# Enhancing transportation security against terrorist attacks

Sunghoon Hong*

March 30, 2009

**Abstract**

We study a model of strategic interaction between a terrorist organization and a security agency in a transportation network carrying passengers and freight between locations. By carrying explosives to a target location through the transportation network, the terrorist organization can damage the target and disrupt the operation of the network. While gaining utility from the damage of the target and from the disruption of the network, the terrorist organization incurs the cost of carrying explosives. A security agency is informed of the terrorist attack. By shutting down some transportation routes in the network, the security agency can protect the target from the attack. Since the shutdown of routes disrupts the operation of the network, the security agency incurs the cost of shutting down transportation routes. The security agency also loses utility from the damage of the target. In this model we find an optimal security policy under which the security agency can protect the target from devastating terrorism and effectively operate the network. To understand how the terrorist organization commits terrorism under the optimal security policy, we find a class of subgame perfect equilibria of this model. We also introduce algorithms to find a maximum flow and a minimum cut in a transportation network.

JEL classification: C63; D85

Keywords: Network security; Transportation networks; Subgame perfect equilibria; Maximum flow; Minimum cut

## 1 Introduction

Network security is a crucial issue for a connected world. The use of networks is pervasive; transportation, communication, trade, social interaction, and financial transactions are a few examples of activities carried out on networks. Because of their importance in modern society networks may be targets for antisocial behavior. There are many ways in which individuals may use and abuse networks. Consider two hypothetical examples:

*E-mail: sunghoon.hong@vanderbilt.edu

(i) A terrorist aims to damage a particular facility and disrupt the operation of a transportation network. The terrorist can choose to carry explosives through the network and possibly even create explosions in the transportation network itself. A security agency aims to protect the terrorist's target and operate the transportation network well. To prevent explosives from being carried to the target, the agency can monitor more closely transportation routes, by inspecting all backpacks and bags, for example, or, in the event of advance information, by shutting down parts or all of the transportation network.

(ii) A malicious hacker aims to damage a particular computer system and disrupt the operation of a computer network. The hacker can choose to transmit malicious code through the network. A network management agency aims to protect the hacker's target and operate the network effectively. To prevent malicious code from gaining access to the target, the agency can choose to slow, hide, or block some connections in the network.

These examples show that it is crucial for network management agencies to find ways to securely operate networks. In this paper we study how to enhance network security against malicious attacks. We analyze this issue by combining game theory with flow networks.

A flow network is a directed graph with two distinguished vertices, a source and a sink, together with arc capacities. As first introduced in Ford and Fulkerson [2] and studied by others[1], there are numerous algorithms to find a maximum flow or a minimum cut (or both) in a flow network. Examples of flow networks are abundant: Transportation networks, computer networks, electricity networks, and financial transaction networks. For expositional purposes, we present all the analysis that follows in terms of the transportation network example (i).

There is a transportation network carrying passengers and freight between locations. By carrying explosives to a target location through the transportation network, a terrorist organization can damage the target and disrupt the operation of the network. While the terrorist organization gains utility from the damage of the target and from the disruption of the network, the terrorist organization incurs the cost of carrying explosives to the target through the network. A security agency is informed of the terrorist attack. By shutting down some routes in the transportation network, the security agency can protect the target from the attack. Since the shutdown of routes disrupts the operation of the network, the security agency incurs the cost of shutting down transportation routes. The security agency also loses utility from the damage of the target.

In this model we first find an optimal security policy for the security agency. Under the optimal policy, the security agency can protect the target from devastating terrorism and effectively operate the transportation network. To understand how the terrorist organization commits terrorism under the optimal security policy, we find a class of subgame perfect equilibria of this model. We

[1] See for example Goldberg and Tarjan [3].

also introduce algorithms to find a maximum flow and a minimum cut in a transportation network.

The formation and operation of secure networks are an increasing concern for a network economy. Hong [6] studies a decentralized formation of secure networks; Goyal and Vigier [4] investigate a centralized formation of secure networks. Baccara and Bar-Isaac [1] study the formation of criminal networks and find optimal policies for a law enforcement agency. In this paper we focus on the secure operation of networks.

Kalai and Zemel [7] introduce a transferable utility game in which the worth of a coalition is defined as the value of a maximum flow in the flow network restricted to the members of the coalition. The game is called a flow game and studied by various authors[2]. In this paper we restrict our attention to a non-cooperative game theoretic analysis.

This paper is organized as follows. Section 2 introduces a transportation security game. Section 3 provides an optimal policy for the security agency and a class of equilibria of this game. Section 4 introduces algorithms that are useful in transportation networks. Concluding remarks follow in Section 5.

## 2 The model

There is a transportation network carrying passengers and freight between locations. Let $V$ be a set of locations with two distinguished locations, a base $x$ and a target $y$. We denote generic locations by $v \in V$ and $w \in V$. Let $A \subset V \times V$ be a set of transportation routes where each transportation route is an ordered pair of distinct locations. A transportation route $(v, w) \in A$ allows passengers and freight to be carried from location $v$ to location $w$. We assume that $|A| \geq |V| - 1$. Let $c : V \times V \to \mathbb{R}_+$ be a capacity function that associates with each $(v, w) \in V \times V$ a non-negative real value $c(v, w)$. The capacity $c(v, w)$ of a transportation route $(v, w)$ represents the maximum amount of passengers and freight that can be carried along the route. We assume that $c(v, w) = 0$ if $(v, w) \notin A$. Formally, a transportation network is defined as a collection $G := (V, A, x, y, c)$.

Let $G = (V, A, x, y, c)$ be a transportation network. There are two players, say player 1 and player 2. Let player 1 be a terrorist organization and let player 2 be a security agency. Player 1 sets up a base in $x \in V$. Player 1 aims to attack a target in $y \in V$ and disrupt the operation of network $G$. To damage the target and disrupt the operation, player 1 can choose to carry explosives from the base to the target through the transportation network. More precisely, player 1 chooses his strategy $s_1$ that is a flow (of carrying explosives) in $G$. A flow $f$ in $G$ is a real-valued function defined on $V \times V$ satisfying the following constraints:

$$f(v, w) \leq c(v, w) \text{ for each } (v, w) \in V \times V \quad \text{(capacity constraint)}, \tag{1}$$

$$f(v, w) = -f(w, v) \text{ for each } (v, w) \in V \times V \quad \text{(antisymmetry constraint)}, \tag{2}$$

[2] See for example Granot and Granot [5], Potters, Reijnierse, and Biswas [8], and Reijnierse, Maschler, Potters, and Tijs [9].

$$\sum_{w \in V} f(w, v) = 0 \text{ for each } v \in V \setminus \{x, y\} \quad \text{(conservation constraint).} \qquad (3)$$

Choosing his strategy $s_1$ that is a flow $f$, player 1 will carry $f(v, w)$ amount of explosives along a transportation route $(v, w)$.

**Remark 1** *The capacity constraint requires that the amount of explosives carried along a route be no more than the capacity of the route. The antisymmetry constraint says that if the amount of explosives carried from $w$ to $v$ is $f(w, v)$ then the amount of explosives carried from $v$ to $w$ is $-f(w, v)$. The conservation constraint requires that for any location $v$ other than the base and the target, the net amount of explosives carried to $v$ be equal to zero.* □

Let $val(f) := \sum_{v \in V} f(v, y)$ be the value of a flow $f$. The value $val(f)$ of a flow $f$ represents the net amount of explosives carried to target $y$ under $f$. A flow $f^o$ in $G$ is the zero flow if for each $(v, w) \in V \times V$, we have $f^o(v, w) = 0$. The zero flow will ensure that every network has at least one flow. Note that $val(f^o) = 0$. A flow $f^*$ in $G$ is a maximum flow if for each flow $f$, we have $val(f^*) \geq val(f)$. How to find a maximum flow in a transportation network will be explained in Section 4.

We assume that player 1 only chooses a strategy $s_1$ whose value is nonnegative; that is, $val(s_1) \geq 0$.

We are ready to introduce examples of strategies for player 1. The zero-flow strategy for player 1 is a strategy $s_1$ that is the zero flow in $G$. A maximum-flow strategy for player 1 is a strategy $s_1$ that is a maximum flow in $G$. Let $\mathcal{S}_1$ be the set of player 1's pure strategies.

Player 2 is informed of player 1's choice of strategy. Given the information, player 2 aims to protect target $y$ and operate network $G$ effectively. To prevent explosives from being carried to the target, player 2 can choose to shut down some transportation routes. More precisely, player 2 chooses her strategy $s_2(\cdot)$ such that for each $s_1 \in \mathcal{S}_1$, $s_2(s_1)$ is either a cut (of transportation routes) in $G$ or the empty set $\emptyset$. A cut $K, \overline{K}$ in $G$ is a partition of the location set $V$ (that is, $K \cup \overline{K} = V$ and $K \cap \overline{K} = \emptyset$) such that $x \in K$ and $y \in \overline{K}$. Given player 1's strategy $s_1$, by choosing her strategy $s_2(\cdot)$ such that $s_2(s_1) = K, \overline{K}$, player 2 will shut down the transportation routes from any location in $K$ to any location in $\overline{K}$. Given $s_1$, by choosing $s_2(\cdot)$ such that $s_2(s_1) = \emptyset$, player 2 will shut down no transportation route.

Let $cap(K, \overline{K}) := \sum_{v \in K, w \in \overline{K}} c(v, w)$ be the capacity of a cut $K, \overline{K}$. The capacity $cap(K, \overline{K})$ of a cut $K, \overline{K}$ represents the total capacity of the transportation routes shut down under $K, \overline{K}$. To simplify the exposition, we assume that the capacity of the empty set equals to zero; that is, $cap(\emptyset) := 0$. A cut $K^*, \overline{K^*}$ in $G$ is a minimum cut if for each cut $K, \overline{K}$, we have $cap(K^*, \overline{K^*}) \leq cap(K, \overline{K})$. How to find a minimum cut in a transportation network will be explained in Section 4.

We are ready to introduce examples of strategies for player 2. The empty-set strategy for player 2 is a strategy $s_2(\cdot)$ such that for each $s_1 \in \mathcal{S}_1$, $s_2(s_1)$ is the empty set. A minimum-cut strategy for player 2 is a strategy $s_2(\cdot)$ such that for each $s_1 \in \mathcal{S}_1$, $s_2(s_1)$ is a minimum cut in $G$. Let $\mathcal{S}_2$ be the set of player 2's pure strategies.

We provide an example of a transportation network, together with examples of strategies for both players.

**Example 1** *Let $G = (V, A, x, y, c)$ be a transportation network where $V = \{x, v_1, v_2, y\}$ is a set of locations with a base $x$ and a target $y$, $A = \{(x, v_1), (x, v_2), (v_1, v_2), (v_2, y), (y, v_1)\}$ is a set of transportation routes, and $c$ is a capacity function such that $c(x, v_1) = 4$, $c(x, v_2) = 1$, $c(v_1, v_2) = 2$, $c(v_2, y) = 5$, and $c(y, v_1) = 2$, as in Figure 1.*

*Player 1 chooses a maximum-flow strategy $s_1 = f^*$ such that $f^*(x, v_1) = 2$, $f^*(x, v_2) = 1$, $f^*(v_1, v_2) = 2$, $f^*(v_2, y) = 3$, and $f^*(y, v_1) = 0$, as indicated in bold numbers. Player 2 chooses a minimum-cut strategy $s_2(\cdot) = K^*, \overline{K^*}$ such that $K^* = \{x, v_1\}$, as indicated in solid circles.* □

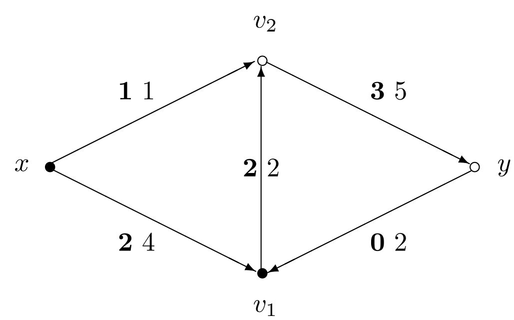


**Figure 1** Each of solid and blank circles indicates a location; each arrow indicates a transportation route; in each pair of numbers, the first bold number indicates a flow of a route and the second light number indicates the capacity of a route.

Once carrying explosives to target $y$, player 1 creates explosions to damage $y$. The net amount of explosives carried to the target determines how severely the target is damaged. Let $r : \mathbb{R}_+ \to \mathbb{R}_+$ be a target damage function that associates with each $q \in \mathbb{R}_+$ a non-negative real value $r(q)$. We assume that $r$ is continuous, (weakly) increasing, and $r(0) = 0$. If player 1 chooses $s_1 \in \mathcal{S}_1$, the net amount of explosives carried to the target is $val(s_1)$ and thus the damage of the target is $r(val(s_1))$.

By shutting down some transportation routes, player 2 prevents explosives from being carried to target $y$. If player 1 chooses $s_1 \in \mathcal{S}_1$ and player 2 chooses $s_2(\cdot) \in \mathcal{S}_2$ such that $s_2(s_1)$ is a cut $K, \overline{K}$ in $G$, player 2 shuts down all the transportation routes from any location in $K$ to any location in $\overline{K}$ and thus player 1 cannot carry explosives to $y$. If player 1 chooses $s_1 \in \mathcal{S}_1$ and player 2 chooses $s_2(\cdot) \in \mathcal{S}_2$ such that $s_2(s_1)$ is the empty set, player 2 operates the transportation network as usual and thus player 1 can carry explosives to $y$. For each $(s_1, s_2(\cdot)) \in \mathcal{S}_1 \times \mathcal{S}_2$, let

$$I(s_2(s_1)) = \begin{cases} 0, & \text{if } s_2(s_1) \text{ is a cut in } G; \\ 1, & \text{if } s_2(s_1) \text{ is the empty set} \end{cases}$$

be a damage indicator function. Then, for each $(s_1, s_2(\cdot)) \in \mathcal{S}_1 \times \mathcal{S}_2$, player 1 gains $r(val(s_1)) \cdot I(s_2(s_1))$ and player 2 loses the same amount from the damage of the target.

The shutdown of routes disrupts the operation of transportation network $G$. The total capacity of the routes shut down by player 2 will determine how severely network $G$ is disrupted. Let $\ell : \mathbb{R}_+ \to \mathbb{R}_+$ be a network disruption function that associates with each $q \in \mathbb{R}_+$ a non-negative real value $\ell(q)$. We assume that $\ell$ is (weakly) increasing and $\ell(0) = 0$. If player 1 chooses $s_1 \in \mathcal{S}_1$ and player 2 chooses $s_2(\cdot) \in \mathcal{S}_2$, the total capacity of the routes shut down under $s_2(s_1)$ is $cap(s_2(s_1))$ and thus the disruption of the network is $\ell(cap(s_2(s_1)))$. Then, for each $(s_1, s_2(\cdot)) \in \mathcal{S}_1 \times \mathcal{S}_2$, player 1 gains $\ell(cap(s_2(s_1)))$ and player 2 loses the same amount from the disruption of the network.

While carrying explosives to target $y$, player 1 incurs the cost of committing terrorism, which is determined by the net amount of explosives carried to $y$. Let $h : \mathbb{R}_+ \to \mathbb{R}_+$ be a terrorism cost function that associates with each $q \in \mathbb{R}_+$ a non-negative real value $h(q)$. We assume that $h$ is continuous, (weakly) increasing, and $h(0) = 0$. If player 1 chooses $s_1 \in \mathcal{S}_1$, the net amount of explosives carried to the target is $val(s_1)$ and thus the cost of committing terrorism is $h(val(s_1))$.

Let $b \in \mathbb{R}_+$ be a benefit obtained by player 2 from operating transportation network $G$.

For each $(s_1, s_2(\cdot)) \in \mathcal{S}_1 \times \mathcal{S}_2$, the utility of player 1 is defined as

$$u_1(s_1, s_2(\cdot)) = r(val(s_1)) \cdot I(s_2(s_1)) + \ell(cap(s_2(s_1))) - h(val(s_1)),$$

and the utility of player 2 is defined as

$$u_2(s_1, s_2(\cdot)) = b - r(val(s_1)) \cdot I(s_2(s_1)) - \ell(cap(s_2(s_1))).$$

**Remark 2** *For each $(s_1, s_2(\cdot)) \in \mathcal{S}_1 \times \mathcal{S}_2$, if $s_2(s_1)$ is a cut in $G$, we have $I(s_2(s_1)) = 0$. Thus, we have $u_1(s_1, s_2(\cdot)) = \ell(cap(s_2(s_1))) - h(val(s_1))$ and $u_2(s_1, s_2(\cdot)) = b - \ell(cap(s_2(s_1)))$. For each $(s_1, s_2(\cdot)) \in \mathcal{S}_1 \times \mathcal{S}_2$, if $s_2(s_1)$ is the empty set, we have $I(s_2(s_1)) = 1$ and $cap(s_2(s_1)) = cap(\emptyset) = 0$. Since $\ell(0) = 0$, we have $u_1(s_1, s_2(\cdot)) = r(val(s_1)) - h(val(s_1))$ and $u_2(s_1, s_2(\cdot)) = b - r(val(s_1))$.*
□

A transportation security game is defined as a collection $\Gamma := (G, \mathcal{S}_1, \mathcal{S}_2, u_1, u_2)$.

**Definition 1** *A strategy profile $(s_1, s_2(\cdot))$ is a subgame perfect equilibrium of transportation security game $\Gamma$ if (i) for each $s_1' \in \mathcal{S}_1$ and each $s_2'(\cdot) \in \mathcal{S}_2$, we have $u_2(s_1', s_2(\cdot)) \geq u_2(s_1', s_2'(\cdot))$, and (ii) for each $s_1' \in \mathcal{S}_1$, we have $u_1(s_1, s_2(\cdot)) \geq u_1(s_1', s_2(\cdot))$.*

For a given strategy profile, a subgame perfect equilibrium (SPE) requires that (i) in each subgame induced by player 1's choice of strategy player 2 maximize her utility and that (ii) player 1 maximize his utility.

# 3 The results

Let $\Gamma = (G, \mathcal{S}_1, \mathcal{S}_2, u_1, u_2)$ be a transportation security game. We show that every transportation security game has a class of subgame perfect equilibria. Since player 1 moves first and player 2 moves second, by backward induction, we first study player 2's choice of strategy.

Let $\theta := \min\{q \in \mathbb{R}_+ : r(q) = \ell(cap(K^*, \overline{K^*}))\}$. The minimum over the empty set is defined as infinity. Clearly, $\ell(cap(K^*, \overline{K^*}))$ represents the disruption of the network by the shutdown of routes under a minimum cut $K^*, \overline{K^*}$. Thus, $\theta$ represents the minimum net amount of explosives such that the damage of the target is equally severe as the disruption of the network. If $\theta = \infty$, then for each $q \in \mathbb{R}_+$, we have $r(q) < \ell(cap(K^*, \overline{K^*}))$. If $\theta < \infty$, then $r(\theta) = \ell(cap(K^*, \overline{K^*}))$.

A $\theta$-threshold minimum-cut strategy for player 2 is a strategy $s_2(\cdot) \in \mathcal{S}_2$ such that for each $s_1 \in \mathcal{S}_1$,

$$s_2(s_1) = \begin{cases} K^*, \overline{K^*}, & \text{if } val(s_1) \geq \theta; \\ \emptyset, & \text{if } val(s_1) < \theta. \end{cases}$$

If $\theta = \infty$, then a $\theta$-threshold minimum-cut strategy for player 2 is the empty-set strategy for player 2. If $\theta = 0$, then a $\theta$-threshold minimum-cut strategy for player 2 is a minimum-cut strategy for player 2.

**Remark 3** *Let $s_2(\cdot)$ be a $\theta$-threshold minimum-cut strategy for player 2. For each $s_1' \in \mathcal{S}_1$, if $val(s_1') \geq \theta$, we have $s_2(s_1') = K^*, \overline{K^*}$. From Remark 2, we have $u_1(s_1', s_2(\cdot)) = \ell(cap(K^*, \overline{K^*})) - h(val(s_1'))$ and $u_2(s_1', s_2(\cdot)) = b - \ell(cap(K^*, \overline{K^*}))$. For each $s_1' \in \mathcal{S}_1$, if $val(s_1') < \theta$, we have $s_2(s_1') = \emptyset$. From Remark 2, we have $u_1(s_1', s_2(\cdot)) = r(val(s_1')) - h(val(s_1'))$ and $u_2(s_1', s_2(\cdot)) = b - r(val(s_1'))$.* □

The following lemma shows that in each subgame induced by player 1's choice of strategy, player 2 can maximize her utility by choosing a $\theta$-threshold minimum-cut strategy. To put it another way, a $\theta$-threshold minimum-cut strategy is an optimal policy for the security agency.

**Lemma 1** *If $s_2(\cdot)$ is a $\theta$-threshold minimum-cut strategy for player 2, then for each $s_1' \in \mathcal{S}_1$ and each $s_2'(\cdot) \in \mathcal{S}_2$, we have $u_2(s_1', s_2(\cdot)) \geq u_2(s_1', s_2'(\cdot))$.*

**Proof.** Let $s_2(\cdot)$ be a $\theta$-threshold minimum-cut strategy for player 2 and let $s_1'$ be a strategy for player 1. We assume that $\theta = \infty$. Then for each $q \in \mathbb{R}_+$, we have $r(q) < \ell(cap(K^*, \overline{K^*}))$. Thus, we have $r(val(s_1')) < \ell(cap(K^*, \overline{K^*}))$. Since $\theta = \infty$ and $val(s_1') < \infty$, we have $val(s_1') < \theta$. From Remark 3, we have $u_2(s_1', s_2(\cdot)) = b - r(val(s_1'))$.

Now, let $s'_2(\cdot)$ be a strategy for player 2. First, if $s'_2(s'_1)$ is a cut $K, \overline{K}$, from Remark 2, we have $u_2(s'_1, s'_2(\cdot)) = b - \ell(cap(K, \overline{K}))$. Since $cap(K, \overline{K}) \geq cap(K^*, \overline{K^*})$ and $\ell$ is increasing, we have $u_2(s'_1, s'_2(\cdot)) = b - \ell(cap(K, \overline{K})) \leq b - \ell(cap(K^*, \overline{K^*}))$. Altogether, we have

$$\begin{aligned} u_2(s'_1, s_2(\cdot)) - u_2(s'_1, s'_2(\cdot)) &\geq b - r(val(s'_1)) - b + \ell(cap(K^*, \overline{K^*})) \\ &= \ell(cap(K^*, \overline{K^*})) - r(val(s'_1)) \\ &> 0, \end{aligned}$$

since $r(val(s'_1)) < \ell(cap(K^*, \overline{K^*}))$. Next, if $s'_2(s'_1)$ is the empty set, from Remark 2, we have $u_2(s'_1, s'_2(\cdot)) = b - r(val(s'_1)) = u_2(s'_1, s_2(\cdot))$.

We now assume that $\theta < \infty$. Then $r(\theta) = \ell(cap(K^*, \overline{K^*}))$. We divide into two cases.

*Case 1*: Let $val(s'_1) \geq \theta$. Since $s_2(\cdot)$ is a $\theta$-threshold minimum-cut strategy for player 2 and $val(s'_1) \geq \theta$, from Remark 3, we have $u_2(s'_1, s_2(\cdot)) = b - \ell(cap(K^*, \overline{K^*}))$.

Now, let $s'_2(\cdot)$ be a strategy for player 2. First, if $s'_2(s'_1)$ is a cut $K, \overline{K}$, from Remark 2, we have $u_2(s'_1, s'_2(\cdot)) = b - \ell(cap(K, \overline{K}))$. Altogether, we have

$$\begin{aligned} u_2(s'_1, s_2(\cdot)) - u_2(s'_1, s'_2(\cdot)) &= b - \ell(cap(K^*, \overline{K^*})) - b + \ell(cap(K, \overline{K})) \\ &= \ell(cap(K, \overline{K})) - \ell(cap(K^*, \overline{K^*})) \\ &\geq 0, \end{aligned}$$

since $cap(K, \overline{K}) \geq cap(K^*, \overline{K^*})$ and $\ell$ is increasing.

Next, if $s'_2(s'_1)$ is the empty set, from Remark 2, we have $u_2(s'_1, s'_2(\cdot)) = b - r(val(s'_1))$. Altogether, we have

$$\begin{aligned} u_2(s'_1, s_2(\cdot)) - u_2(s'_1, s'_2(\cdot)) &= b - \ell(cap(K^*, \overline{K^*})) - b + r(val(s'_1)) \\ &= r(val(s'_1)) - \ell(cap(K^*, \overline{K^*})) \\ &\geq r(\theta) - \ell(cap(K^*, \overline{K^*})) \\ &= 0, \end{aligned}$$

since $val(s'_1) \geq \theta$, $r$ is increasing, and $r(\theta) = \ell(cap(K^*, \overline{K^*}))$.

*Case 2*: Let $val(s'_1) < \theta$. Since $s_2(\cdot)$ is a $\theta$-threshold minimum-cut strategy for player 2 and $val(s'_1) < \theta$, from Remark 3, we have $u_2(s'_1, s_2(\cdot)) = b - r(val(s'_1))$. Since $val(s'_1) < \theta$ and $r$ is increasing, we have $u_2(s'_1, s_2(\cdot)) = b - r(val(s'_1)) \geq b - r(\theta)$.

Now, let $s'_2(\cdot)$ be a strategy for player 2. First, if $s'_2(s'_1)$ is a cut $K, \overline{K}$, from Remark 2, we have $u_2(s'_1, s'_2(\cdot)) = b - \ell(cap(K, \overline{K}))$. Since $cap(K, \overline{K}) \geq cap(K^*, \overline{K^*})$ and $\ell$ is increasing, we have $u_2(s'_1, s'_2(\cdot)) = b - \ell(cap(K, \overline{K})) \leq b - \ell(cap(K^*, \overline{K^*}))$. Altogether, we have

$$\begin{aligned} u_2(s'_1, s_2(\cdot)) - u_2(s'_1, s'_2(\cdot)) &\geq b - r(\theta) - b + \ell(cap(K^*, \overline{K^*})) \\ &= \ell(cap(K^*, \overline{K^*})) - r(\theta) \\ &= 0, \end{aligned}$$

since $r(\theta) = \ell(cap(K^*, \overline{K^*}))$.

Next, if $s_2'(s_1')$ is the empty set, from Remark 2, we have $u_2(s_1', s_2'(\cdot)) = b - r(val(s_1')) = u_2(s_1', s_2(\cdot))$. Therefore, we conclude that for each $s_1' \in \mathcal{S}_1$ and each $s_2'(\cdot) \in \mathcal{S}_2$, $u_2(s_1', s_2(\cdot)) \geq u_2(s_1', s_2'(\cdot))$. $\square$

Let $val(\mathcal{S}_1) := \{val(s_1) : s_1 \in \mathcal{S}_1\}$ be the image of $\mathcal{S}_1$ under $val(\cdot)$. The following lemma shows that $val(\mathcal{S}_1)$ is a compact set.

**Lemma 2** *The set $val(\mathcal{S}_1)$ is compact.*

**Proof.** We first note that for each $s_1 \in \mathcal{S}_1$, we have $0 \leq val(s_1) \leq val(f^*)$. Thus, we have $val(\mathcal{S}_1) \subseteq [0, val(f^*)]$. We now show that $[0, val(f^*)] \subseteq val(\mathcal{S}_1)$. It suffices to show that for each $q \in [0, val(f^*)]$, there is $s_1 \in \mathcal{S}_1$ such that $val(s_1) = q$. For each $q \in [0, val(f^*)]$, let $\lambda = \frac{q}{val(f^*)}$ and let $f' = \lambda f^*$. Since $f^*$ is a maximum flow in $G$ and $\lambda \in [0,1]$, $f'$ satisfies constraints (1), (2), and (3). Thus, $f'$ is a flow in $G$. Since $val(\lambda f^*) = \lambda val(f^*)$ and $\lambda = \frac{q}{val(f^*)}$, we have $val(f') = q$. Therefore, for each $q \in [0, val(f^*)]$, there is a flow in $G$ whose value is $q$. Since $val(\mathcal{S}_1) = [0, val(f^*)]$, the set is compact. $\square$

For each $q \in val(\mathcal{S}_1)$, let

$$\widetilde{u}_1(q) = \begin{cases} \ell(cap(K^*, \overline{K^*})) - h(q), & \text{if } q \geq \theta; \\ r(q) - h(q), & \text{if } q < \theta \end{cases}$$

be the utility of player 1 defined on $val(\mathcal{S}_1)$ when player 2 chooses a $\theta$-threshold minimum-cut strategy. The following lemma shows that $\widetilde{u}_1$ is a continuous function on $val(\mathcal{S}_1)$.

**Lemma 3** *The function $\widetilde{u}_1$ is continuous on $val(\mathcal{S}_1)$.*

**Proof.** We first assume that $\theta = \infty$. Then for each $q \in val(\mathcal{S}_1)$, we have $\widetilde{u}_1(q) = r(q) - h(q)$. Since both functions $r$ and $h$ are continuous and $val(\mathcal{S}_1) \subset \mathbb{R}_+$, $\widetilde{u}_1$ is continuous on $val(\mathcal{S}_1)$. We next assume that $\theta < \infty$. Then $r(\theta) = \ell(cap(K^*, \overline{K^*}))$. Since $r$ and $h$ are continuous, $\widetilde{u}_1$ is continuous at each $q \neq \theta$. Since $r$ and $h$ are continuous and $r(\theta) = \ell(cap(K^*, \overline{K^*}))$, $\widetilde{u}_1$ is continuous at $\theta$. Thus, $\widetilde{u}_1$ is continuous on $val(\mathcal{S}_1)$. $\square$

From Lemmas 2 and 3, we know that $val(\mathcal{S}_1)$ is a compact set and that $\widetilde{u}_1$ is a continuous function on $val(\mathcal{S}_1)$. Then $\widetilde{u}_1$ has a maximizer. Let

$$q^* \in \arg \max_{q \in val(\mathcal{S}_1)} \widetilde{u}_1(q)$$

be a maximizer of $\widetilde{u}_1$ on $val(\mathcal{S}_1)$. Let $\lambda^* := \frac{q^*}{val(f^*)}$ be the ratio of $q^*$ to $val(f^*)$. A $q^*$-value flow strategy for player 1 is a strategy $s_1 \in \mathcal{S}_1$ such that $val(s_1) = q^*$. An example of a $q^*$-value flow strategy for player 1 is $s_1 = \lambda^* f^*$.

For each $(s_1, s_2(\cdot)) \in \mathcal{S}_1 \times \mathcal{S}_2$, we say that $(s_1, s_2(\cdot))$ is a $q^*$-value $\theta$-threshold strategy profile if $s_1$ is a $q^*$-value flow strategy for player 1 and $s_2(\cdot)$ is a $\theta$-threshold minimum-cut strategy for player 2. Now we are ready to show that every transportation security game has a class of subgame perfect equilibria.

**Proposition 1** *Every $q^*$-value $\theta$-threshold strategy profile is a subgame perfect equilibrium of transportation security game $\Gamma$.*

**Proof.** Let $\Gamma = (G, \mathcal{S}_1, \mathcal{S}_2, u_1, u_2)$ be a transportation security game. Let $(s_1, s_2(\cdot))$ be a $q^*$-value $\theta$-threshold strategy profile. Since $s_2(\cdot)$ is a $\theta$-threshold minimum-cut strategy for player 2, from Lemma 1, for each $s_1' \in \mathcal{S}_1$ and each $s_2'(\cdot) \in \mathcal{S}_2$, we have $u_2(s_1', s_2(\cdot)) \geq u_2(s_1', s_2'(\cdot))$. Since $s_1$ is a $q^*$-value flow strategy for player 1, we have $val(s_1) = q^*$. Since $q^*$ is a maximizer of $\widetilde{u}_1$ on $val(\mathcal{S}_1)$, for each $s_1' \in \mathcal{S}_1$, we have $\widetilde{u}_1(val(s_1)) \geq \widetilde{u}_1(val(s_1'))$. Since $s_2(\cdot)$ is a $\theta$-threshold minimum-cut strategy for player 2, we have $\widetilde{u}_1(val(s_1)) = u_1(s_1, s_2(\cdot))$ and $\widetilde{u}_1(val(s_1')) = u_1(s_1', s_2(\cdot))$. Thus, for each $s_1' \in \mathcal{S}_1$, we have $u_1(s_1, s_2(\cdot)) \geq u_1(s_1', s_2(\cdot))$. Therefore, $(s_1, s_2(\cdot))$ is a subgame perfect equilibrium of transportation security game $\Gamma$. $\square$

# 4 The algorithms

To damage a target location and disrupt the operation of a transportation network, a terrorist organization may want to carry as many explosives as possible through the network. In other words, the terrorist organization may want to choose a maximum flow of carrying explosives in the network. To protect the target and operate the transportation network effectively, a security agency may want to shut down some routes in the network without unnecessary sacrifice. In other words, the security agency may want to choose a minimum cut of transportation routes in the network. In this section we present an algorithm to find a maximum flow in a transportation network as well as an algorithm to find a minimum cut in a transportation network.

We first introduce some notations. Let $G = (V, A, x, y, c)$ be a transportation network. A preflow $\widetilde{f}$ in $G$ is a real-valued function defined on $V \times V$ satisfying the following constraints:

$$\widetilde{f}(v, w) \leq c(v, w) \text{ for all } (v, w) \in V \times V \quad \text{(capacity constraint)}, \tag{4}$$

$$\widetilde{f}(v, w) = -\widetilde{f}(w, v) \text{ for all } (v, w) \in V \times V \quad \text{(antisymmetry constraint)}, \tag{5}$$

$$\sum_{w \in V} \widetilde{f}(w, v) \geq 0 \text{ for all } v \in V \setminus \{x\} \quad \text{(non-negativity constraint)}. \tag{6}$$

For each $v \in V$ and each preflow $\widetilde{f}$, let $e(v; \widetilde{f}) = \sum_{w \in V} \widetilde{f}(w, v)$ be the flow excess of $v$ under $\widetilde{f}$. For each $(v, w) \in V \times V$ and each preflow $\widetilde{f}$, let $z(v, w; \widetilde{f}) = c(v, w) - \widetilde{f}(v, w)$ be the residual capacity of $(v, w)$ under $\widetilde{f}$. For each $(v, w) \in V \times V$ and each preflow $\widetilde{f}$, we say that $(v, w)$ is a residual route under $\widetilde{f}$ if $z(v, w; \widetilde{f}) > 0$. Let $d : V \to \mathbb{N}_0 \cup \{\infty\}$ be a labeling function that associates with each $v \in V$ either a non-negative integer or infinity $d(v)$. For each $v \in V \setminus \{x, y\}$ and each preflow $\widetilde{f}$, we say that location $v$ is active under $\widetilde{f}$ if $d(v) < \infty$ and $e(v; \widetilde{f}) > 0$. For each $v, w \in V$, we say that $\{v, w\}$ is an (undirected) edge of $G$ if either $(v, w) \in A$ or $(w, v) \in A$.

We are ready to introduce two operations, push and relabel, which are used repeatedly in the algorithms. For each $(v, w) \in V \times V$, we say that $push(v, w)$ is applicable if $v$ is active, $(v, w)$ is a residual route, and $d(v) = d(w) + 1$. We apply $push(v, w)$ by sending $\delta$ units of flow from $v$ to $w$ where $\delta$ is the minimum of the flow excess of $v$ and the residual capacity of $(v, w)$. For each $v \in V$, we say that $relabel(v)$ is applicable if $v$ is active and for each $w \in V$ such that $(v, w)$ is a residual route, $d(v) \leq d(w)$. We apply $relabel(v)$ by replacing the current labeling of $v$ with the minimum labeling that makes a push operation applicable. A summary of the operations appears in Figure 2.

*Push*$(v, w)$.
Applicability: $v$ is active, $z(v, w; \widetilde{f}) > 0$ **and** $d(v) = d(w) + 1$.
Action: Send $\delta = \min\{e(v; \widetilde{f}), z(v, w; \widetilde{f})\}$ units of flow from $v$ to $w$ as follows:
$\widetilde{f}(v, w) \leftarrow \widetilde{f}(v, w) + \delta$; $\widetilde{f}(w, v) \leftarrow \widetilde{f}(w, v) - \delta$;
$e(v; \widetilde{f}) \leftarrow e(v; \widetilde{f}) - \delta$; $e(w; \widetilde{f}) \leftarrow e(w; \widetilde{f}) + \delta$.

*Relabel*$(v)$.
Applicability: $v$ is active **and** $\forall w \in V$ such that $z(v, w; \widetilde{f}) > 0$, $d(v) \leq d(w)$.
Action: $d(v) \leftarrow \min\{d(w) + 1 : (v, w)$ is a residual route under $\widetilde{f}\}$.
If this minimum is over the empty set, $d(v) \leftarrow \infty$.

**Figure 2** Push and relabel operations

We now present the Goldberg-Tarjan Maximum-Flow algorithm. Let $G = (V, A, x, y, c)$ be a transportation network. Let $\pi$ be an ordering of $V$.

**Algorithm 1** *Goldberg-Tarjan Maximum-Flow*

At Step 0, we construct sequences of edges, initialize a preflow and a labeling function, and construct a sequence of active locations, as listed as follows.

(i) For each $v \in V$, we construct a sequence $E_v$ of the edges to which $v$ is incident. Give an ordering to $E_v$ according to $\pi$.

(ii) We initially set up a preflow $\widetilde{f}$ that is equal to the route capacity on each route leaving the base and zero on all other routes, and set up a labeling function $d$ that is equal to $|V|$ on the base and zero on all other locations.

(iii) We construct a sequence $Q$ of all active locations under $\widetilde{f}$. Give an ordering to $Q$ according to $\pi$.

At Step $k = 1, 2, \cdots$, we modify a preflow, a labeling function and a sequence of active locations repeatedly. From Step $k - 1$, we inherited $\widetilde{f}$, $d$, and $Q$.

(i) If the sequence $Q$ is empty, this algorithm terminates. We return $\widetilde{f}$. If not, we remove the location $v$ on the front of $Q$. We make the first edge in $E_v$ the current edge of $v$.

(ii) Let $\{v, w\}$ be the current edge of $v$.

(iii) If $push(v, w)$ is applicable, then we apply $push(v, w)$. If $w$ becomes active during the operation, then we add $w$ to the rear of $Q$.

(iv) If $e(v; \widetilde{f}) = 0$, then we go to the next step. If not, then we go to (v).

(v) If $push(v, w)$ is not applicable and $\{v, w\}$ is not the last edge in $E_v$, then we make the next edge in $E_v$ the current edge of $v$. We go to (ii).

(vi) If $push(v, w)$ is not applicable and $\{v, w\}$ is the last edge in $E_v$, then we make the first edge in $E_v$ the current edge of $v$. We apply $relabel(v)$ and add $v$ to the rear of $Q$. We go to the next step. □

Goldberg and Tarjan [3] show that the preflow returned from Algorithm 1 is a maximum flow in $G$. They also provide an algorithm to find a minimum cut in a transportation network by modifying their maximum-flow algorithm.

For each $v \in V \setminus \{x, y\}$ and each preflow $\widetilde{f}$, we say that location $v$ is strongly active under $\widetilde{f}$ if $d(v) < |V|$ and $e(v; \widetilde{f}) > 0$. For each preflow $\widetilde{f}$, let $G_{\widetilde{f}} = (V, A_{\widetilde{f}}, x, y, c)$ be the residual transportation network under $\widetilde{f}$ where $A_{\widetilde{f}}$ is the set of residual routes under $\widetilde{f}$.

The necessary modification is simple: The Goldberg-Tarjan Minimum-Cut algorithm requires a location to be strongly active rather than to be active. When the modified algorithm terminates, the flow excess $e(y; \widetilde{f})$ of $y$ under $\widetilde{f}$ is the value of a maximum flow, and the cut $K, \overline{K}$ such that $\overline{K}$ contains exactly those locations from which $y$ is reachable in $G_{\widetilde{f}}$ is a minimum cut.

The following example shows how to find a maximum flow and a minimum cut in a transportation network by using the Goldberg-Tarjan algorithms.

**Example 2** *Let $G = (V, A, x, y, c)$ be the transportation network considered in Example 1. Let $\pi = (x, v_1, v_2, y)$ be an ordering of $V$. First, we find a maximum flow in $G$. At Step 0, we construct sequences of edges, initialize a preflow and a labeling function, and construct a sequence of active locations. Thus, $E_x = (\{x, v_1\}, \{x, v_2\})$, $E_{v_1} = (\{v_1, x\}, \{v_1, v_2\}, \{v_1, y\})$, $E_{v_2} = (\{v_2, x\}, \{v_2, v_1\}, \{v_2, y\})$, $E_y = (\{y, v_1\}, \{y, v_2\})$, $\widetilde{f}(x, v_1) = 4$, $\widetilde{f}(x, v_2) = 1$, $\widetilde{f}(v_1, v_2) = 0$, $\widetilde{f}(v_2, y) = 0$, $\widetilde{f}(y, v_1) = 0$, $d(x) = 4$, $d(v_1) = d(v_2) = d(y) = 0$, and $Q = (v_1, v_2)$.*

*At Step 1, since all of $push(v_1, x)$, $push(v_1, v_2)$, and $push(v_1, y)$ are not applicable, we apply $relabel(v_1)$. We remove $v_1$ on the front of $Q$ and add $v_1$ to the rear of $Q$. Thus, $d(v_1) = 1$ and $Q = (v_2, v_1)$.*

*At Step 2, since all of $push(v_2, x)$, $push(v_2, v_1)$, and $push(v_2, y)$ are not applicable, we apply $relabel(v_2)$. We remove $v_2$ on the front of $Q$ and add $v_2$ to the rear of $Q$. Thus, $d(v_2) = 1$ and $Q = (v_1, v_2)$.*

*At Step* $3$*, since all of* $push(v_1, x)$*,* $push(v_1, v_2)$*, and* $push(v_1, y)$ *are not applicable, we apply* $relabel(v_1)$*. We remove* $v_1$ *on the front of* $Q$ *and add* $v_1$ *to the rear of* $Q$*. Thus,* $d(v_1) = 2$ *and* $Q = (v_2, v_1)$*.*

*At Step* $4$*, since* $push(v_2, y)$ *is applicable,* $e(v_2; \widetilde{f}) = 1$*, and* $z(v_2, y; \widetilde{f}) = 5$*, we apply* $push(v_2, y)$ *by sending* $\delta = \min\{e(v_2; \widetilde{f}), z(v_2, y; \widetilde{f})\} = 1$ *unit of flow from* $v_2$ *to* $y$*. We remove* $v_2$ *on the front of* $Q$*. Thus,* $\widetilde{f}(v_2, y) = 1$ *and* $Q = (v_1)$*.*

*At Step* $5$*, since* $push(v_1, v_2)$ *is applicable,* $e(v_1; \widetilde{f}) = 4$*, and* $z(v_1, v_2; \widetilde{f}) = 2$*, we apply* $push(v_1, v_2)$ *by sending* $\delta = 2$ *units of flow from* $v_1$ *to* $v_2$*. Since* $v_1$ *is still active under the modified preflow but* $push(v_1, y)$ *is not applicable, we apply* $relabel(v_1)$*. We remove* $v_1$ *on the front of* $Q$ *and add* $v_2$ *and* $v_1$ *sequentially to the rear of* $Q$*. Thus,* $\widetilde{f}(v_1, v_2) = 2$*,* $d(v_1) = 5$*, and* $Q = (v_2, v_1)$*.*

*At Step* $6$*, since* $push(v_2, y)$ *is applicable,* $e(v_2; \widetilde{f}) = 2$*, and* $z(v_2, y; \widetilde{f}) = 4$*, we apply* $push(v_2, y)$ *by sending* $\delta = 2$ *units of flow from* $v_2$ *to* $y$*. We remove* $v_2$ *on the front of* $Q$*. Thus,* $\widetilde{f}(v_2, y) = 3$ *and* $Q = (v_1)$*.*

*At Step* $7$*, since* $push(v_1, x)$ *is applicable,* $e(v_1; \widetilde{f}) = 2$*, and* $z(v_1, x; \widetilde{f}) = 4$*, we apply* $push(v_1, x)$ *by sending* $\delta = 2$ *units of flow from* $v_1$ *to* $x$*. We remove* $v_1$ *on the front of* $Q$*. Thus,* $\widetilde{f}(x, v_1) = 2$ *and* $Q = \emptyset$*.*

*At Step* $8$*, since* $Q$ *is empty, this algorithm terminates. We return* $\widetilde{f}$*. Therefore,* $\widetilde{f}(x, v_1) = 2$*,* $\widetilde{f}(x, v_2) = 1$*,* $\widetilde{f}(v_1, v_2) = 2$*,* $\widetilde{f}(v_2, y) = 3$*, and* $\widetilde{f}(y, v_1) = 0$*.*

*We next find a minimum cut in* $G$*. Up to Step* $4$*, the modified algorithm yields the same preflow, labeling function, and sequence of strongly active locations as the maximum-flow algorithm.*

*At Step* $5$*, since* $push(v_1, v_2)$ *is applicable, we apply* $push(v_1, v_2)$ *by sending* $2$ *units of flow from* $v_1$ *to* $v_2$*. Since* $v_1$ *is active under the modified preflow but* $push(v_1, y)$ *is not applicable, we apply* $relabel(v_1)$*. Since* $d(v_1) = 5 > 4 = |V|$*,* $v_1$ *is not strongly active. We remove* $v_1$ *on the front of* $Q$ *and add* $v_2$ *to the rear of* $Q$*. Thus,* $\widetilde{f}(v_1, v_2) = 2$*,* $d(v_1) = 5$*, and* $Q = (v_2)$*.*

*At Step* $6$*, since* $push(v_2, y)$ *is applicable, we apply* $push(v_2, y)$ *by sending* $2$ *units of flow from* $v_2$ *to* $y$*. We remove* $v_2$ *on the front of* $Q$*. Thus,* $\widetilde{f}(v_2, y) = 3$ *and* $Q = \emptyset$*.*

*At Step* $7$*, since* $Q$ *is empty, the modified algorithm terminates. Since* $e(y; \widetilde{f}) = 3$*, the value of a maximum flow is* $3$*. Since* $A_{\widetilde{f}} = \{(v_2, y), (y, v_1)\}$*, the cut* $K, \overline{K}$ *such that* $K = \{x, v_1\}$ *and* $\overline{K} = \{v_2, y\}$ *is a minimum cut.* □

# 5 Concluding remarks

In this paper we analyze strategic interaction between a terrorist organization and a security agency. The terrorist organization sets up a base and aims to damage a target through a transportation network. The security agency aims to protect the target from terrorism and effectively operate the transportation network. We find an optimal security policy under which the security agency can protect the target from devastating terrorism and effectively operate the network. To understand how the terrorist organization commits terrorism under the optimal security policy, we find a class of subgame perfect equilibria of this model.

One possible extension of this model is to allow multiple bases and targets. Another possible extension is to analyze strategic interaction between terrorist organizations and security agencies under incomplete information when the agencies do not know the terrorists' whereabouts. We hope to address these issues in our future research.

# References


[1] Baccara, M., Bar-Isaac, H., 2008. How to organize crime. Review of Economic Studies 75, 1039–1067.

[2] Ford, L. R., Fulkerson, D. R., 1956. Maximal flow through a network. Canadian Journal of Mathematics 8, 399–404.

[3] Goldberg, A. V., Tarjan, R. E., 1988. A new approach to the maximum-flow problem. Journal of the Association for Computing Machinery 35, 921–940.

[4] Goyal, S., Vigier, A., 2008. Robust networks, working paper.

[5] Granot, D., Granot, F., 1992. On some network flow games. Mathematics of Operations Research 17, 792–841.

[6] Hong, S., 2008. Hacking-proofness and stability in a model of information security networks, working paper.

[7] Kalai, E., Zemel, E., 1982. Totally balanced games and games of flow. Mathematics of Operations Research 7, 476–478.

[8] Potters, J., Reijnierse, H., Biswas, A., 2006. The nucleolus of balanced simple flow networks. Games and Economic Behavior 54, 205–225.

[9] Reijnierse, H., Maschler, M., Potters, J., Tijs, S., 1996. Simple flow games. Games and Economic Behavior 16, 238–260.