o.g., suppose W to be full rank.[1] At round t, as illustrated in Algorithm 10, with probability λ, SBGA uniformly samples one allocation S in $\mathcal{S}^{eb}$ to play (line 10), and observes the interdicted flow for each operated checkpoint i with amount $r^{S,i,\mathbf{f}^t} = \mathbf{w}^{S,i} \cdot \mathbf{f}^t$ (line 13). SBGA estimates flow $\mathbf{f}^t$ according to the idea mentioned before. The estimation $\hat{\mathbf{f}}^t = (W^\dagger)^{-1}\hat{\mathbf{r}}^t$ where the vector $\hat{\mathbf{r}}^t$ is defined as follows: $\hat{r}_l^t = (\bar{m}/\gamma)\mathbf{b}^l \cdot \mathbf{f}^t$ if $\mathbf{b}^l \in W^S$ and $\hat{r}_l^t = 0$ otherwise (lines 14 & 15). Proposition 5.2 shows that $\mathbb{E}[\hat{\mathbf{f}}^t] = \mathbf{f}^t$.

Proposition 5.2. *$\hat{\mathbf{f}}^t$ in SBGA is an unbiased estimator of $\mathbf{f}^t$.*

[1]These assumptions are only for simplifying the analysis. The algorithm and analysis work for general cases where usually $\bar{m} = \lceil \frac{m}{k} \rceil$. In practice, the full rank assumption of W may not hold true for any set of $\bar{m}$ allocations. Thus, the allocation with larger rank of W^S will be selected into $\mathcal{S}^{eb}$ to reduce the size of exploration basis.

Proof. Let $\hat{\mathbf{r}}^{S,t}$ denote the vector $\hat{\mathbf{r}}^t$ when $S \in \mathcal{S}^{eb}$ is drawn. $\hat{r}_l^{S,t} = (\bar{m}/\gamma)\mathbf{b}^l \cdot \mathbf{f}^t$ if $\mathbf{b}^l \in W^S$, and we have:

$$\sum\nolimits_{S\in\mathcal{S}^{eb}} \hat{\mathbf{r}}^{S,t} = (\bar{m}/\gamma)W^\dagger \mathbf{f}^t. \tag{5.2}$$

$$\begin{aligned}\mathbb{E}[\hat{\mathbf{f}}^t] &= (W^\dagger)^{-1}\mathbb{E}[\hat{\mathbf{r}}^t] = (W^\dagger)^{-1}\frac{\gamma}{\bar{m}}\sum\nolimits_{S\in\mathcal{S}^{eb}} \hat{\mathbf{r}}^{S,t} \\ &= (W^\dagger)^{-1}W^\dagger\mathbf{f}^t = \mathbf{f}^t.\end{aligned}$$

□

5.3.3 Exploitation with GEX

At this point, the unbiased estimator of the adversarial flow is obtained. At each round t, with probability $1-\lambda$, SBGA applies GEX routine to *exploit* the "imaginary" game, where the adversarial flows are replaced by their unbiased estimators $\hat{\mathbf{f}}^1, ..., \hat{\mathbf{f}}^{t-1}$, to generate decision $\mathbf{w}^t$ and the corresponding allocation S^t to play (line 6).[1] Parameter λ balances the tradeoff between exploitation of playing decisions provided by GEX and exploration of sampling from $\mathcal{S}^{eb}$ to obtain an low-variance estimation of adversarial flow. However, it is not clear yet whether SBGA can achieve acceptable sublinear regret against adaptive adversary since the analysis of existing approaches aimed at cost minimization [52], while the repeated NIG is formulated as a utility maximization problem. The following section shows that the answer is affirmative.

5.4 Theoretical Analysis

We first show that SBGA achieves a sublinear regret $\mathcal{O}(T^{2/3})$ against the adaptive adversary, compared with the best fixed allocation on hindsight.

Theorem 5.3. *If $\bar{m} = 1$, set $\gamma = T^{-1/3}$ and $\epsilon = \sqrt{\frac{m}{T}}$ (roughly optimal), we have: $R_T(SBGA) \leq 5m^{1/2}T^{2/3}$; Otherwise, set $\gamma = \bar{m}T^{-1/3}$ and $\epsilon = \frac{1}{m}\sqrt{\gamma/T}$ (roughly optimal), we have: $R_T(SBGA) = \mathcal{O}((\beta_\infty + C)m\sqrt{\bar{m}}T^{2/3})$, where $\beta_\infty = \|(W^\dagger)^{-1}\|_\infty$ and C is the spanning ratio of W with respect to $\phi(\mathcal{S})$ defined below (Definition 4).*

[1] In order to apply the approach of Kalai and Vempala [105], the estimated flows need to be transformed to meet its requirements, which is omitted here for the ease of reading, and the readers can refer to appendix A of [52] for details.

Algorithm 10: SBGA

Parameter: γ and ϵ, where ϵ is a parameter of GEX
Initialize $\mathcal{S}^{eb}$, let $W = \bigcup_{S\in\mathcal{S}^{eb}} W^S = [\mathbf{b}^1, .., \mathbf{b}^m]$
for $t = 1, ..., T$ **do**
  Let $\chi^t = 1$ with probability γ and $\chi^t = 0$ otherwise
  if $\chi^t = 0$ **then**
    Select $\mathbf{w}^t$ from the distribution GEX($\hat{\mathbf{f}}^1, ..., \hat{\mathbf{f}}^{t-1}$)
    Receive utility $u^t = \mathbf{w}^t \cdot \mathbf{f}^t$
    $\hat{\mathbf{f}}^t = \mathbf{0} \in \mathbb{R}^m$
  else
    Draw S uniformly at random from $\mathcal{S}^{eb}$
    $\mathbf{w}^t = \phi(S)$
    Receive utility $u^t = \mathbf{w}^t \cdot \mathbf{f}^t$
    Observe interdicted flow at each $i \in S$: $r^{S,i,\mathbf{f}^t}$
    Let $\hat{\mathbf{r}}^t \in \mathbb{R}^m$ by $\hat{r}^t_l = 0$ for $\mathbf{b}^l \notin W^S$ and $\hat{r}^t_l = (\bar{m}/\gamma)\mathbf{b}^l \cdot \mathbf{f}^t$ otherwise
    $\hat{\mathbf{f}}^t = (W^\dagger)^{-1}\hat{\mathbf{r}}^t$

Algorithm 11: GEX

Parameter: ϵ, network flows $\mathbf{f}^1, ..., \mathbf{f}^{t-1}$ of previous rounds
Define a noise vector $\mathbf{z} \in \mathbb{R}^m$ such that z_i is uniformly drawn from $[0, \frac{1}{\epsilon}]$
$\mathbf{w}^t = \arg\max_{\mathbf{w}\in\phi(\mathcal{S})} \mathbf{w} \cdot (\mathbf{f}^1 + ... + \mathbf{f}^{t-1} + \mathbf{z})$
return $\mathbf{w}^t$

Definition 4. Given a basis of $\mathbb{R}^m$: $W = \{\mathbf{b}^1, ..., \mathbf{b}^m\}$, the spanning ratio of W with respect to the subset $H \subseteq \mathbb{R}^m$ is the minimal value of C such that: for any point $\mathbf{w} \in H$, we can write $\mathbf{w} = \sum_{l=1}^m \alpha_l \mathbf{b}^l$ where the coefficient $\alpha_l \in [-C, C]$.

A quick comparison of the regret bound of SBGA with the state of the art one of BGA in [53], which is $\mathcal{O}(mT^{2/3})$, shows that both have the same dependence of T which is to the power of $2/3$, and thus, both satisfy the Hannan consistency. When $\bar{m} = 1$, SBGA has better reliance on the number of paths m and the regret bound is strictly less than that of BGA. However, when $\bar{m} \geq 2$, the dependence of SBGA over m is unknown yet due to the term $(\beta_\infty + C)$ which relies on the choice of W.

5.4.1 Proof of $\mathcal{O}(T^{2/3})$ Regret

We first explain some necessary notations. Let $\mathbf{r}^t \in [0, 1]^m$ denote the vector where $r^t_l = \mathbf{b}^l \cdot \mathbf{f}^t$ for $\mathbf{b}^l \in W$. $\hat{\mathbf{r}}^t \in \mathbb{R}^m$ is SBGA's estimation of $\mathbf{r}^t$ in Algorithm 1. $\mathbf{w}^t \in \phi(\mathcal{S})$

is the decision made by SBGA in round t. Let $\hat{\mathbf{w}}^t \in \phi(\mathcal{S})$ be the decision recommended by GEX in round t, and $\hat{\mathbf{w}}^t = \mathbf{w}^t$ if SBGA does not sample the exploration basis in round t. $u^t \in [0,1]$ is SBGA's utility in round t, i.e., $u^t = \mathbf{w}^t \cdot \mathbf{f}^t$. Let $\hat{u}^t \in \mathbb{R}$ denote the utility of GEX in the "imaginary" game, i.e., $\hat{u}^t = \hat{\mathbf{w}}^t \cdot \hat{\mathbf{f}}^t$. Let $\mathcal{G}^t$ denote the history of the game by time t (inclusive), which consists of all valid information of the game played so far. In particular, if we condition on a history $\mathcal{G}^t$, the random variables $\mathbf{f}^1, ..., \mathbf{f}^t, \hat{\mathbf{r}}^1, ..., \hat{\mathbf{r}}^t, \hat{\mathbf{f}}^1, ..., \hat{\mathbf{f}}^t, \mathbf{w}^1, ..., \mathbf{w}^t$ and $\chi^1, ..., \chi^t$ are fully determined. Let $\mathbf{w}^{1:t}$ be a sequence of decisions $\{\mathbf{w}^1, ..., \mathbf{w}^t\}$ and let $\mathbf{f}^{1:t}$ be a sequence of adversarial flows $\{\mathbf{f}^1, ..., \mathbf{f}^t\}$. We denote

$$\begin{aligned} utility(\mathbf{w}^{1:T}, \mathbf{f}^{1:T}) &= \sum\nolimits_{t=1}^{T} \mathbf{w}^t \cdot \mathbf{f}^t \\ best(\mathbf{f}^{1:T}) &= \arg\max\nolimits_{\mathbf{w} \in \phi(\mathcal{S})} \sum\nolimits_{t=1}^{T} \mathbf{w} \cdot \mathbf{f}^t \\ opt(\mathbf{f}^{1:T}) &= best(\mathbf{f}^{1:T}) \cdot \sum\nolimits_{t=1}^{T} \mathbf{f}^t. \end{aligned}$$

Theorem 5.3 can be directly derived from four inequalities:

$$\mathbb{E}[opt(\mathbf{f}^{1:\mathbf{T}})] \leq T \qquad \text{since } \|\mathbf{f}^t\|_1 \leq 1 \tag{5.3}$$

$$(1-\gamma)\mathbb{E}[utility(\hat{\mathbf{w}}^{1:T}, \hat{\mathbf{f}}^{1:T})] - \mathbb{E}[utility(\mathbf{w}^{1:T}, \mathbf{f}^{1:T})] \leq \gamma T \tag{5.4}$$

$$\begin{aligned} &\mathbb{E}[opt(\hat{\mathbf{f}}^{1:T})] - \mathbb{E}[utility(\hat{\mathbf{w}}^{1:T}, \hat{\mathbf{f}}^{1:T})] \\ &\qquad \leq \begin{cases} (8m+2)C\sqrt{T/\gamma}, & \text{if } \bar{m} \geq 2; \\ 2\sqrt{mT}, & \text{if } \bar{m} = 1. \end{cases} \end{aligned} \tag{5.5}$$

$$\begin{aligned} &\mathbb{E}[opt(\mathbf{f}^{1:\mathbf{T}})] - \mathbb{E}[opt(\hat{\mathbf{f}}^{1:T})] \\ &\qquad \leq \begin{cases} m\bar{m}(\beta_\infty + 1)\sqrt{T/\gamma}, & \text{if } \bar{m} \geq 2; \\ \sqrt{mT/\gamma}, & \text{if } \bar{m} = 1. \end{cases} \end{aligned} \tag{5.6}$$

Proof of (5.4). We follow the similar procedure to prove (5.4) as the proof of Theorem 3 in [52].

$$\bar{\mathbf{w}}^t = \sum\nolimits_{\hat{\mathbf{w}}^t \in \phi(\mathcal{S})} Pr(\hat{\mathbf{w}}^t | \mathcal{G}^{t-1}) \hat{\mathbf{w}}^t.$$

Let $\hat{\mathbf{r}}^{t,S}$ and $\hat{\mathbf{f}}^{t,S}$ be the estimators of $\mathbf{r}^t$ and $\mathbf{f}^t$ when $S \in \mathcal{S}^{eb}$ is sampled in round t, according to (5.2):

$$\sum\nolimits_{S\in\mathcal{S}^{eb}} \hat{\mathbf{f}}^{t,S} = (W^\dagger)^{-1} \sum\nolimits_{S\in\mathcal{S}^{eb}} \hat{\mathbf{r}}^{t,S} = \frac{\bar{m}}{\gamma}\mathbf{f}^t. \tag{5.7}$$

$$\begin{aligned}
\mathbb{E}[\hat{u}^t|\mathcal{G}^{t-1}] &= \gamma \sum\nolimits_{S\in\mathcal{S}^{eb}} \frac{1}{\bar{m}} \sum\nolimits_{\hat{\mathbf{w}}^t\in\phi(\mathcal{S})} Pr(\hat{\mathbf{w}}^t|\mathcal{G}^{t-1})(\hat{\mathbf{f}}^{t,S}\cdot\hat{\mathbf{w}}^t) \\
&= \gamma[\sum\nolimits_{S\in\mathcal{S}} \frac{1}{\bar{m}}\hat{\mathbf{f}}^{t,S}]\cdot\bar{\mathbf{w}}^t = \frac{\gamma}{\bar{m}}[\sum\nolimits_{S\in\mathcal{S}^{eb}} \hat{\mathbf{r}}^{t,S}]\cdot\bar{\mathbf{w}}^t \\
&= \mathbf{f}^t\cdot\bar{\mathbf{w}}^t \qquad \text{according to (5.7).}
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}[u^t|\mathcal{G}^{t-1}] &= (1-\gamma)(\mathbf{f}^t\cdot\bar{\mathbf{w}}^t) + \gamma \sum\nolimits_{S\in\mathcal{S}^{eb}} \frac{1}{\bar{m}}(\mathbf{f}^t\cdot\phi(S)) \\
&\geq (1-\gamma)\mathbb{E}[\hat{u}^t|\mathcal{G}^{t-1}] - \gamma. \quad \text{since } \|\mathbf{f}^t\|_1 \leq 1
\end{aligned}$$

$$\begin{aligned}
\mathbb{E}[u^t] &= \mathbb{E}[\mathbb{E}[u^t|\mathcal{G}^{t-1}]] \geq \mathbb{E}[(1-\gamma)\mathbb{E}[\hat{u}^t|\mathcal{G}^{t-1}] - \gamma] \\
&= (1-\gamma)\mathbb{E}[\mathbb{E}[\hat{u}^t|\mathcal{G}^{t-1}]] - \gamma = (1-\gamma)\mathbb{E}[\hat{u}^t] - \gamma.
\end{aligned}$$

Sum it up over $t = 1, ..., T$, and we will get the inequality (5.4). □

Proof of Inequality (5.5). We follow the sketch of proof of (2) in [53]: Let $nonzero(\hat{\mathbf{f}}^{1:T})$ denote the sequence of non-zero flows in $\hat{\mathbf{f}}^{1:T}$ and let ξ denote the length of $nonzero(\hat{\mathbf{f}}^{1:T})$. Let $regret(GEX, \hat{\mathbf{f}}^{1:T})$ denote the regret of GEX in the "imaginary" game.

Case 1 [$\bar{m} \geq 2$]: If $\bar{m} \geq 2$, $\hat{\mathbf{f}}^t$ can be negative, and the approach by [105] [105] has to be adapted to serve as the GEX, which has been illustrated in appendix A of [52]:

$$\mathbb{E}[regret(GEX, \hat{\mathbf{f}}^{1:T})|\xi] \leq \epsilon(4m+2)R^2\xi + 4m/\epsilon, \tag{5.8}$$

where R is an upper bound of $|\hat{\mathbf{f}}^t\cdot\mathbf{w}|$ for $\mathbf{w}\in\phi(\mathcal{S})$:

$$\begin{aligned}
|\hat{\mathbf{f}}^t\cdot\mathbf{w}| &= |(W^\dagger)^{-1}\hat{\mathbf{r}}^t\cdot W\boldsymbol{\alpha}| = |\hat{\mathbf{r}}^t\cdot\boldsymbol{\alpha}| \leq \|\hat{\mathbf{r}}^t\|_1\|\boldsymbol{\alpha}\|_\infty \\
&\leq Cm/\gamma \qquad \text{since } \|\hat{\mathbf{r}}^t\|_1 \leq k\bar{m}/\gamma = m/\gamma,
\end{aligned}$$

where $\mathbf{w} = W\boldsymbol{\alpha}$ with $\|\boldsymbol{\alpha}\|_\infty \leq C$ and C is the spanning ratio of W w.r.t. $\phi(\mathcal{S})$. Substitute to Eq.(5.8) with $\mathbb{E}[\xi] = \gamma T$:

$$\begin{aligned}&\mathbb{E}[regret(GEX, \hat{\mathbf{f}}^{1:T})] = \mathbb{E}[\mathbb{E}[regret(GEX, \hat{\mathbf{f}}^{1:T})|\xi]] \\ &\leq \epsilon(4m+2)C^2m^2T/\gamma + 4m/\epsilon.\end{aligned}$$

Let $\epsilon = \sqrt{\gamma/T}/m$, we have:

$$\mathbb{E}[regret(GEX, \hat{\mathbf{f}}^{1:T})] \leq (8m+2)C\sqrt{T/\gamma}.$$

Case 2 [$\bar{m} = 1$]: In this case, $\hat{\mathbf{f}}^t = \frac{1}{\gamma}\mathbf{f}^t$. The algorithm by [105] [105] can be directly applied for GEX:

$$\begin{aligned}\mathbb{E}[regret(GEX, \hat{\mathbf{f}}^{1:T})|\xi] &\leq \epsilon\xi/\gamma + m/\epsilon \\ \mathbb{E}[regret(GEX, \hat{\mathbf{f}}^{1:T})] &\leq \epsilon T + m/\epsilon.\end{aligned}$$

Set $\epsilon = \sqrt{m/T}$, we get:

$$\mathbb{E}[regret(GEX, \hat{\mathbf{f}}^{1:T})] \leq 2\sqrt{mT}.$$

□

Proof of Inequality (5.6). We follow the sketch of proof of (3) in [53], since $\|\mathbf{w}\|_2 \leq \sqrt{m}$ for $\mathbf{w} \in \phi(\mathcal{S})$:

$$|opt(\hat{\mathbf{f}}^{1:T}) - opt(\mathbf{f}^{1:T})| \leq \sqrt{m}\|\sum\nolimits_{t=1}^{T}(\hat{\mathbf{f}}^t - \mathbf{f}^t)\|_2. \tag{5.9}$$

Define $\Delta^t = \hat{\mathbf{f}}^t - \mathbf{f}^t$, according to [53]:

$$\mathbb{E}[\|\sum\nolimits_{t=1}^{T}\Delta^t\|_2]^2 \leq \sum\nolimits_{t=1}^{T}\mathbb{E}[\|\Delta^t\|_2^2]. \tag{5.10}$$

Case 1 [$\bar{m} \geq 2$]:

$$\|\hat{\mathbf{f}}^t\|_2 = \|(W^\dagger)^{-1}\hat{\mathbf{r}}^t\| \leq \sqrt{m}\|(W^\dagger)^{-1}\|_\infty \|\hat{\mathbf{r}}^t\|_\infty \leq \bar{m}\sqrt{m}\beta_\infty/\gamma$$
$$\|\mathbf{f}^t\|_2\|\mathbf{f}^t\|_1 \leq 1.$$

$$\|\Delta^t\|_2 \leq \|\hat{\mathbf{f}}^t\|_2 + \|\mathbf{f}^t\|_2 \leq \begin{cases} 1, & \text{with prob. } 1-\gamma; \\ \frac{\bar{m}\sqrt{m}\beta_\infty}{\gamma} + 1, & \text{with prob. } \gamma. \end{cases}$$
$$\mathbb{E}[\|\Delta^t\|_2^2] \leq (\bar{m}\sqrt{m}\beta_\infty + 1)^2/\gamma.$$

Substitute to (5.9) & (5.10):

$$\mathbb{E}[|opt(\hat{\mathbf{f}}^{1:T}) - opt(\mathbf{f}^{1:T})|] \leq \bar{m}m(\beta_\infty + 1)\sqrt{T/\gamma}$$

Case 2 [$\bar{m} = 1$]: Similarly, since $\|\hat{\mathbf{f}}^t\|_2 = \|\frac{1}{\gamma}\mathbf{f}^t\|_2 \leq \frac{1}{\gamma}$:

$$\mathbb{E}[|opt(\hat{\mathbf{f}}^{1:T}) - opt(\mathbf{f}^{1:T})|] \leq \sqrt{mT/\gamma}.$$

□

5.4.2 Regret of SBGA with the Optimal Adaptive Policy as Benchmark

Besides the best fixed allocation on hindsight, we also set the optimal *adaptive* strategy as benchmark. Specifically, the defender applying the optimal adaptive strategy plays the optimal allocation against the adversarial flow at each round. Let OPT be the cumulative defender utility of the optimal adaptive strategy. Our next result shows that SBGA achieves a low regret compared with a constant fraction δ of OPT. The intuition is to show that the cumulative utility of the best fixed allocation on hindsight is at least $\delta \cdot OPT$, which is true since the defender with fixed allocation can always operate k checkpoints on those paths with maximum cumulative flows.

Theorem 5.4. *Let* $\tau_{min} = \min_{i\in\mathcal{I}} \tau_i$ *and* $\tau_{max} = \max_{i\in\mathcal{I}} \tau_i$*, we have* $\delta \cdot OPT - Reward(SBGA) \leq \mathcal{O}(T^{2/3})$ *where* $\delta = \min\{1, \frac{k}{m}\}\frac{\tau_{min}}{1-(1-\tau_{max})^k}$.

Proof. Suppose $k \leq m$ and let $\delta = k/m$. Consider the fixed allocation which allocates the k checkpoints on k pathes with maximal cumulative flows in $\sum_{t=1}^{T} \mathbf{f}^t$:

$$\max_{S \in \mathcal{S}} U_d(S, \sum_{t=1}^{T} \mathbf{f}^t) \geq \frac{k}{m} \cdot \tau_{min} \sum_{p \in \mathcal{P}} \sum_{t=1}^{T} f_p^t.$$

For the optimal adaptive defender strategy,

$$\max_{S \in \mathcal{S}} U_d(S, \mathbf{f}^t) \leq [1 - (1 - \tau_{max})^k] \sum_{p \in \mathcal{P}} f_p^t.$$

Therefore, we have:

$$\frac{\max_{S \in \mathcal{S}} U_d(S, \sum_{t=1}^{T} \mathbf{f}^t)}{OPT(F)} \geq \frac{k}{m} \frac{\tau_{min}}{1 - (1 - \tau_{max})^k} = \delta$$

□

5.5 Experimental Evaluation

To demonstrate the practical applicability of our approach, we evaluate its performance through extensive experiments. All computations were performed on a 64-bit PC with 16 GB RAM and a quad-core 3.4 GHz processor. The tested networks are random planar graphs generated by the Waxman geographical model (*WG*) suitable for modeling transportation networks [92]. By default, the instances are parameterized as follows: the number of nodes $|\mathcal{N}| = 200$ and the average degree is 3.0. The number of inspection stations $n = 100$. The number of candidate paths for adversarial flow $m = 20$, and the number of resources $k \in \{10, 20\}$, corresponding with $\bar{m} \in \{1, 2\}$. It is worthwhile to point out that $\bar{m}$ is usually not large (around 2) in practice. Take the illegal drug trafficking scenario as an example. It is reported that 88 percent of all drug shipment is transported to the major drug markets within the United States through *eight* principal corridors [49]. Meanwhile, there are 39 tactical checkpoints owned by the Border Patrol in 2009 and the average operation percentage is about 8% for traffic jam concern [46]. Thus, the settings here are realistic. All inspection stations are randomly placed on the graph and all candidate paths for the adversarial flow are randomly generated. The edge

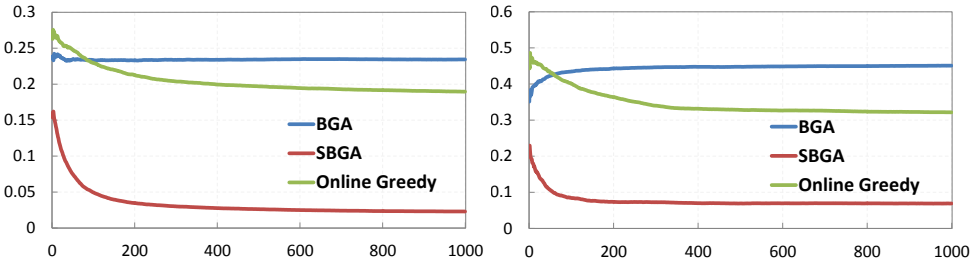
capacity c_e is randomly chosen in $[0.5, 1.0]$, and inspection probability $\tau_i \sim [0.2, 0.6]$. The total number of rounds $T = 1000$. The results are averaged on 100 runs on *one* randomly generated instance. However, we do emphasize that the convergence trend is almost the same across simulated instances except in the initial rounds.

We mainly compare SBGA against two benchmark algorithms: i) *BGA* [52], and ii) *Online Greedy* algorithm for online submodular maximization [51]. The learning parameters in all algorithms are set to their optimal or rough optimal values w.r.t. the theoretical regret bounds. The other candidate benchmarks, including FPL for semi-bandit optimization [99] and its variant FPL-UE [23], and the Geometric Hedge algorithm [106], cannot scale up to instances with 100 checkpoints and over 10 resources. Thus, we only test their performance on small instances with 20 checkpoints and 5 resources.

We test our algorithm against four types of attackers, which together represent the majority of typical attacking models: i) *Uniform*: The attacker with uniform network flow at each round; ii) *BestResponse*: At round t, the attacker best responds to the empirical mixed defender strategy $\mathbf{x}$ of history with $x_S = t_S/(t-1)$ where t_S represents the number of times that defender plays allocation S in first $t-1$ rounds; iii) *Adversarial*: The adversarial type attacker aims at minimizing the defender's utility. In particular, at round t, the attacker plays the network flow which minimizes the defender utility of the best fixed allocation on hindsight, under the restriction that the amount of flow is no smaller than certain threshold, set to be half of the maximal flow; and iv) *QuantalResponse (QR)*: It is non-trivial to apply QR model in network security domain. Thus, we provide a simple implementation where we first uniformly generate a set of 50 network flows $\mathcal{F}^{qr}$. At round t, the attacker responds to the empirical mixed defender strategy $\mathbf{x}$ by playing a probability distribution over $\mathcal{F}^{qr}$ such that a network flow $\mathbf{f} \in \mathcal{F}^{qr}$ is chosen with probability $p_{\mathbf{f}} = \frac{e^{\lambda U_a(\mathbf{x},\mathbf{f})}}{\sum_{\mathbf{f}' \in \mathcal{F}^{qr}} e^{\lambda U_a(\mathbf{x},\mathbf{f}')}}$. The parameter λ controls the rationality level of the attacker. When $\lambda = 0$, the attacker uniformly chooses the network flow in $\mathcal{F}^{qr}$ to play; while when $\lambda = \infty$, the attacker plays the best response flow in $\mathcal{F}^{qr}$.

The submodular maximization problem of GEX subroutine in SBGA (Line 6 of Algorithm 10) and BGA (Line 6 of Algorithm 3) is easily formulated as a convex integer program as program 4.8 in Section 4.4, which can be solved by KNITRO (version 9.0.0)

efficiently in seconds for most instances tested. Observe that the GEX subroutine is the most time-consuming part in SBGA and BGA as the remaining operations at each round are simple matrix multiplications. However, with 100 random runs and 1000 rounds for each run of the algorithm, the total runtime of the experiments could easily exceed the limits. Therefore, we use the greedy algorithm (Algorithm 7 in Section 4.4) to solve the submodular maximization problem of GEX instead, which has been shown to be good enough in the last chapter. To this end, the runtime per round of SBGA and BGA is negligible (less than 0.01 seconds in the tested experiments).



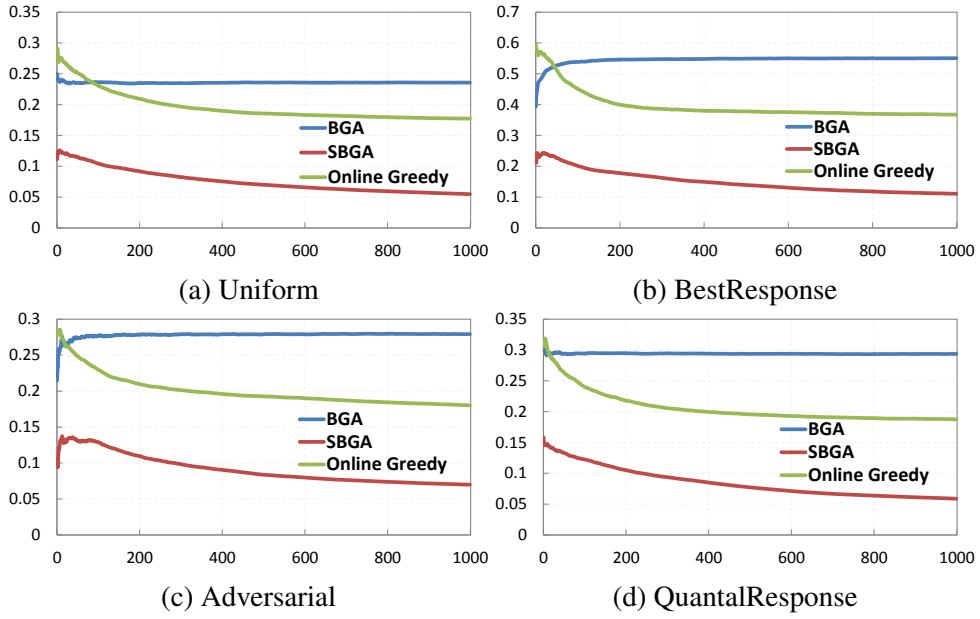
(a) Uniform (b) BestResponse

(c) Adversarial (d) QuantalResponse

FIGURE 5.2: Convergence analysis of the average regret of BGA, online greedy and SBGA ($\bar{m} = 1$).

5.5.1 Solution Quality

The performance of tested algorithms is depicted in Figures 5.2 & 5.3. We can observe that: i) the convergence rate of the average regret for SBGA is extremely fast and solutions with low enough regret (less than 20% of the average reward of the best fixed hindsight allocation) are obtained after about 50 rounds; ii) SBGA outperforms BGA and online greedy algorithm significantly in both convergence rate and average regret per round; iii) the convergence of the average regret for BGA cannot be observed due to the extensive exploration. In fact, according to the optimal learning parameter, for



(a) Uniform (b) BestResponse

(c) Adversarial (d) QuantalResponse

FIGURE 5.3: Convergence analysis of the average regret of BGA, online greedy and SBGA ($\bar{m} = 2$).

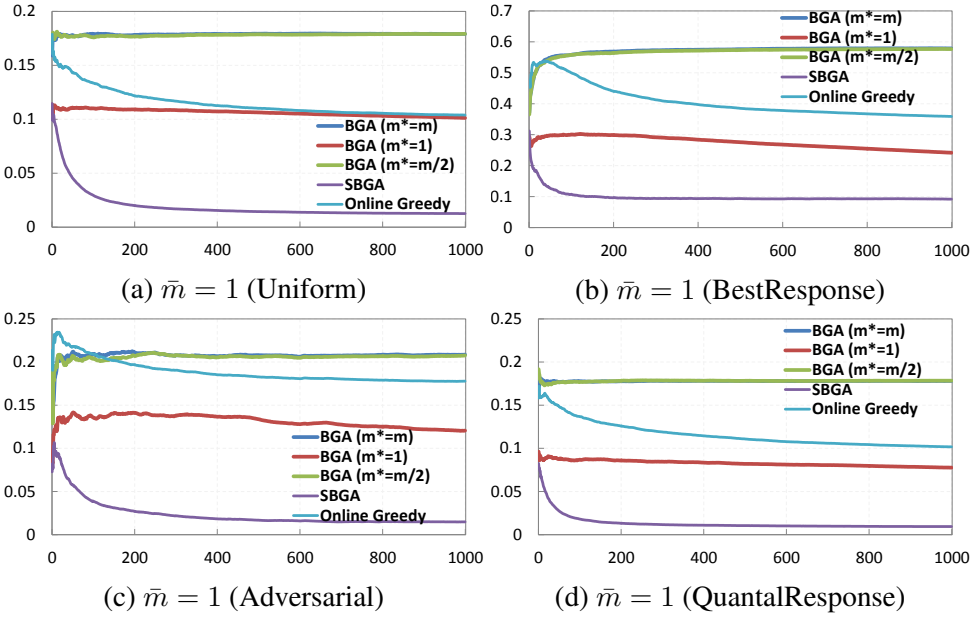
$T = 1000$ and $m = 20$, the exploration probability γ is 1. Furthermore, even if we manually tune the parameter of BGA, it cannot catch up with SBGA (see 5.5.2); and iv) it is obvious to see that the convergence rate of the average regret for online greedy algorithm slows down at a high regret level, which is reasonable since it only achieves low approximate regret.

Regarding the adversarial types, we can observe that the scale of regret against the BestResponse type adversary is significantly larger than others, which is reasonable since the BestResponse adversary will maximize the overall successful flows, and the corresponding network flow should have a larger amount. Even though, the average regret of SBGA is still low enough (around 0.1) for practical use. All these results support out intuition of exploiting the semi-bandit feedback and show that SBGA achieves low regret solutions with fast convergence rate against various realistic adaptive adversaries, whose practical applicability is acknowledged.

5.5.2 BGA with Manually Tuned Parameters

In previous evaluation, the learning parameters are set to the values that achieve the optimal theoretical bound, which, in practice, may not perform well. For example,

consider the game instances with $T = 1000$ and $m = 10$. In order to achieve the optimal theoretical bound of BGA, $\gamma = mT^{-1/3}$ will be set to 1. That is, BGA shows over-exploration. The reason for such performance is that the learning parameter declared in the theoretical analysis is optimal only for the theoretical upper bound, while may not be the best selection for actual performance. Thus, we also compare SBGA and BGA with manually tuned parameter, in particular γ which controls the balance of exploration and exploitation, on game instances generated with $n = 100$ and $k = m = 20$. Figure 5.4 illustrates the experimental results. The learning parameter of BGA is tuned that $\gamma = m^*T^{-1/3}$ with $m^* \in \{1, m/2, m\}$. BGA with $m^* = 1$ shows the best performance among the three BGA installations. However, it is still outperformed by SBGA significantly.



(a) $\bar{m} = 1$ (Uniform) (b) $\bar{m} = 1$ (BestResponse)

(c) $\bar{m} = 1$ (Adversarial) (d) $\bar{m} = 1$ (QuantalResponse)

FIGURE 5.4: Comparion of SBGA and BGA with manually tuned parameters.

5.5.3 Robustness

Since SBGA requires a good estimation of the interdiction probability on each checkpoint, we evaluate the robustness of our algorithm under the uncertainty of the estimation of interdiction probability. In particular, let $\tilde{\tau}_i$ denote the defender's estimation of the interdiction probability of checkpoint i, which is generated in such way: $\tilde{\tau}_i \sim \tau_i \cdot [1 - \delta, 1 + \delta]$ where τ_i is the true interdiction probability and δ measures

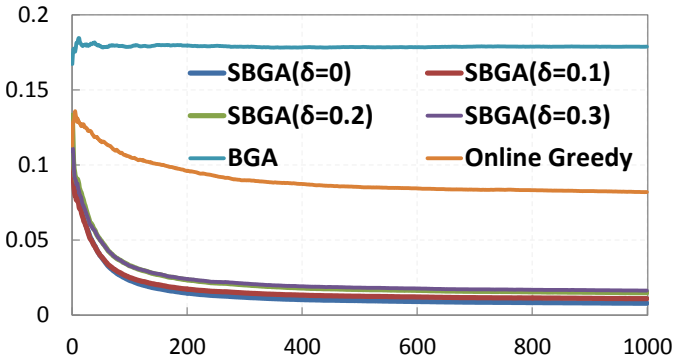
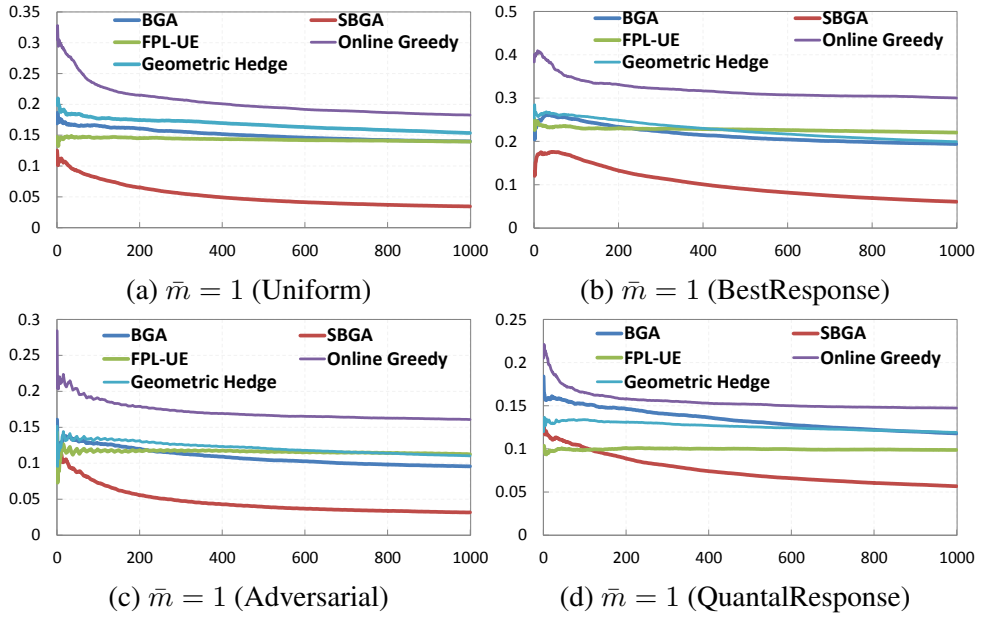


FIGURE 5.5: Robustness analysis of SBGA

the defender's uncertainty on the estimation. We measure the performance of SBGA in games with $n = 100$, $k = m = 20$, and varying values of $\delta \in \{0, 0.1, 0.2, 0.3\}$, compared with BGA and online greedy algorithm, and the result is shown in Figure 5.5. We can see that with higher uncertainty, the regret is larger, which is reasonable. Even though, the average regret of SBGA still outperforms BGA and online greedy algorithm significantly and its convergence rate did not decrease a lot.

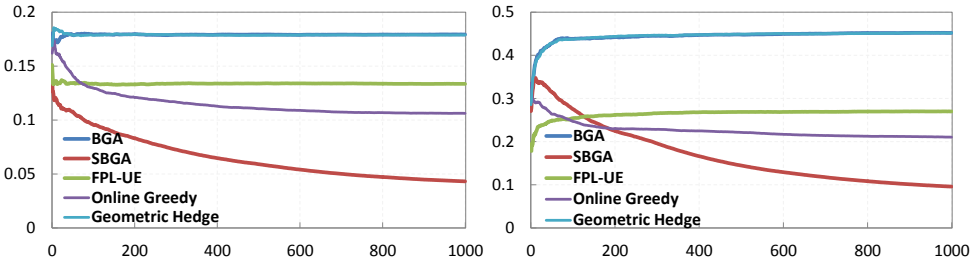
5.5.4 Small-scale Instances



(a) $\bar{m} = 1$ (Uniform) (b) $\bar{m} = 1$ (BestResponse)

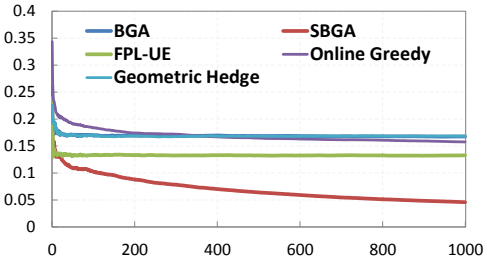
(c) $\bar{m} = 1$ (Adversarial) (d) $\bar{m} = 1$ (QuantalResponse)

FIGURE 5.6: Comparison of SBGA with BGA, Online Greedy, FPL-UE and Geometric Hedge on small-scale instances ($\bar{m} = 1$).

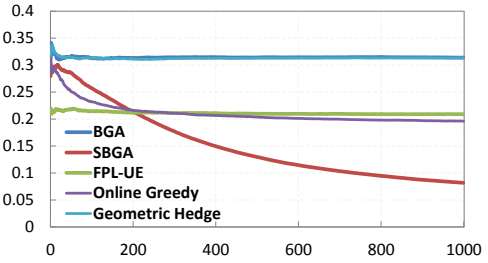


(a) $\bar{m} = 2$ (Uniform)

(b) $\bar{m} = 2$ (BestResponse)



(c) $\bar{m} = 2$ (Adversarial)



(d) $\bar{m} = 2$ (QuantalResponse)

FIGURE 5.7: Comparison of SBGA with BGA, Online Greedy, FPL-UE and Geometric Hedge on small-scale instances ($\bar{m} = 2$).

We test different algorithms including FPL-UE and Geometric Hedge on small-scale instances with $n = 20$, $k = 5$, $m \in \{5, 10\}$ and correspondingly, $\bar{m} \in \{1, 2\}$. FPL-UE is designed for combinatorial adversarial online learning problem with reward maximization objective [23]. Geometric Hedge algorithm achieves $\mathcal{O}(T^{1/2})$ regret against adaptive adversary with high probability [106]. The results are shown in Figures 5.6 & 5.7. Both FPL-UE and Geometric Hedge require the computation on the entire decision set $\phi(\mathcal{S})$. Therefore, the scalability of them is poor and the convergence rate is slow. SBGA outperforms all other algorithms significantly after 200 rounds.

5.6 Game Theoretical Online Promotion of Exergames for the Elderly

The methodologies proposed in this chapter, including game theoretic reasoning and online learning, also have various potential applications. One of them is the propagation of positive influence in the social network of the elderly. We use the popularization of rehabilitation exercises as an example to illustrate our proposed hybrid framework

of influence maximization on the social network of the elderly, which integrates the models and approaches provided by this chapter.

We are living in an aging world. It is estimated that by 2050, the global elderly population is projected to more than double its size in 2015, almost reaching 2.1 billion [107]. The elderly will face various health related issues when ageing, which will weaken their physical and cognitive abilities.

Rehabilitation exercises are critical for the elderly to recover their physical capabilities and to be able to live independently in order to decrease the social home's burden [108]. The elderly are usually advised to take the exercises frequently in order to achieve the goal of physical recovery. However, it is a critical issue to keep enough incentives of the elderly to take the exercises by themselves. Traditional rehabilitation exercises are often guided in the one-to-one fashion, which makes the treatments hard to implement due to the high cost and dependency on human resources, and the time consumed to go to the clinics. Under these restrictions, the adherence rate of rehabilitation exercises is quite low in practice. It is pointed out that only 31% elderly follow the prescribed exercises, which makes the treatments less effective [109].

Recently, new technologies such as video games, augmented reality and virtual reality have drawn extensive attentions in the rehabilitation treatment [110]. These advances in technology can efficiently improve the rehabilitation treatment since they can decrease the monotony of the repeated exercises and provide instant feedback of elderly's motions [111]. Video games with motion detection devices, such as Microsoft Kinect, can stimulate the elderly to take exercises by attractive interfaces and interesting tasks (Figure 5.8). Therefore, the elderly can complete rehabilitation treatment in the gaming environment with enjoyment. Exergames can be played at home and many other places, such as nursing house and community centers, and the elderly can take the treatment anytime they want, which is much more convenient compared with the traditional rehabilitation exercises.

However, despite these advantages of exergames, the incentive for the elderly to play them is still a major issue which is neglected by the rehabilitation exergames design. The reason that exergames are not popular in aging population is the lack of

engagement [112]. The elderly are usually unfamiliar with the new technology, such as exergames, and they may even feel anxious or hostile when playing exergames [112]. Therefore, extra work has been conducted to encourage and teach the elderly to play the exergames. For example, in the past few years, The Joint NTU-UBC Research Centre of Excellence in Active Living for the Elderly (LILY) has participated into various social activities in Singapore, such as the countdown in New year's day, in order to promote their novel exergames to the elderly, and the feedbacks are positive.

Unfortunately, it is impossible for the government or the institute of elderly caring to promote these exergames to every senior citizen or to carry out such promotion at every community. In this case, how to popularize exergames under budget constraints to achieve the maximal influence of rehabilitation exercises is an extremely challenging task. Though the problem is easily recognized to be closely related to the well known *influence maximization* topic, several unique properties of the exergame popularization issue call for specifically tailored models and methodologies, rather than existing ones in influence maximization literature.

First, in most influence propagation models, their is a real-valued weight in the range $[0, 1]$ associated with each edge (i, j) between two individuals i and j, which represents the probability that an active individual i successfully activates the neighbor node j. This simple probabilistic model is suitable for many promotions in viral markets, such as App advertisement through social networks. However, the dynamics of social innovation is certainly more complex than the linear independent model [113]. Second, due to the shortage of existing data and domain knowledge, it is critical to learn the promotion dynamics online.

To address these challenges, we provide an online promotion framework of the exergames for the elderly with following contributions: i) we adopt the game theoretical dynamics of the exergame promotion, which is inspired by the dynamics of social innovation [113]; ii) we formulate an online promotion problem where the decision-maker is facing an unknown promotion dynamics; and iii) we provide a Thompson sampling algorithm to resolve the exploration versus exploitation in the online promotion. Extensive experimental evaluation is conducted.

FIGURE 5.8: Kinect basketball exergame.

5.6.1 Related Work

Recently, novel technologies in virtual reality, augmented reality and motion detection make it possible of playing exergames for the purpose of rehabilitation. These exergames provide a relaxing and entertaining environment for the elderly to take the treatment instead of feeling monotony and isolation in traditional exercises. We review some well established exergames in Section 5.6.1.1. In Section 5.6.1.2, we review some research on the influence maximization in the social network.

5.6.1.1 Exergames for Elderly

Elderly may suffer from various physical issues when ageing, such as the reduction of the motor and cognitive capabilities. It is shown that repetitive exercises can provide a sufficient amount of stimulation on elderly's brain and hence recover the control on one's motion capability [114]. Other studies on exergames indicate that new technologies, such as video games, can increase the elderly's motivation of repetitive exercises and enhance the efficiency of recovery of motor and cognitive capability [110, 115]. Exergames can provide attractive graphical virtual environments, which can be designed in multiple ways to help the elderly be more engaged, immersed and motivated in the rehabilitation treatment [116].

Exergames are usually implemented on the wirelessly controllable platforms, including Nintendo Wiimote and Microsoft Kinect, which are popular due to their non-intrusive and control-free properties. These wireless devices require less human intervention and make the elderly feel more comfortable in an unrestricted environment. Therefore, they can make the elderly feel independent and self-determinate, and the quality of rehabilitation treatment is improved [117].

Due to the unfamiliarity of the exergames to the elderly, some works focus on incentivizing the elderly to play the exergames. Various key factors to the exergames design are pointed out, including the visibility, feedback and identification of the target users [110]. Motition-based games are proposed which combines the motion detection with entertainment in games [111].

5.6.1.2 Influence Maximization

In an informal definition, the influence maximization aims at finding a seed set of nodes which can maximize the influence propagation in networks. The influence maximization is conducted in the social network, which can either be online or bonded by friendship and relationship. The problem has drawn extensive attentions from academy and industry who wants to promote their products, services and innovative ideas through the powerful word-of-mouth effect. Recently, some applications based on influence maximization algorithms have been deployed in the real world to recommend intervention plan for use by homeless shelters, with the goal of raising awareness about HIV within homeless youth [118], which supports the potential of influence maximization to solve the realistic problems.

Influence maximization was first studied by Domingos and Richardson [119, 120] as an algorithmic problem, where they provide the probabilistic methods. Kempe, Kleinberg, and Tardos [121] model the influence maximization problem as a discrete optimization problem. In their model, a social network is formulated as a graph where the nodes represent the individuals and edges denote the connections and relationship

between pairs of individuals. The propagation of influence within the network is captured by a stochastic cascade model. The influence maximization problem is defined as follows:

Definition 5.5. *Given a social network G, an influence cascade model and a fixed number k, find a set of k nodes in G (*seeds*) such that following the designated influence cascade model, the expected number of nodes which are influenced by the seeds is maximized.*

It is proved that the optimization problem associated with influence maximization is NP-hard [121], and a greedy based approximation algorithm is available to guarantee that the objective is within $(1 - 1/e)$ of the optimum.

5.6.2 Game Theoretical Online Promotion

Singapore has over 100 community centers (Fig. 5.9), which play an important in active aging and are ideal for promoting the exergames. Different community centers are not separated, instead, they affect each other. For example, the citizens visiting neighborhood community centers are more likely to be friends and a community center with exergames promotion could spread the positive attitude towards the exergames to its neighborhoods. We capture the dynamics of influence propagation with the coordination game which is suitable to model human's decision making in social life [113]. We then provide an online promotion framework where the decision-maker sequentially selects subsets of community centers of size k to conduct exergames promotion and aims to maximize the expected number of community centers with positive attitudes toward exergames.

(a) Clementi community center (b) Singapore zones map

FIGURE 5.9: Community center and zones in Singapore.

5.6.2.1 Coordination Games

	P	N
P	$1+\alpha, 1+\alpha$	$0, 0$
N	$0, 0$	$1, 1$

Let $G = (V, E)$ be a directed graph with node set V representing all community centers and edge set E. The number of nodes is denoted by n. Each edge (i, j) is associated with a value w_{ij}, which can be interpreted as the impact that i has on j. Let N_i denote the set of neighbors of i. That is, $N_i = \{j \in V | (j, i) \in E\}$. As one can imagine, w_{ij} is higher for a pair of geographically close community centers i and j. As we can set $w_{ij} = 0$ when $(i, j) \notin E$, the graph G can be completely represented by the node set V and a nonnegative $n \times n$ matrix $W = \langle w_{ij} \rangle$ where $w_{ii} = 0$ for all $i \in V$.

Assume each community center has two attitudes toward the exergames, P for "positive" and N for "negative". Our goal is to make more nodes select P. The state of the process at time t is a vector $\mathbf{x}^t \in \{P, N\}^n$ where x_i^t is the choice of node i. Each node plays the coordination game with its neighbors, displayed in the above table where $\alpha > 0$ is the benefit added by P. The utility of i is

$$U_i(\mathbf{x}^t) = \sum_{j \in N_i} w_{ji} \cdot u_i(x_i^t, x_j^t) \tag{5.11}$$

At each time t, we assume that each agent will update his choice following the log-linear rule [113] where the agent will choose P at time $t+1$ with probability ($\beta \geq 0$):

$$\frac{e^{\beta U_i(P, \mathbf{x}_{-i}^t)}}{e^{\beta U_i(P, \mathbf{x}_{-i}^t)} + e^{\beta U_i(N, \mathbf{x}_{-i}^t)}} \tag{5.12}$$

Our goal is to pick a set of K nodes $S \subseteq V$ to maximize the expected number of nodes choosing P when t is large enough. However, in practice, we have no information regarding the influence propagation dynamics on the network, including W, α and β. Thus, we formulate an online promotion problem to learn the influence propagation dynamics.

5.6.2.2 Online Promotion

Online Promotion

input: $G = (V, E)$, K

for $t = 1, 2, ...$

- select a set of K nodes $S \subseteq V$ based on $\mathbf{x}^{t-1}$
- set $y_i^{t-1} = P$ for all $i \in S$ and $y_i^{t-1} = x_i^{t-1}$ otherwise
- observe $\mathbf{x}^t$
- receive an unknown reward $r(\mathbf{x}^{t-1}, \mathbf{y}^{t-1})$

The reward $r(\mathbf{x}^{t-1}, \mathbf{y}^{t-1})$ is defined as $\sigma(\mathbf{y}^{t-1}) - \sigma(\mathbf{x}^{t-1})$ where

$$\sigma(\mathbf{x}) = \lim_{t \to \infty} \mathbb{E}[\sum_{i \in V} \mathbb{I}_{x_i^t = P} | \mathbf{x}^0 = \mathbf{x}] \tag{5.13}$$

In other words, $\sigma(\mathbf{x})$ denotes the expected number of influenced nodes when $t \to \infty$ starting from initial state $\mathbf{x}$. $r(\mathbf{x}^{t-1}, \mathbf{y}^{t-1})$ can be interpreted as the marginal improvement of influence when promote exergames on S. The goal of the decision-maker is to maximize the cumulative reward during play. The difficulty here is that, the reward $r(\mathbf{x}^{t-1}, \mathbf{y}^{t-1})$, depending on the influence propagation dynamics, is unknown to the decision-maker, thus, the observed information of states are critical to estimate the true parameters of the dynamic.

5.6.3 Thompson Sampling

We adopt the Thompson sampling method to solve the above online promotion problem, which has drawn increasing attention in solving contextual bandit problems [122]. The idea is as follows. Let $\mathbf{\Theta} = \{W, \alpha, \beta\}$ be the parameter of the model, and let $p(\mathbf{\Theta})$ be the prior distribution of $\mathbf{\Theta}$. At each time step t, we sample a parameter $\mathbf{\Theta}$ from the posterior distribution $p(\mathbf{\Theta}|\mathcal{H}^t)$ where $\mathcal{H}^t$ denotes the history of observed states in the past $t-1$ time steps, i.e., $\mathcal{H}^t = \{\mathbf{x}^0, \mathbf{y}^0, ..., \mathbf{x}^{t-1}\}$. After $\mathbf{\Theta}$ is sampled, we call an oracle $\mathcal{O}(\mathbf{\Theta}, \mathbf{x}^{t-1})$ to solve for the set of K nodes which bring the maximal marginal

improvement. In other words, the oracle solves the following optimization problem.

$$\max_{S} \quad \sigma(\mathbf{y}) - \sigma(\mathbf{x}) \tag{5.14a}$$

$$\text{s.t.} \quad y_i^{t-1} = P \forall i \in S \tag{5.14b}$$

$$y_i^{t-1} = x_i \forall i \notin S \tag{5.14c}$$

$$|S| \leq K. \tag{5.14d}$$

5.6.4 Preliminary Experimental Evaluation

We conduct experiments on a simpler model of the influence propagation, independent cascade model, which is widely adopted in influence maximization literature. We leave the implementation of Thompson sampling for game theoretical online promotion in future work.

5.6.4.1 Independent Cascade Model

The random process of influence propagation follows *independent cascade* (IC) model [123, 124]. In IC model, each edge $(i, j) \in E$ is associated with two real values $p_{i,j} \in [0, 1]$ where $p_{i,j}$ represents the probability of j being influenced by i if i is already influenced and has positive attitude to exergames. Starting with the initial set of influenced nodes S, the influence spreads in discrete steps randomly as follows. When node i is first influenced at step t, it has a single chance to affect each currently uninfluenced neighborhood j with a successful probability $p_{i,j}$. If i succeeds to affect j, then j will be influenced in step $t + 1$; but whether or not i succeeds, it cannot make any further attempts to affect j in subsequent rounds. The process runs until no more influence are possible.

Kempe *et al.* show that the above influence maximization problem is NP-hard due to a reduction from Hitting Set problem [121]. Despite the negative result of NP-hardness, it is shown that the influence function $\sigma(S)$ is submodular [121]. In other words, the marginal gain of $\sigma(S)$ from adding an element to a set S is at least as high as the

Algorithm 12: Greedy Hill-Climbing Algorithm

```
Initialize S = ∅;
while |S| < k do
    i* = arg max_{i∈V} σ(S ∪ {i}) − σ(S);
    S = S ∪ {i*};
return S.
```

marginal gain from adding the same element to a superset of S. Formally, $\sigma(S)$ satisfies:

$$\sigma(S \cup \{i\}) - \sigma(S) \geq \sigma(T \cup \{i\}) - \sigma(T),$$

for any individual i and any pair of sets $S \subseteq T$. It is well known that the submodular maximization problem admits an approximation based on greedy search which can solve for an approximated optimum to within a factor of $(1-1/e)$. The greedy algorithm starts with the empty set, and adds an element which brings the maximum marginal improvement repeatedly until k elements are added.

Theorem 5.6. *[125, 126]* For a monotone and non-negative submodular function σ, let S be the set obtained by the above greedy algorithm which chooses an element bringing the maximum marginal improvement repeatedly until $|S| = k$. Let S^* be a set which maximizes the value of σ over all k-element sets. Then $\sigma(S) \geq (1 - 1/e) \cdot \sigma(S^*)$.

5.6.4.2 Methodology for IC Model

We provide the algorithms based on Theorem 5.6. Algorithm 12 illustrates the greedy hill-climbing method as we explained before. However, algorithm 12 cannot be directly implemented as we may not have close form representation of σ. In other words, given a set S, it is not trivial to evaluate the underlying function exactly.

Fortunately, the simulation of the influence propagation is easy to conduct and we can get arbitrarily close approximations to $\sigma(S)$ with high probability given enough sampling of the influenced sets from simulation. Furthermore, Kempe *et al.* [121] extend the result of Nemhauser *et al.* [126] and show that for any $\epsilon > 0$, there exists a $\gamma > 0$ such that we can ensure a $(1 - 1/e - \epsilon)$-approximation of σ with Algorithm 12 by evaluation the value of σ with its $(1 + \gamma)$-approximation. Algorithm 13 simulates

Algorithm 13: Monte-Carlo Simulation of σ

Parameter: M sampling trails, a random generator $rand$ which outputs a random real number in $[0, 1]$, the set S to be evaluated, T steps of influence propagation;
Initialize $\bar{\sigma} = 0$, $NumOfTrails = 0$;
while $NumOfTrails < M$ **do**
    $NumOfTrails = NumOfTrails + 1$;
    $\phi(S) = S$;
    $S^0 = S$;
    $S^1 = ... = S^T = \emptyset$;
    for $t = 0$ *to* $T - 1$ **do**
        for $i \in S^t$ **do**
            for $j \in Neighbor(i)$ **do**
                if $j \notin \phi(S)$ & $rand < p_{i,j}$ **then**
                    $S^{t+1} = S^{t+1} \cup \{j\}$;
                    $\phi(S) = \phi(S) \cup \{j\}$;
    $\bar{\sigma} = \bar{\sigma} + \frac{|\phi(S)|}{M}$;
return $\sigma(S) = \bar{\sigma}$;

the value of $\sigma(S)$ with Monte-Carlo sampling, by averaging M sampled sets of $\phi(S)$. When $M \to \infty$, the output value $\bar{\sigma}$ approaches the exact value of $\sigma(S)$ with probability equals to 1. M needs to be tuned to balance between the accuracy and the computational cost.

5.6.4.3 Experimental Evaluation

Experiments were conducted to support our IC approach to achieve maximum influence in an elderly social network. The programs for simulation were written in Python 2.7 and were performed on a 64-bit PC with 16 GB RAM and Intel(R) Core(TM) i7-4770 CPU @ 3.40GHz. The operating system is Linux Ubuntu 17.04. All simulated social networks were generated by a random undirected graph generator in a Python package called SNAP developed by researchers in Stanford University. By specifying the number of nodes and number of edges, the graph can be randomly created. And by specifying a fixed random number in the generator, the network can be ensured to be the same in each experiment setting.In each experiment, we fixed the seed set size as 5, and the propagation step as 3.

We conducted the experiments to evaluate i) the runtime performance and number of influenced nodes under different scales of the network; ii) the runtime performance and number of influenced nodes under different Monte-Carlo sampling size M and iii) the number of influenced nodes compared with a heuristic approach.

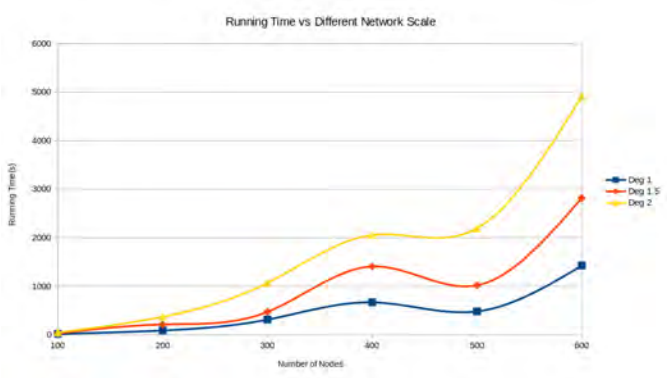


FIGURE 5.10: Running time vs different scale of network.

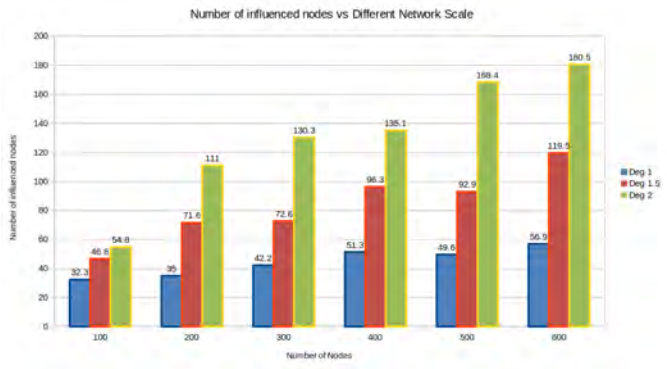


FIGURE 5.11: Number of influenced nodes vs different scale of network.

Scalability Analysis: By fixing the Monte-Carlo sampling size M as 10, we changed the scalability of the network by changing nodes from 100 to 600 with step 100, each group with 3 cases of average degrees, 1,1.5 and 2, for each node to evaluate the running time and the number of influenced nodes. The running time was in the unit of seconds. From Fig 3, we can see that there is an inconsistency at nodes 500, and this is probably due to twice experiment conduction. However, the running time is approximately having a quadratic relationship with the number of nodes(i.e. the scale).While from Fig 4, the number of influenced nodes is increasing if average of degrees of each node increase and if the nodes increases.

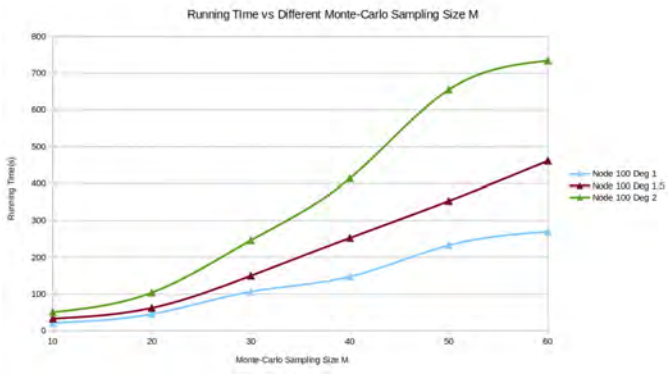


FIGURE 5.12: Running time vs different Monte-Carlo sampling size.

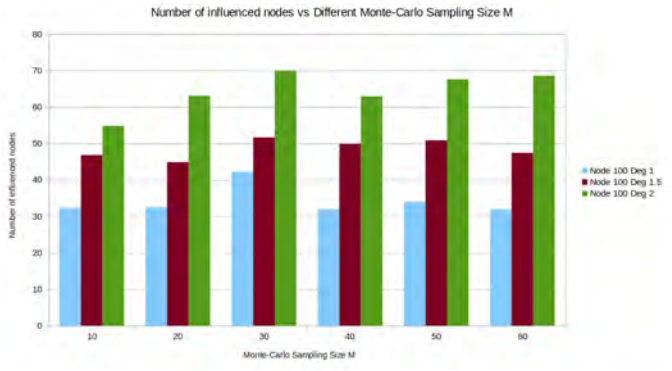


FIGURE 5.13: Number of influenced nodes vs different Monte-Carlo sampling size.

Sampling Size Analysis: Under different Monte-Carlo sampling sizes M, we select nodes 100 as basis with average degree of 1,1.5 and 2 to evaluate the running time and the number of influenced nodes. The running time was in the unit of seconds. From Fig 5, we can see that the running time is approximately having a linear relationship with the number of nodes(i.e. the scale).While from Fig 6, the number of influences keeps approximately as a constant with increasing M.

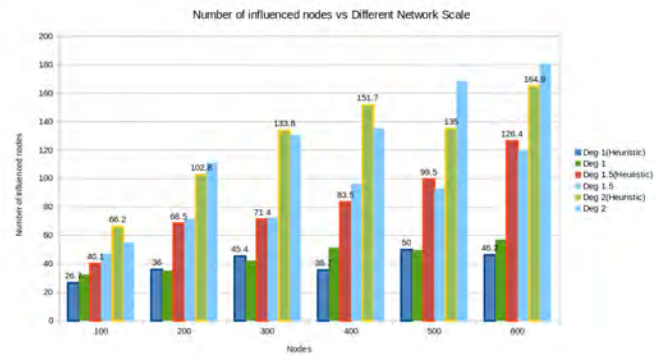


FIGURE 5.14: Comparison of number of influenced nodes vs different scale of network.

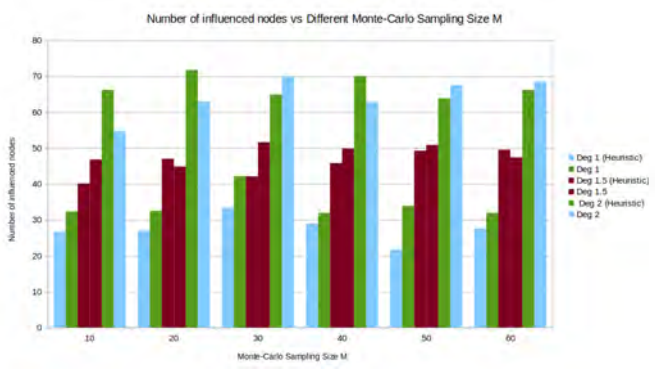


FIGURE 5.15: Comparison of number of influenced nodes vs different Monte-Carlo sampling size.

Comparison with Heuristic Approach: The heuristic approach we used was to specify each node with a heuristic , which is the sum of its connected edges probabilities (edge weight).Then we selected the seed set based on that and to evaluate the number of influenced nodes based on the seed set. A comparison was made to our approach and limitations were found out and given in the later part of the paper. From Fig 7, with node 100 as basis, our approach is slightly better than the heuristic approach (10 out of 18 cases). And from Fig 8, with increasing Monte-Carlo sampling size, our approach is much better than the heuristic approach(13 out of 18 cases).

5.7 Chapter Summary

This chapter provides the first defender strategy in repeated network security domains with no prior knowledge of the adversarial behavior model and the environment. In particular, we proposed an adversarial online learning approach and non-trivially modeled the repeated network interdiction game as an online linear optimization problem, for which we provided a novel online learning algorithm, SBGA, to exploit the unique semi-bandit feedback in network interdiction domains. We formally proved that SGBA achieves low regret bounds compared both with best fixed strategy on hindsight, and the near optimal adaptive strategy. We have also run extensive experiments to show that SBGA efficiently obtains robust solutions with fast convergence rate against various realistic adversarial types. In addition, it significantly outperforms the existing methods. These imply the usefulness of our algorithm in many practical scenarios. We finally

show that the game theoretical reasoning and online learning paradigm can be utilized to solve the promotion of exergames for the elderly.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

In the past decade, the security game research, as a creative application of game theory in security domains, pioneers a vast number of applications and systems which have been successfully deployed in the real world, and make significantly positive influence in several critical domains. Along with the success and progress of the security game models, several critical factors have been revealed, such as the existence of multiple adversaries, the human behavior modeling, and planning and learning in repeated games. The network-structured security games, with the capability to model many realistic and important security domains, have long been investigated in a narrow scope, while neglect the forementioned issues. This thesis, extends the existing network security games research, and provides the first models and algorithms for several fundamental problems.

Motivated by realistic scenarios, the models and algorithms in this thesis stand in high levels and generalize various real-world problems. The network restricted cooperation captures not only the interaction among terrorist groups, but individuals in certain organization also. Many adversarial measures on the network are in the form of a flow, including the illegal drug smuggling, the disease infection, and so on. The semi-bandit

type of feedback commonly exists, and furthermore, the philosophy of exploiting all pieces of information also provides inspiration in application of repeated games for other domains.

The first contribution of this thesis is a comprehensive solution for combating the cooperative adversaries on a communication network. We provide the first incorporation between Stackelberg security games and the cooperative game theory. The proposed branch and price algorithmic framework effectively mitigates the computational complexity revealed by our theoretical analysis which proves the non-existence of the polynomial-time approximation scheme for the defender's decision-making problem. Our framework involves several novel approaches, including a lossless linear relaxation, interior point stabilization, and constant factor approximation, to achieve a scalable solution for realistic-sized instances.

The second contribution is a thorough investigation of the network flow interdiction problem. The key formulation is a linear program, which involves exponentially large numbers of variables and constraints. While existing approaches fail to provide scalable and optimal solutions, we provide a CCG (Column and Constraint Generation) framework, which generalizes the double oracle algorithm from mixed strategies of players to strategies in explicitly represented convex hulls. The CCG algorithm is ensured to converge to the optimal solution according to the weak version of Sion's minimax theorem. The computational complexity of the column generation sub-problem is presented, which is shown in the class of NP-hard problems. We provide a polynomial-time dynamic programming algorithm for the column generation sub-problem with tree-like network structures. Convex integer programs are proposed for both the column generation and constraint generation sub-problems, as well as efficient and competitive approximation methods. The mixture of various novel approaches inside the CCG framework can solve large-scale instances with realistic network structures, as shown with extensive experimental evaluation.

The final contribution is the first low-regret and effective online policy for the defender in repeated network interdiction games. We are the first to investigate the repeated interaction between the defender and attacker in network security games. We apply the online learning methodology, and introduce several key components to achieving

the low-regret solution. We first provide a novel transformation of the straightforward online submodular maximization problem into an online linear optimization problem. While both state-of-the-art algorithms for online submodular maximization and online linear optimization fail in convergence rate of the average regret or providing sublinear regret, we exploit the semi-bandit information in the network interdiction domain and propose the SBGA algorithm, which not only enjoys nice theoretical property of $\mathcal{O}(T^{2/3})$ regret, and shows a significantly fast convergence rate of the average regret compared with the benchmark approaches.

6.2 Future Directions

This thesis provides the first models and algorithms for several network-structured security domains. There are several extensions and applications to further enrich the network security games research. The first extension is to dig into the human behavior models to improve the effectiveness of the defense. Notice that the online learning approaches mostly focus on the worst case regret, which, although are robust enough, may be too conservative. We will incorporate novel human behavior models to enhance our prior knowledge of the adversary, and provide better solutions. The challenge is, we need to understand what is the proper form of the adversarial behavior model and how to infer the adversarial behavior model with incomplete feedback.

The second extension is the combination of the various factors such as the existence of multiple cooperative adversaries with the repeated game modeling. Though we propose to model the cooperation among multiple adversaries with coalitional games, the perfect rationality is hard to be satisfied and the network interdiction problems naturally involve repeated interactions. One possible direction is to utilize the existing models on the dynamic of the terrorist network, such as how the network is recovered. Based on such domain knowledge, we can build a comprehensive model and consider the repeated network blocking against adversary with certain behavior models.

Bibliography

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

[2] James Pita, Manish Jain, Janusz Marecki, Fernando Ordóñez, Christopher Portway, Milind Tambe, Craig Western, Praveen Paruchuri, and Sarit Kraus. Deployed ARMOR protection: the application of a game theoretic model for security at the los angeles international airport. In *7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'08)*, pages 125–132, 2008.

[3] Jason Tsai, Christopher Kiekintveld, Fernando Ordonez, Milind Tambe, and Shyamsunder Rathi. IRIS-A tool for strategic security allocation in transportation networks. In *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems Industry Track (AAMAS'09)*, pages 37–44, 2009.

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

[5] James Pita, Milind Tambe, Christopher Kiekintveld, Shane Cullen, and Erin Steigerwald. GUARDS: game theoretic security allocation on a national scale. In *10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'11)*, pages 37–44, 2011.

[6] Eric Anyung Shieh, Bo An, Rong Yang, Milind Tambe, Craig Baldwin, Joseph DiRenzo, Ben Maule, and Garrett Meyer. PROTECT: an application of computational game theory for the security of the ports of the united states. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI'10)*, pages 2173–2179, 2012.

[7] Albert Xin Jiang, Zhengyu Yin, Chao Zhang, Milind Tambe, and Sarit Kraus. Game-theoretic randomization for security patrolling with dynamic execution uncertainty. In *International conference on Autonomous Agents and Multi-Agent Systems (AAMAS'13)*, pages 207–214, 2013.

[8] Rong Yang, Benjamin J. Ford, Milind Tambe, and Andrew Lemieux. Adaptive resource allocation for wildlife protection against illegal poachers. In *Proceedings of the 13th International conference on Autonomous Agents and Multi-Agent Systems (AAMAS'14)*, pages 453–460, 2014.

[9] Praveen Paruchuri, Jonathan P. Pearce, Milind Tambe, Fernando Ordóñez, and Sarit Kraus. An efficient heuristic approach for security against multiple adversaries. In *6th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'07)*, page 181, 2007.

[10] Christopher Kiekintveld, Manish Jain, Jason Tsai, James Pita, Fernando Ordóñez, and Milind Tambe. Computing optimal randomized resource allocations for massive security games. In *8th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'09)*, pages 689–696, 2009.

[11] Manish Jain, Erim Kardes, Christopher Kiekintveld, Fernando Ordóñez, and Milind Tambe. Security games with arbitrary schedules: A branch and price approach. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, (AAAI'10)*, 2010.

[12] Daniel Kahneman. Prospect theory: An analysis of decisions under risk. *Econometrica*, 47(2):263–292, 1979.

[13] Richard D McKelvey and Thomas R Palfrey. Quantal response equilibria for normal form games. *Games and economic behavior*, 10(1):6–38, 1995.

[14] James Pita, Manish Jain, Fernando Ordóñez, Milind Tambe, Sarit Kraus, and Reuma Magori-Cohen. Effective solutions for real-world stackelberg games: when agents must deal with human uncertainties. In *8th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'09)*, pages 369–376, 2009.

[15] Thanh Hong Nguyen, Amulya Yadav, Bo An, Milind Tambe, and Craig Boutilier. Regret-based optimization and preference elicitation for stackelberg security games with uncertainty. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence (AAAI'14)*, pages 756–762, 2014.

[16] Rong Yang, Christopher Kiekintveld, Fernando Ordóñez, Milind Tambe, and Richard John. Improving resource allocation strategy against human adversaries in security games. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI'11)*, pages 458–464, 2011.

[17] Rong Yang, Fernando Ordóñez, and Milind Tambe. Computing optimal strategy against quantal response in security games. In *International Conference on Autonomous Agents and Multiagent Systems, (AAMAS'12)*, pages 847–854, 2012.

[18] Fei Fang, Peter Stone, and Milind Tambe. When security games go green: Designing defender strategies to prevent poaching and illegal fishing. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI'15)*, pages 2589–2595, 2015.

[19] Yundi Qian, William B. Haskell, Albert Xin Jiang, and Milind Tambe. Online planning for optimal protector strategies in resource conservation games. In *International conference on Autonomous Agents and Multi-Agent Systems (AAMAAS'14)*, pages 733–740, 2014.

[20] Janusz Marecki, Gerald Tesauro, and Richard Segal. Playing repeated stackelberg games with unknown opponents. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS'12)*, pages 821–828, 2012.

[21] Joshua Letchford, Vincent Conitzer, and Kamesh Munagala. Learning and approximating the optimal strategy to commit to. In *Algorithmic Game Theory, Second International Symposium (SAGT'09)*, pages 250–262, 2009.

[22] Avrim Blum, Nika Haghtalab, and Ariel D. Procaccia. Learning optimal commitment to overcome insecurity. In *Proceedings of the Twenty-Eighth Annual Conference on Neural Information Processing Systems (NIPS'14)*, pages 1826–1834, 2014.

[23] Haifeng Xu, Long Tran-Thanh, and Nicholas R. Jennings. Playing repeated security games with no prior knowledge. In *Proceedings of the 15th International Conference on Autonomous Agents & Multiagent Systems (AAMAS'16)*, pages 104–112, 2016.

[24] Douglas S Altner, Özlem Ergun, and Nelson A Uhan. The maximum flow network interdiction problem: Valid inequalities, integrality gaps, and approximability. *Operations Research Letters*, 38(1):33–38, 2010.

[25] Michael O Ball, Bruce L Golden, and Rakesh V Vohra. Finding the most vital arcs in a network. *Operations Research Letters*, 8(2):73–76, 1989.

[26] Eitan Israeli and R Kevin Wood. Shortest-path network interdiction. *Networks*, 40(2):97–111, 2002.

[27] Kavindra Malik, AK Mittal, and Santosh K Gupta. The k most vital arcs in the shortest path problem. *Operations Research Letters*, 8(4):223–227, 1989.

[28] R Kevin Wood. Deterministic network interdiction. *Mathematical and Computer Modelling*, 17(2):1–18, 1993.

[29] Manish Jain, Dmytro Korzhyk, Ondrej Vanek, Vincent Conitzer, Michal Pechoucek, and Milind Tambe. A double oracle algorithm for zero-sum security games on graphs. In *Proceedings of the 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'11)*, pages 327–334, 2011.

[30] Zhen Wang, Yue Yin, and Bo An. Computing optimal monitoring strategy for detecting terrorist plots. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI'16)*, pages 637–643, 2016.

[31] Jason Tsai, Zhengyu Yin, Jun-young Kwak, David Kempe, Christopher Kiekintveld, and Milind Tambe. Urban security: Game-theoretic resource allocation in networked domains. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI'10)*, pages 881–886, 2010.

[32] Yue Yin and Bo An. Efficient resource allocation for protecting coral reef ecosystems. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI'16)*, pages 531–537, 2016.

[33] George W Bush. *National strategy for combating terrorism*. Wordclay, 2009.

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

[35] Alexander Nekrassov. Chechen attack: 'terrorists get weapons from abroad, linked to mideast groups'. http://rt.com/op-edge/211767-chechnya-russia-terrorism-attack-mideast/, 2014.

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

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

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

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

[40] Peter 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.

[41] RHA Lindelauf, HJM Hamers, and BGM 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.

[42] Tomasz P Michalak, Talal Rahwan, Nicholas R Jennings, Piotr L Szczepański, Oskar Skibski, Ramasuri Narayanam, and Michael 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'13)*, pages 293–301, 2013.

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

[44] Yevgeniy Vorobeychik, Bo An, Milind Tambe, and Satinder P. Singh. Computing solutions in infinite-horizon discounted adversarial patrolling games. In *Proceedings of the 24th International Conference on Automated Planning and Scheduling (ICAPS'14)*, pages 314–322, 2014.

[45] Joshua Letchford and Vincent Conitzer. Solving security games on graphs via marginal probabilities. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence (AAAI'13)*, pages 591–597, 2013.

[46] GAO. *BORDER PATROL: Checkpoints Contribute to Border Patrols Mission, but More Consistent Data Collection and Performance Measurement Could Improve Effectiveness.* U.S. Government Accountability Office, 2009.

[47] R.L. Church, M.P. Scaparra, and R.S. Middleton. Identifying critical infrastructure: The median and covering facility interdiction problems. *Annals of the Association of American Geographers*, 94(3):491–502, 2004.

[48] Manish Jain, Vincent Conitzer, and Milind Tambe. Security scheduling for real-world networks. In *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'13)*, pages 215–222, 2013.

[49] US DEPARTMENT OF JUSTICE NATIONAL DRUG INTELLIGENCE CTR. National drug threat assessment, 2010.

[50] Debarun Kar, Fei Fang, Francesco Maria Delle Fave, Nicole Sintov, and Milind Tambe. "a game of thrones": When human behavior models compete in repeated Stackelberg security games. In *Proceedings of the 14th International Conference on Autonomous Agents & Multiagent Systems (AAMAS'15)*, pages 1381–1390, 2015.

[51] Matthew J. Streeter and Daniel Golovin. An online algorithm for maximizing submodular functions. In *Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems (NIPS'08)*, pages 1577–1584, 2008.

[52] H. Brendan McMahan and Avrim Blum. Online geometric optimization in the bandit setting against an adaptive adversary. In *Proceedings of the 17th Annual Conference on Learning Theory (COLT'04)*, pages 109–123, 2004.

[53] Varsha Dani and Thomas P Hayes. Robbing the bandit: Less regret in online geometric optimization against an adaptive adversary. In *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'06)*, pages 937–943. Society for Industrial and Applied Mathematics, 2006.

[54] Heinrich von Stackelberg. Marktform und gleichgewicht. *Springer*, 1934.

[55] Bernhard Von Stengel and Shmuel Zamir. Leadership with commitment to mixed strategies. *Technical Report LSE-CDAM-2004-01, CDAM Research Report.*, 2004.

[56] H. Brendan McMahan, Geoffrey J. Gordon, and Avrim Blum. Planning in the presence of cost functions controlled by an adversary. In *Machine Learning, Proceedings of the Twentieth International Conference (ICML'03)*, pages 536–543, 2003.

[57] Maurice Sion. On general minimax theorems. *Pacific Journal of mathematics*, 8 (1):171–176, 1958.

[58] Yoram Bachrach and Jeffrey S Rosenschein. Coalitional skill games. In *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'08)*, pages 1023–1030, 2008.

[59] Vincent Conitzer and Tuomas Sandholm. Computing the optimal strategy to commit to. In *Proceedings 7th ACM Conference on Electronic Commerce (EC'06),*, pages 82–90, 2006.

[60] Bo An, David Kempe, Christopher Kiekintveld, Eric Shieh, Satinder P. Singh, Milind Tambe, and Yevgeniy Vorobeychik. Security games with limited surveillance. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI'12)*, 2012.

[61] Bo An, Matthew Brown, Yevgeniy Vorobeychik, and Milind Tambe. Security games with surveillance cost and optimal timing of attack execution. In *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'13)*, pages 223–230, 2013.

[62] Thanh Hong Nguyen, Rong Yang, Amos Azaria, Sarit Kraus, and Milind Tambe. Analyzing the effectiveness of adversary modeling in security games. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence (AAAI'13)*, 2013.

[63] F Pan, W Charlton, and DP Morton. Stochastic network interdiction of nuclear material smuggling. *Network Interdiction and Stochastic Integer Programming*, pages 1–19, 2001.

[64] F. Pan. *Stochastic network interdiction: models and methods*. PhD thesis, University of Texas, Austin, 2005.

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

[66] Colin L. Powell. Remarks to the United Nations Security Council. http://2001-2009.state.gov/secretary/former/powell/remarks/2003/17300.htm, 2003.

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

[68] John S. Pistole. Identifying, Tracking and Dismantling the Financial Structure of Terrorist Organizations. http://www2.fbi.gov/congress/congress03/pistole092503.htm, 2003.

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

[70] Georgios Chalkiadakis, Edith Elkind, and Michael Wooldridge. Computational aspects of cooperative game theory. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 5(6):1–168, 2011.

[71] Elias Dahlhaus, David S. Johnson, Christos H. Papadimitriou, Paul D. Seymour, and Mihalis Yannakakis. The complexity of multiterminal cuts. *SIAM J. Comput.*, 23(4):864–894, 1994.

[72] Christos Papadimitriou and Mihalis Yannakakis. Optimization, approximation, and complexity classes. In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing (STOC'88)*, pages 229–234, 1988.

[73] Louis-Martin Rousseau, Michel Gendreau, and Dominique Feillet. Interior point stabilization for column generation. *Oper. Res. Lett.*, 35(5):660–668, 2007.

[74] Dimitris Bertsimas and John N Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, 1997.

[75] Siqian Shen, J. Cole Smith, and Roshan Goli. Exact interdiction models and algorithms for disconnecting networks via node deletions. *Discrete Optimization*, 9(3):172–188, 2012.

[76] David Pritchard. An LP with integrality gap 1+ ϵ for multidimensional knapsack. *arXiv preprint arXiv:1005.3324*, 2010.

[77] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

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

[79] Melanie Mitchell. *An Introduction to Genetic Algorithms*. MIT press, 1998.

[80] David E Goldberg, Bradley Korb, and Kalyanmoy Deb. Messy genetic algorithms: Motivation, analysis, and first results. *Complex systems*, 3(5):493–530, 1989.

[81] Yue Yin, Bo An, and Manish Jain. Game-theoretic resource allocation for protecting large public events. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence (AAAI'14)*, pages 826–834, 2014.

[82] Yue Yin, Haifeng Xu, Jiarui Gan, Bo An, and Albert Xin Jiang. Computing optimal mixed strategies for security games with dynamic payoffs. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI'15)*, pages 681–688, 2015.

[83] Jiarui Gan, Bo An, and Yevgeniy Vorobeychik. Security games with protection externalities. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI'15)*, pages 914–920, 2015.

[84] June S. Beittel. *Mexico: Organzied Crime and Drug Trafficking Organizations*. Congressional Research Service, 2015.

[85] Chuck Rosenberg. *2015-National Drug Threat Assessment Summary.* U.S. Department of Justice Drug Enforcement Administration, 2015.

[86] DOJ. *National Drug Threat Assessment 2010.* U.S. Department of Justice National Drug Intelligence Center, 2010.

[87] Robert L Steinrauf. Network interdiction models. Technical report, DTIC Document, 1991.

[88] Kumar C. Kibble. *Law enforcement responses to mexican drug cartels.* US Department of Homeland Security, 2009.

[89] Ondrej Vanek, Zhengyu Yin, Manish Jain, Branislav Bosanský, Milind Tambe, and Michal Pechoucek. Game-theoretic resource allocation for malicious packet detection in computer networks. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'12)*, pages 905–912, 2012.

[90] George L Nemhauser and Leonard A Wolsey. Best algorithms for approximating the maximum of a submodular set function. *Mathematics of operations research*, 3(3):177–188, 1978.

[91] Pierre Hansen and Nenad Mladenović. Variable neighborhood search: Principles and applications. *European journal of operational research*, 130(3):449–467, 2001.

[92] Bernard M Waxman. Routing of multipoint connections. *IEEE Journal on Selected Areas in Communications*, 6(9):1617–1622, 1988.

[93] Michael T Gastner and Mark EJ Newman. The spatial structure of networks. *The European Physical Journal B-Condensed Matter and Complex Systems*, 49 (2):247–252, 2006.

[94] GAO. *BORDER PATROL: Available Data on Interior Checkpoints Suggest Differences in Sector Performance.* U.S. Government Accountability Office, 2005.

[95] Colin Camerer. *Behavioral Game Theory: Experiments in Strategic Interaction*. Princeton University Press, 2003.

[96] Daniel L McFadden. Quantal choice analaysis: A survey. In *Annals of Economic and Social Measurement, Volume 5, number 4*, pages 363–390. 1976.

[97] William B. Haskell, Debarun Kar, Fei Fang, Milind Tambe, Sam Cheung, and Elizabeth Denicola. Robust protection of fisheries with compass. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence (AAAI'14)*, pages 2978–2983, 2014.

[98] Peter L. Bartlett, Varsha Dani, Thomas P. Hayes, Sham Kakade, Alexander Rakhlin, and Ambuj Tewari. High-probability regret bounds for bandit online

linear optimization. In *Proceedings of the 21st Annual Conference on Learning Theory (COLT'08)*, pages 335–342, 2008.

[99] Gergely Neu and Gábor Bartók. Importance weighting without importance weights: An efficient algorithm for combinatorial semi-bandits. *Journal of Machine Learning Research*, 17:154:1–154:21, 2016.

[100] Peter Auer, Nicolo Cesa-Bianchi, Yoav Freund, and Robert E Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1): 48–77, 2002.

[101] Sham M. Kakade, Adam Tauman Kalai, and Katrina Ligett. Playing games with approximation algorithms. *SIAM J. Comput.*, 39(3):1088–1106, 2009.

[102] Baruch Awerbuch and Robert D. Kleinberg. Adaptive routing with end-to-end feedback: Distributed learning and geometric approaches. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC'04)*, pages 45–53, 2004.

[103] Jacob D. Abernethy, Elad Hazan, and Alexander Rakhlin. Competing in the dark: An efficient algorithm for bandit linear optimization. In *Proceedings of the 21st Annual Conference on Learning Theory (COLT'08)*, pages 263–274, 2008.

[104] Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

[105] Adam Kalai and Santosh Vempala. Efficient algorithms for online decision problems. In *Proceedings of the 16th Annual Conference on Computational Learning Theory and 7th Kernel Workshop (COLT'03)*, pages 26–40, 2003.

[106] Varsha Dani, Thomas P. Hayes, and Sham Kakade. The price of bandit information for online optimization. In *Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems (NIPS'07)*, pages 345–352, 2007.

[107] United Nations. World population ageing 2015. *Department of Economic and Social Affairs: Population Division*, 129, 2015.

[108] Di Wang, Budhitama Subagdja, Yilin Kang, Ah-Hwee Tan, and Daqing Zhang. Towards intelligent caring agents for aging-in-place: Issues and challenges. In *Computational Intelligence for Human-like Intelligence (CIHLI), 2014 IEEE Symposium on*, pages 1–8. IEEE, 2014.

[109] Marianne Shaughnessy, Barbara M Resnick, and Richard F Macko. Testing a model of post-stroke exercise behavior. *Rehabilitation nursing*, 31(1):15–21, 2006.

[110] Gerard Jounghyun Kim et al. A swot analysis of the field of virtual reality rehabilitation and therapy. *Presence: Teleoperators and Virtual Environments*, 14(2): 119–146, 2005.

[111] Gazihan Alankus, Amanda Lazar, Matt May, and Caitlin Kelleher. Towards customizable games for stroke rehabilitation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2113–2122. ACM, 2010.

[112] C Leonardi, C Mennecozzi, E Not, F Pianesi, and M Zancanaro. Designing a familiar technology for elderly people. *Gerontechnology*, 7(2):151, 2008.

[113] H Peyton Young. The dynamics of social innovation. *Proceedings of the National Academy of Sciences*, 108(Supplement 4):21285–21291, 2011.

[114] Jeffrey A Kleim, Theresa A Jones, and Timothy Schallert. Motor enrichment and the induction of plasticity before or after brain injury. *Neurochemical research*, 28(11):1757–1769, 2003.

[115] David Jack, Rares Boian, Alma S Merians, Marilyn Tremaine, Grigore C Burdea, Sergei V Adamovich, Michael Recce, and Howard Poizner. Virtual reality-enhanced stroke rehabilitation. *IEEE transactions on neural systems and rehabilitation engineering*, 9(3):308–318, 2001.

[116] Bob G Witmer and Michael J Singer. Measuring presence in virtual environments: A presence questionnaire. *Presence: Teleoperators and virtual environments*, 7(3):225–240, 1998.

[117] David Felce and Jonathan Perry. Quality of life: Its definition and measurement. *Research in developmental disabilities*, 16(1):51–74, 1995.

[118] Amulya Yadav, Hau Chan, Albert Xin Jiang, Haifeng Xu, Eric Rice, and Milind Tambe. Using social networks to aid homeless shelters: Dynamic influence maximization under uncertainty. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*, pages 740–748, 2016.

[119] Pedro Domingos and Matt Richardson. Mining the network value of customers. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 57–66, 2001.

[120] Matthew Richardson and Pedro Domingos. Mining knowledge-sharing sites for viral marketing. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 61–70, 2002.

[121] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146, 2003.

[122] Shipra Agrawal and Navin Goyal. Thompson sampling for contextual bandits with linear payoffs. In *Proceedings of the 30th International Conference on Machine Learning, (ICML'13)*, pages 127–135, 2013.

[123] Jacob Goldenberg, Barak Libai, and Eitan Muller. Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing letters*, 12(3):211–223, 2001.

[124] Jacob Goldenberg, Barak Libai, and Eitan Muller. Using complex systems analysis to advance marketing theory development: Modeling heterogeneity effects on new product growth through stochastic cellular automata. *Academy of Marketing Science Review*, 2001:1, 2001.

[125] Gerard Cornuejols, Marshall L Fisher, and George L Nemhauser. Exceptional paper–location of bank accounts to optimize float: An analytic study of exact and approximate algorithms. *Management science*, 23(8):789–810, 1977.

[126] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions–i. *Mathematical Programming*, 14(1):265–294, 1978.

Notations of Chapter 2

a_i	an action or pure strategy in for player i
A_i	a set of actions or pure strategies for player i
$\mathcal{A}$	online algorithm
$\mathbf{b}^j$	the j-th base in B
B	basis of dimension n in BGA
$BR(\sigma_l)$	the set of best response strategies for the follower
$\mathbf{c}^t$	cost vector at iteration t
$\hat{\mathbf{c}}^t$	unbiased estimator of $\mathbf{c}^t$
C	a coalition, i.e., a subset of N
CS	a coalition structure, i.e., a partition of N
E	the edge set in G
f	the player "follower" in Stackelberg games
f^t	a convex loss function at iteration t
$\mathcal{F}$	the bounded family of loss functions available for the adversary
g	the follower response function
G	a network or graph

$\mathcal{H}_t$	the history information by time t (inclusive)
$\mathcal{H}_t^l$	the history information available for the learner by time t
i	a player in set N
l	the player "leader" in Stackelberg games
l^t	linear loss in online linear optimization at iteration t
$\hat{l}^t$	unbiased estimator of l^t
N	a set of players
R_T	cumulative regret until time T
S	a set of skills
S_i	the set of skills owned by agent i
$S(C)$	the set of skills owned by a coalition C
$S(t)$	the set of skills required by a task t
t	a task in T
T	a set of tasks (coalitional skill games); time horizon (online convex optimization)
$T(C)$	the set of tasks the coalition C can perform
u	task value function mapping a subset of tasks to a real number
u_i	payoff function for player i
$U_i(\sigma_i, \sigma_{-i})$	utility (expected payoff) of player i w.r.t. σ
$\mathcal{V}$	adversary model
v	characteristic function

V	the vertex set in G
$\mathbf{w}^t$	decision point picked at iteration t
$\mathbf{y}$	payoff vector
$\mathbf{z}^t$	random noise at iteration t in FPL
γ	learning parameter in BGA
Δ^{A_i}	the simplex set of all probability distributions over $A_i i$
ϵ	learning parameter in FPL
σ	the mixed strategy profile of all players
σ_i	the mixed strategy of player i
$\sigma_i(a_i)$	probability of selecting action a_i
σ_{-i}	the mixed strategy profile for players except i

Notations of Chapter 3

a	an instance of optimization problem A
$\mathbf{a}$	an attacking plan
a_t	the number of targets of type t that coalition C plans to attack
A	the optimization problem to be reduced to B in linear reduction
b	an instance of optimization problem B
B	the optimization problem reduced from A in linear reduction
$\mathbf{B}$	decision variable representing the defender's strategy
B_{ij}	binary indicator denoting whether edge (i, j) is blocked
$cost(x)$	objective value of x in problem a
$cost(y)$	objective value of y in problem b
C	a coalition, i.e., a set of terrorists
$\mathcal{C}$	the set of all coalitions
$\mathcal{C}'$	the set of coalitions whose induced subgraphs are connected components themselves or consist of several connected components of $G(\mathbf{B})$

C_k	the k-th coalition in $\mathcal{C}$
CQ	a collection of subsets of Q
CQ^*	the exact cover of Q
CS	a coalition structure
CS^*	the coalition structure with the maximum total values
$CS^*(\mathbf{B})$	the coalition structure consisting of all coalitions whose induced subgraphs are connected components of $G(\mathbf{B})$
$\mathcal{D}$	a polyhedron of the dual master problem confined by the complementary slackness conditions
E	the set of edges representing connections between pairs of terrorists
E'	the set of edges in G'
E^*	the minimum weight set of edges in k-TERMINAL CUT problem
$E(\mathbf{B})$	the set of remaining edges after defender's block specified by $\mathbf{B}$
$E(C)$	the edge set of induced subgraph G_C
f	a polynomial time algorithm in linear reduction
g	a polynomial time algorithm in linear reduction
$\mathbf{f}, \mathbf{g}$	dual solution of the restricted master problem
G	terrorist network
G'	a connected component of G
G_C	induced subgraph of C

$G(\mathbf{B})$	the remaining graph after defender's block specified by $\mathbf{B}$
h_{ij}^{+}, h_{ij}^{-}	the amount of flow from starting point to ending point l, passing through edge (i, j) in positive direction $i \rightarrow j$ and negative direction $j \rightarrow i$ respectively
H	the set of k specified terminals in k-TERMINAL CUT problem
i	a terrorist in N
k	number of terminals in k-TERMINAL CUT problem
K_l	the l-th connected component in $\mathcal{K}$
$\mathcal{K}$	the set of connected components of $G(\mathbf{B})$
m_{is}	capacity of skill s for terrorist i, i.e., the maximum number of attacks supported by skill s
M	a large enough constant
N	a set of terrorists (individuals or groups)
N'	the set of terrorists in G'
$opt(a)$	the optimal solution of problem a
$opt(b)$	the optimal solution of problem b
p_t	value on a target of type t for the terrorists
Q	a finite set in the EXACT-COVER problem
r_k	reduced cost of coalition C_k, associated with variable x_k
S	the set of all skills
S_i	the set of skills owned by terrorist i
$S(t)$	the set of skills required to attack a target of type t

t	a type of target in T
T	a set of types of targets
$u(\mathbf{a})$	the sum of values of all targets attacked
$U_d(\mathbf{B})$	the defender's utility of playing strategy $\mathbf{B}$
v_k	the value of the k-th coalition in $\mathcal{C}$
$v(C)$	value of coalition C
w_{ij}	positive weight associated with edge (i, j) in k-TERMINAL CUT problem
W_{max}	the maximum cost of blocking an edge in G
x	a solution of problem a
$\mathbf{x}$	decision variable of attacker's strategy
x_k	binary indicator denoting whether C_k belongs to the coalition structure formed by the terrorists
y	a solution of problem b
$\mathbf{y}$	payoff vector
y_i	payoff for each terrorist i
α	a positive constant in linear reduction
α_i	indicator of whether i is in C of the BMILP (3.10)
α_{ki}	binary indicator of whether i is in coalition C_k
β	a positive constant in linear reduction
β_l^C	binary indicator indicating whether the induced subgraph G_C is K_l itself or consists of K_l

ϵ	parameter of ϵ-core, the set of all outcomes that any coalition cannot gain more than ϵ by deviating from the current coalition structure
ϵ^*	parameter of the least-core
λ_{ij}	blocking cost associated with edge (i, j)
$\boldsymbol{\phi}, \boldsymbol{\varphi}$	auxiliary variables in reformulated MILP (3.12)
μ, ω	vectors whose elements are uniformly distributed between 0 and 1
ξ	binary variable that $\xi_{ij} = \alpha_i \alpha_j$

Notations of Chapter 4

c_e	capacity on edge e
$\mathcal{C}$	cover set in $\mathcal{Q}$
d	Euclidean distance
e	an edge in E
E	the edges set in G
$\mathbf{f}$	an attacker strategy, i.e., a network flow
$\mathbf{f}^*$	minimax strategy for attacker
$\mathcal{F}$	the set of all feasible attacker strategies
f_p	the amount of adversary flow passing along path p
G	capacitated graph representing the transportation network
h	the marginal benefit of an allocation over the non-defending defender strategy
$\mathbf{H}$	the Hessian matrix of $\Phi(S, p)$
i	an inspection station
I	the set of all inspection stations
k	the number of security resources owned by the defender; the capacity in the Set-Cover problem

$\mathcal{N}^l(p)$	the set of neighbor paths of p within distance l
p	an s-t path in G
$\mathcal{P}$	the set of all s-t paths in G
$\mathcal{P}'$	the restricted set of s-t paths in G
$\mathcal{Q}$	a collection of subsets of U
R	reduced cost associated with path p
s	the source node in V
S	a defender pure strategy
S_i	binary indicator of whether i is operated
$\mathcal{S}$	the defender pure strategy space
$\mathcal{S}'$	the restricted defender pure strategy space
t	the sink node in V
U	ground set in the Set-Cover problem
u_e	dual variable associated with e
U_a	attacker's utility
U_d	defender's utility
v	a node in V
V	the nodes set in G
$\mathbf{x}$	a mixed defender strategy
$\mathbf{x}^*$	minimax strategy for defender
x_S	the probability that S is played
$\mathcal{X}$	the defender mixed strategy space

$y(\mathbf{x})$	the best response flow against defender mixed strategy $\mathbf{x}$
$\mathbf{y}$	$y_i = \alpha_{pi} \ln(1 - \tau_i)$
$\boldsymbol{\gamma}$	binary variable representing an s-t path p
λ	constant parameter in Waxman geographical model
τ_i	inspection probability of i
$\Phi(S, p)$	the proportion of adversary flow on path p not interdicted by the defender strategy S

Notations of Chapter 5

c_e	capacity on edge e
C	the spanning ratio of W
$\mathbf{f}$	an attacker strategy, i.e., a network flow
$\hat{\mathbf{f}}$	unbiased estimator of $\mathbf{f}$
$\mathbf{f}^t$	the adversarial network flow at round t
$\mathcal{F}^{qr}$	a pool of uniformly generated network flows for QR model adversary
G	capacitated graph representing the transportation network
m	the number of candidate paths, i.e., $m = \lvert\mathcal{P}\rvert$
$\bar{m}$	$\frac{m}{k}$
n	number of inspection stations
$\lvert\mathcal{N}\rvert$	number of nodes
$\mathcal{H}_0$	notation for no history information
$\mathcal{H}_t$	the history information of the game by time t (inclusive)
k	number of resources
p	an s-t path in G

$\mathcal{P}$	the set of candidate s-t paths in G
$r^{S,i,\mathbf{f}}$	the amount of interdicted flow on operated checkpoint $i \in S$ against adversarial flow $\mathbf{f}$ when S is played
$\mathbf{r}^t$	the vector of semi-bandit feedbacks
$\hat{\mathbf{r}}^t$	unbiased estimator of $\mathbf{r}^t$
R_T	cumulative regret until time T
S	a defender pure strategy
$\mathcal{S}$	the defender pure strategy space
$\mathcal{S}^{eb}$	exploration basis
S^t	real-time defense at round t
$S(p,i)$	the set of checkpoints in S such that path p passes through before checkpoint i
t	an iteration of the repeated network flow interdiction game
u^t	SBGA's utility in round t
$\hat{u}^t$	the utility of GEX in the "imaginary" game
U_d	defender's utility
$\mathbf{w}$	a vector associated with S where $w_p = 1 - \Phi(S,p)$
$\mathbf{w}^t$	the decision point selected by defender at round t
$\hat{\mathbf{w}}^t$	the decision recommended by GEX in round t
$\mathbf{w}_p^{S,i}$	the proportion of flow on path p interdicted at checkpoint i when S is operated
W^S	matrix consisting of $\mathbf{w}_p^{S,i}$ over paths in $\mathcal{P}$

$\mathbf{z}$	random noise
$\alpha_{p,i}$	indicator of whether i is on path p
Δ^t	$\hat{\mathbf{f}}^t - \mathbf{f}^t$
ϵ	learning parameter in GEX
λ	the learning parameter of SBGA; the parameter in QR model
τ_{min}	minimum interdiction probability of all checkpoints
τ_{max}	maximum interdiction probability of all checkpoints
ϕ	a mapping which maps S to its corresponding vector $\mathbf{w}$
$\phi(\mathcal{S})$	the set of all points mapped from $\mathcal{S}$ with ϕ
$\Phi(S, p)$	the proportion of adversary flow on path p not interdicted by the defender strategy S