Monitoring in Selfish Routing Games

I consider a model of nonatomic selfish routing with agents of different types and a network operator. In particular, I suppose that some fixed fraction of the commuters are either in violation of a traffic contract or are unwanted by the network operator, who in turn has the ability and authority to impose additional costs via fines on these types of agents. This model is motivated by several situations that arise in transportation systems. The primary applications are in fare evasion in public transportation networks, the location of mobile weigh stations for the monitoring of freight transportation, and the location of law-enforcement for traffic violations. In this paper, I characterize properties of equilibria in terms of the potential externalities that can arise as a result of the inclusion of fines in the presence of heterogeneous agents. A discussion of the network topologies in which the inclusion of a revenue maximizing network operator cannot harm those not subject to fines at an equilibrium follows.

1 INTRODUCTION

Recently there has been a growing literature on network interdiction and pricing games, wherein a player or group of players seek to interdict along or increase the cost of paths used by agents (attackers) whose objective is to traverse a network. Several papers in this literature [5, 9] consider models without congestion where a single network operator places fines on edges in a network in order to maximize revenue. On the other hand, models of network pricing [1, 4] that include congestion often consider agents with homogeneous cost functions. While the contributions of these papers and other models of network interdiction are interesting in their own right, the motivations behind these models often involve situations in which honest agents travel alongside those being interdicted. This paper seeks to address how the inclusion of security to interdict a fraction of the flow affects the congestion experienced by the honest agents traversing the network.

The primary applications are in fare evasion in public transportation networks, the location of mobile weigh stations for the monitoring of freight transportation, and the location of law-enforcement and speed traps for traffic violations. While other applications, such as the location of DUI checkpoints and the implementation of random checkpoints to monitor borders, may seem relevant as well, we will see that the model described below excludes certain assumptions that are significant in these situations. The model of this paper is stylized, and attempting to make statements about more general situations of network pricing and interdiction would call for additional modeling assumptions in order to justify them.

1.1 Related Literature

This paper is primarily related to the literature on pricing and fines in network routing games. The most related papers to this one are Correa et al. (2017)[5] and Borndorfer et al. (2015)[9]. Correa et al. (2017)[5] consider several variants on a leader-follower framework for fare inspection and fines in public transportation. In their model, the split of agent types is determined endogenously by the fare, fines and inspection strategy of the leader. The model presented in this paper assumes that the masses of evaders and honest agents are exogenously given. This model also assumes that the fine received upon being caught is exogenously given, and that the network authority issuing fines to evaders does not have a first-mover advantage. At the same time, I allow for congestion effects to be present within the network since the primary interest of this paper is to determine the latency externalities experienced by honest agents as a consequence of the inclusion of fines for evaders. In a similar vein, Borndorfer et al. (2015)[9] consider a model of network spot-checking games without congestion in which inspectors are to be distributed across a network to hand out fines to agents evading fares. They show that equilibria can be computed using linear programming and place bounds on the difference in revenue obtained between a setting in which the network authority can move first and when the game is simultaneous moves, referred to as the price of spite. The monopoly setting of Acemoglu and Ozaglar (2007)[1] is also somewhat similar to the model in this paper, but they restrict attention to parallel links and assume that edge costs are homogeneous among players.

This paper is also related to the literature on tolling in routing games with heterogeneous agents. In particular, one can view the agents in this model as trading off latency and money in different ways as in Cole, Dodis and Roughgarden (2003)[11]. In this model, I consider an extreme case in which one group, the honest agents, do not consider tolls at all when making routing decisions (this corresponds to $\alpha = 0$ for these agents in their model). Furthermore, the goal of the fines in this model are to maximize revenue rather than induce optimal flows. The heterogeneous agents models of Cole, Lianeas, and Nikolova (2018)[10] and Meir and Parkes (2018) are also quite similar to this paper. The two types of agents in this model can indeed be viewed as a special case of their models, but the idiosyncratic component of θ_b type preferences are endogenously determined by the network operator in this paper.

A number of other papers consider a graph-theoretic approach to the inefficiencies that can arise in selfish routing games. In particular, we will make use of the definitions and results from Chen, Diao, and Hu (2016) and Cenciarelli, Gorla, and Salvo (2018) in the later sections of this paper.

2 MODEL

I consider a finite, connected, directed multigraph $G = (V, E)$ and a source-sink pair (s,t) with $s, t \in V$. A source node, also called an origin, is the starting point for some set of agents. A sink, also called a destination, is a terminal node for a set of agents. An agent with source-sink pair (s, t) seeks to route from node s to node t on a shortest path. I will restrict attention to a single-source single-sink setup for this paper.

Definition 1. A *path* from node s to node t is a sequence of edges $\{e_1, e_2, e_3, ..., e_n\}$ and connecting nodes $\{s, v_1, v_2, ..., v_{n-1}, t\}$ such that $e_1 = (s, v_1)$, $e_j = (v_{j-1}, v_j)$ for $j \in 2, 3, ..., n-1$ and $e_n = (v_{n-1}, t)$. That is, a path is a sequence of edges and intermediate nodes that connects node s to node t in the graph G. A path is *simple* if none of the nodes in the sequence are repeated. I will restrict attention to *simple* paths in this paper.

Denote the set of simple paths from source s to sink t by P. I will suppose that total mass 1 of agents seek to route from source to sink. When agents route from source to sink along the paths available to them, one obtains a flow f on the graph G.

Definition 2. A *flow* on graph G is a function $f : P \longrightarrow \mathbb{R}_+$ that associates to each simple path connecting a source-sink pair a mass of agents routed along that path. The function f must satisfy

$$\sum_{p \in P} f(p) = 1$$

The amount of flow on an edge is given, with some abuse of notation, by

$$f(e) := \sum_{\{p:\, e \in p\}} f(p)$$

where p is a path from a source to a sink. Associated with each edge is a latency function $C_e : \mathbb{R}_+ \longrightarrow \mathbb{R}_+$ providing the latency to an agent for using edge e as a function of the flow on edge e. I will assume that C_e is nondecreasing and continuous for every edge e. Under flow f, an agent taking path p incurs a latency cost

$$C_p(f) = \sum_{e \in p} C_e(f(e))$$

This cost is the same for any agent taking this path. There are three kinds of agents in this model. There are agents of type θ_g, "good" types, which are expected cost minimizers and seek only to route from source to sink on a path of lowest cost. There are also agents of type θ_b, "bad" types, which are also expected cost minimizers. Agents of type θ_b have a similar objective as agents of type θ_g, but with one difference. When an agent of type θ_b passes through an edge which I will call "secure", the agent incurs a cost $\kappa \in \mathbb{R}_{++}$. Thus the cost to these agents of choosing any path is given by the cost that would be incurred by a θ_g type, with an additional cost κ of passing through a secure edge if one exists along the path. Agents of type θ_b are "nonadaptive" using the terminology of Correa et al. (2017) in the sense that paths are chosen ex ante and are not revised by θ_b types if they are caught during travel.The set of secure edges is determined by a single agent, referred to as security. I think of the fine level κ to be fixed and exogeneous, with security randomizing over edges in the digraph with the objective of receiving as much revenue from θ_b types as possible. I suppose that security chooses one edge in expectation.

This is a simultaneous moves game. I suppose that the relative weights of θ_g and θ_b types are described by $w(\theta_g) \in [0, 1]$ and $w(\theta_b) \in [0, 1]$ respectively, with the requirement that $w(\theta_g)+w(\theta_b) = 1$. I will restrict attention to deterministic flows in this paper, requiring that agents of type θ_g or θ_b are choosing paths with probability 1 (playing pure strategies)[1]. A pure strategy for security is to secure a single edge $e \in E$. Security's set of mixed strategies is then the set of all probability distributions over edges in G, here denoted by ΔE. Outcomes of this game are then described by a triple (f_g, f_b, a_s), where f_g is a flow of value $w(\theta_g)$, f_b is a flow of value $w(\theta_b)$, and $a_s \in \Delta E$ is a randomization over edges for security. The probability that edge e is secured is denoted by $a_s(e)$. I will denote the total flow by f, so that $f = f_g + f_b$.

Given an outcome (f_g, f_b, a_s), the cost to a θ_g types to taking path p is just the latency:

$$C^g_p(f) = \sum_{e \in p} C_e(f(e))$$

The cost to a θ_b types to taking path p is latency plus expected fine, given by

$$C^b_p(f, a_s) = \sum_{e \in p} (C_e(f(e)) + a_s(e)\kappa)$$

The payoff to security is given by

$$\pi(f, a_s) = \sum_{e \in E} \kappa a_s(e) f_b(e)$$

2.1 Equilibria

I consider Nash equilibria of this game in which θ_g and θ_b types play pure strategies.[2] Throughout, I refer to an equilibrium of this form as a *Security Equilibrium.*

Definition 3. A *Security Equilibrium* is a triple (f_g, f_b, a_s) such that:

- for any path p such that $f_g(p) > 0$

$$\sum_{e \in p} C_e(f(e)) \leq \sum_{e \in \hat{p}} C_e(f(e)) \;\; \forall \hat{p} \in P$$

[1] One can do the usual "purification" of commuter mixed strategies when latency functions are linear.

[2] Existence of equilibria when allowing for mixed strategies is an application of the results of Mas-Colell (1984).[7]

- for any path p such that $f_b(p) > 0$

$$\sum_{e \in p} C_e(f(e)) + a_s(e)\kappa \leq \sum_{e \in \hat{p}} C_e(f(e)) + a_s(e)\kappa \quad \forall \hat{p} \in P$$

- $\sum_{p \in P} f_g(p) = w(\theta_g)$
- $\sum_{p \in P} f_b(p) = w(\theta_b)$
- $\forall p \in P,\ f_g(p), f_b(p) \geq 0$
- a_s solves the following linear program given the flow f_b

$$max_{\{a_s \in \triangle E\}} \sum_{e \in E} a_s(e) f_b(e)$$

$$s.t. \quad \sum_{e \in E} a_s(e) = 1 \qquad (LP)$$

$$a_s(e) \geq 0 \ \forall e \in E$$

One can characterize security equilibria of this game as solutions to a collection of convex programs. In particular, one can use the potential function representation of the games induced by the decisions of the other players to obtain the best responses of the θ_g and θ_b type agents. Beckmann (1956)[6] was the first to utilize the potential function to characterize equilibria in traffic problems. The method of using potential functions in games induced by the decisions of other types of players has also been used before (see Babaioff, Kleinberg, and Papadimitrou (2007)[8] for one example). In particular, given f_b, the best response of the θ_g type agents are the solutions to the following convex program:

$$min_{\{f_g\}} \sum_{e \in E} \int_0^{f_g(e)} C_e(f_b(e) + t)dt$$

$$s.t. \quad \sum_{p} f_g(p) = w(\theta_g) \qquad (CP1)$$

$$f_g(p) \geq 0 \quad \forall p \in P$$

One can also use a similar approach to obtain best responses for agents of type θ_b given f_g and a_s. The best response for θ_b types is obtained by solving the following convex program:

$$min_{\{f_b\}} \sum_{e \in E} \int_0^{f_b(e)} C_e(f_g(e) + t) + a_s(e)\kappa \ dt$$

$$s.t. \quad \sum_{p} f_b(p) = w(\theta_b) \qquad (CP2)$$

$$f_b(p) \geq 0 \quad \forall p \in P$$

Lemma 2.1. *If a tuple (f_g, f_b, a_s) is such that:*

- *given f_b, f_g solves (CP1) defined above*
- *given f_g and a_s, f_b solves (CP2) defined above*
- *given f_b, a_s solves (LP)*

then (f_g, f_b, a_s) constitutes a secrutiy equilibrium of this game.

Proof. To prove the result, I'll show that that the optimality conditions of the above convex programs imply the conditions laid out in the definition of security equilibria. Consider the optimality conditions of (CP1) given f_b, in particular

$$\sum_{e \in p} C_e(f(e)) = -\lambda + \gamma_p \ \ \forall p$$

$$\sum_{p} f_g(p) = w(\theta_g)$$

$$f_g(p) \geq 0 \ \ \forall p \in P$$

$$\gamma_p f_g(p) = 0 \ \ \forall p \in P$$

$$\gamma_p \geq 0 \ \ \forall p \in P$$

where γ_p are the Lagrange multipliers associated with the non-negativity constraints of $f_g(p)$ and λ is the Lagrange multiplier associated with the equality constraint. Note that constraint qualification is satisfied. By the complementary slackness conditions, if $f_g(p) > 0$ then it must be that $\gamma_p = 0$. This immediately implies (from the first order condition) that if $f_g(p) > 0$ then

$$\sum_{e \in p} C_e(f(e)) = -\lambda \leq \sum_{e \in \hat{p}} C_e(f(e)) \ \ \forall \hat{p} \in P$$

Which is the same as the first condition for security equilibrium.

Let's now consider solutions to (CP2) given f_g. The optimality conditions for (CP2) are:

$$\sum_{e \in p} C_e(f(e)) + a_s(e)\kappa = -\lambda + \gamma_p \ \ \forall p \in P$$

$$\sum_{p} f_b(p) = w(\theta_b)$$

$$f_b(p) \geq 0 \ \ \forall p \in P$$

$$\gamma_p f_b(p) = 0 \ \ \forall p \in P$$

$$\gamma_p \geq 0 \ \ \forall p \in P$$

where γ_p are the Lagrange multipliers associated with the non-negativity constraints and λ is the Lagrange multiplier associated with the equality constraint. Note that constraint qualification is satisfied. By the complementary slackness conditions, if $f_b(p) > 0$ then $\gamma_p = 0$. We then obtain that if $f_b(p) > 0$,

$$\sum_{e \in p} C_e(f(e)) + a_s(e)\kappa = -\lambda \leq \sum_{e \in \hat{p}} C_e(f(e)) + a_s(e)\kappa \ \ \forall \hat{p} \in P$$

which is the second condition for security equilibrium. Finally, note that if a_s solves (LP) given f_b, then the final condition of security equilibrium is satisfied. □

Evidently then existence of security equilibrium boils down to showing existence of such a triple. The next result uses standard methods from game theory to show existence of security equilibria.

Proposition 2.2. *Suppose that the latency on each edge C_e is a nonnegative, increasing, continuous function of the load on that edge. Then the game defined above has a Wardrop equilibrium.*

Proof. Define by $F(w(\theta_g))$ to be the collection of all flows on paths from source to sink with value $w(\theta_g)$. Define the analogous collection of flows $F(w(\theta_b))$. These are finite dimensional simplices. Recall that $\triangle E$ is the collection of all probability distributions over edges. It is also a finite dimensional simplex. Thus $F(w(\theta_g)) \times F(w(\theta_b)) \times \triangle E$ is closed and convex. Consider the mapping $\Phi : (F(w(\theta_g)) \times F(w(\theta_b)) \times \triangle E) \longrightarrow 2^{\{F(w(\theta_g)) \times F(w(\theta_b)) \times \triangle E\}}$ defined by

$$\Phi(f_g, f_b, a_s) = B_g(f_b) \times B_b(f_g, a_s) \times B_s(f_b)$$

where $B_g(f_b))$ is the set of solutions to (CP1) given f_b, $B_b(f_g, a_s)$ is the set of solutions to (CP2) given f_g and a_s, and $B_s(f_b)$ is the set of solutions to (LP) given f_b. To complete the proof, follow the usual steps of verifying that Φ satisfies the conditions of Kakutani's fixed point theorem by using the properties of solutions to the above optimization problems. It is easy to see that the solutions to each problem are nonempty (Weierstrass), convex (by convexity of the objective function), compact (since the objective function is continuous and the constraint set is compact). We get upper-hemicontinuity of Φ by applying Berge's (Minimum) Theorem. □

It turns out that one can collapse the two potential functions for θ_g and θ_b types into a single potential function by modifying the graph G appropriately. A consequence of this is the following lemma.

Lemma 2.3. *Suppose that the latency function on each edge is a strictly increasing, nonnegative, and continuous function of the load on that edge. Then for a fixed decision of security a_s, the payoffs of any of the mutual best responses of θ_g and θ_b types are unique. That is, for a fixed a_s and κ, the (f_g, f_b) that are such that*

- *given f_b, f_g solves (CP1) as above*
- *given f_g and a_s, f_b solves (CP2) as above*

induce payoffs that are unique for the θ_g and θ_b types.

Proof. The proof relies on a useful reformulation of the problem.[3] For each edge $e = (v_i, v_j)$ in the graph G, insert an additional node v_{ij} and create a directed edge from v_i to v_{ij} and another from v_{ij} to v_j. Remove the edge $e = (v_i, v_j)$. Finally, construct an additional edge from v_{ij} to v_j. We will allow all types to pass through the edge (v_i, v_{ij}), and the cost function for this edge is equal to the cost function of the deleted edge $e = (v_i, v_j)$. Now, for the two edges from v_{ij} to v_j, one will have constant cost of 0 and permits only θ_g types to pass. The other edge has constant cost equal to the expected fine that was placed on edge $e = (v_i, v_j)$, and permits only bad types to pass. Performing this change to every edge yields a new graph, call this graph $\hat{G} = (\hat{V}, \hat{E})$. Now, using the new graph and the additional constraints imposed on flows of each type, we collapse the two potential functions in Lemma 2.1 into a single potential function. In particular, we have that security equilibria are exactly the flows that satisfy security's linear program in addition to the potential function below. In what follows, we define P_g as the set of paths that can be used by θ_g types in the new graph, and P_b the set of paths that can be used by θ_b types in the new graph.

$$\begin{aligned} min_{\{f_g, f_b\}} \sum_{e \in \hat{E}} \int_0^{f(e)} & C_e(t) + a_s(e)\kappa dt \\ s.t. \qquad \sum_{p \in P_g} & f_g(p) = w(\theta_g) \\ \sum_{p \in P_b} & f_b(p) = w(\theta_b) \end{aligned}$$

[3] This reformulation was suggested by Eva Tardos.

$$f_b(p), f_g(p) \geq 0, \quad \forall p \in P$$

Note that on any given edge the objective function will take on one of three forms, in particular one of $C_e(t)$, $a_s(e)\kappa$, or 0. Using the fact that this is a convex program when latency functions are nondecreasing, we obtain that θ_g and θ_b types have unique payoffs given the decision of security. [4]

□

An important remark about the previous lemma is that this implies that a Stackelberg setting in which the network operator moves first and chooses a randomization is well-defined as long as latency functions are strictly increasing on every edge. That is, a given randomization for the network operator will result in a unique level of revenue. To see why, note that the (strict) convexity of the potential function and the fact that latency functions are strictly increasing implies that not only the latency (and fine) on any given edge in $\hat{G}$ is unique for any solution to the above convex program, but also the exact flow on any given edge in G. This implies uniqueness of the total flow at the resulting equilibrium. Note that the exact composition of this total flow in terms of θ_g and θ_b can vary, but I claim that it does not vary in a way that is payoff-relevant. With uniqueness of total flow established, this pins down a unique level of revenue using the uniqueness of payoffs to θ_b types.

3 RESULTS

This section is devoted to examining the properties of security equilibria and how they relate to the equilibrium of the game without security. In particular, we are interested in comparing the latency incurred by θ_g types at equilibria of the security game as above to the latency incurred by all agents at equilibria of the regular selfish routing setting on the same graph with the same cost functions. As a preliminary, Wardrop equilibria[13] of the regular selfish routing game in this setting are given by solutions to the following convex program:

$$min_{\{f\}} \sum_{e \in E} \int_0^{f(e)} C_e(t)dt$$

$$s.t. \quad \sum_p f(p) = 1$$

$$f(p) \geq 0 \;\; \forall p \in P$$

Nash equilibria of the selfish routing game have the property that all paths with positive flow have weakly less latency than any other path. That is if $f(p) > 0$, then $\sum_{e \in p} C_e(f(e)) \leq \sum_{e \in \hat{p}} C_e(f(e))$ for any $\hat{p} \in P$. For an explanation of the properties of Nash equilibria in nonatomic selfish routing games, see Roughgarden and Tardos (2001) [12].

I begin by discussing some basic properties of security equilibria. Throughout this section, I will use the notation of:

- $P_s^g(f)$: The set of paths such that if $p \in P_s^g(f)$, then $\sum_{e \in p} C_e(f(e)) \leq \sum_{e \in p'} C_e(f(e))$ for all p'.
- $P_s^b(f)$: The set of paths such that if $p \in P_s^b(f)$, then $\sum_{e \in p} C_e(f(e)) + a_s(e)\kappa \leq \sum_{e \in p'} C_e(f(e)) + a_s(e)$ for all p'.
- $E_s(a_s)$: The set of edges such that $a_s(e) > 0$

I would like to point out the difference between P_g, the paths available to θ_g types in the extended graph $\hat{G}$ used in the proof of Lemma 2.3, and $P_s^g(f)$ defined here. The next lemma simply states that θ_b types all experience the same total expected costs at equilibrium, and that this expected cost is strictly higher than θ_b types. The proof is straightforward.

[4] The existence of a potential function for this game is a consequence of Theorem 4.4 of Farokhi et al. (2014)[2].

Lemma 3.1. *At any security equilibrium (f_g, f_b, a_s), $\exists \alpha \in (0, 1]$ such that for any $p \in P_s^b(f)$ and any $\hat{p} \in P_s^g(f)$*

$$\sum_{e \in p} C_e(f(e)) + a_s(e)\kappa = \sum_{e \in \hat{p}} C_e(f(e)) + \alpha\kappa$$

In other words, if $C_b^*(f_g, f_b, a_s)$ denotes the cost experienced by any θ_b types at a security equilibrium, and $C_g^*(f_g, f_b, a_s))$ denotes the cost experienced by any θ_g type at a security equilibrium, then

$$C_b^*(f_g, f_b, a_s) = C_g^*(f_g, f_b, a_s) + \alpha\kappa$$

for some $\alpha \in (0, 1]$.

Proof. Consider any edge $e_s \in E_s$. Using the fact that a_s is a best response, there must be at least one path $p^* \in P_s^b(f)$ and an edge $e^* \in P_s^b(f) \cap E_s(a_s)$. Then the cost of taking path p^* for a θ_b type is

$$\sum_{e \in p^*} C_e(f(e)) + a_s(e)\kappa > \sum_{e \in p^*} C_e(f(e)) \geq \sum_{e \in p} C_e(f(e)) \;\; \forall p : f_g(p) > 0$$

. The result then follows from the fact that all θ_b must be indifferent between any path in $P_s^b(f)$ and all θ_g must be indifferent between any path in $P_s^g(f)$ at the equilibrium (f_g, f_b, a_s). □

This easily implies the following result which characterizes in part the equilibrium assignment of security probabilities.

Corollary 3.2. *Fix a security equilibrium (f_g, f_b, a_s). Consider the subgraph $G(f_g)$ of G formed by the set $P_s^g(f)$ of shortest latency paths. Claim: The set $E_s(a_s)$ of secure edges forms a source-sink cut of $G(f_g)$.*

Proof. It suffices to show that for every path $p \in P_s^g(f)$, $\exists e \in p$ such that $e \in E_s$. Suppose, seeking contradiction, that there is a path $p^* \in P_s^g(f)$ such that there is no secure edge along p^*. Since $p^* \in P_s^g(f)$

$$\sum_{e \in p^*} C_e(f(e)) \leq \sum_{e \in p} C_e f(e)) \;\; \forall p \in P$$

By assumption,

$$\sum_{e \in p^*} C_e(f(e)) + a_s(e)\kappa = \sum_{e \in p^*} C_e(f(e))$$

Now pick some edge $e_s \in E_s$ and a corresponding path $p_b \in P_s^b(f)$ such that $e_s \in p_b$. This implies that

$$\sum_{e \in p_b} C_e(f(e)) + a_s(e)\kappa > \sum_{e \in p_b} C_e(f(e)) \geq \sum_{e \in p^*} C_e(f(e)) + a_s(e)\kappa$$

Since $f_b(p_b) > 0$, we have that not all θ_b are choosing shortest paths according to their cost function. A contradiction of the equilibrium assumption. □

The next result takes the previous corollary a step further by using the objectives of the network operator.

Lemma 3.3. *Consider a security equilibrium (f_g, f_b, a_s) and the corresponding sets of paths $P_s^g(f)$ and $P_s^b(f)$. Claim: $P_s^g(f) \subseteq P_s^b(f)$.*

Proof. Suppose, seeking contradiction, that there is some path $p \in p_s^g(f)$ with $p \notin P_s^b(f)$. Subpath optimality must then imply that there is some subpath $\{e_1, e_2, ...e_n\}$ such that $f_b(e_i) = 0$ for $i = 1, ..., n$ and $e_i \in p$ for $i = 1, ..., n$. Choose a maximal such subpath along p and note that this subpath cannot equal p itself due to corollary 3.2 above. Suppose that this maximal subpath connects the vertices u and v. Then since $p \notin P_s^b(f)$ but $p \in P_s^g(f)$, there must be some edge $e^* \in E_s \cap p$ that lies between u and v along p. But the fact that $e^* \in E_s$ implies that $f_b(e^*) > 0$. Thus we obtain a contradiction. □

In the following results, I will refer frequently refer to f^* as an equilibrium of the selfish routing game without security. I will also use f_g, f_b and f to be θ_g flow, θ_b flow and total flow respectively at a security equilibrium. Note that for the flow f^*, one could in principle split θ_g and θ_b types in any way that results in the same total flow. I will refer to f_b^* and f_g^* to be the particular split of θ_g and θ_b types chosen for an equilibrium flow of the game without security. We have that $f_g^* + f_b^* = f^*$ by construction. One interesting component of the following results is that they hold regardless of how one chooses to split the types. Given a security equilibrium (f_g, f_b, a_s), let $C_g^*(f_g)$ denote the common latency experienced by θ_g types. Similarly, let $C_b^*(f_b)$ be the common cost experienced by θ_b types. Finally, let $C^*(f^*)$ denote the common latency experienced by all agents at an equilibrium of the game without security.

Proposition 3.4. *Suppose that (f_g, f_b, a_s) is a security equilibrium and f^* any equilibrium flow of the game without security (with any split f_g^* and f_b^* of θ_g and θ_b types among paths). Then*

$$\sum_{e \in E} a_s(e) f_b(e) \leq \sum_{e \in E} a_s(e) f_b^*$$

That is to say, security obtains more revenue from security equilibrium randomization a_s under any equilibrium flow of the game without security than in the security equilibrium (f_g, f_b, a_s).

Proof. The proof relies on the collapsed potential function from Lemma 2.3. Using the fact that, given a_s, (f_g, f_b) is a global minimum of

$$\sum_{e \in \hat{E}} \int_0^{f(e)} C_e(t) + a_s(e)\kappa dt$$

in the extended graph $\hat{G} = (\hat{V}, \hat{E})$, one has that

$$\sum_{e \in \hat{E}} \int_0^{f(e)} C_e(t) + a_s(e)\kappa dt \leq \sum_{e \in \hat{E}} \int_0^{f^*(e)} C_e(t) + a_s(e)\kappa dt$$

Which implies that

$$\sum_{e \in P_g \cap P_b} \int_0^{f(e)} C_e(t) dt + \sum_{e \in P_b \setminus P_g} f(e) a_s(e)\kappa \leq \sum_{e \in P_g \cap P_b} \int_0^{f^*(e)} C_e(t) dt + \sum_{e \in P_b \setminus P_g} f^*(e) a_s(e)\kappa \quad (1)$$

On the other hand, the equilibrium flow f^* of the game without security is a global minimum of the potential function

$$\sum_{e \in \hat{E}} \int_0^{f^*(e)} C_e(t) dt$$

Therefore

$$\sum_{e \in P_g \cap P_b} \int_0^{f(e)} C_e(t) dt - \sum_{e \in P^g \cap P^b} \int_0^{f^*(e)} C_e(t) dt \geq 0$$

Combining this fact with inequality (1), we have

$$\sum_{e \in P_b \setminus P_g} a_s(e)\kappa[f(e) - f^*(e)] \leq 0$$

Which completes the proof using the equivalence of payoffs between G and $\hat{G}$. □

The previous result seems to suggest that θ_b types are contorting themselves, at least to some extent, off of equilibrium paths in order to avoid security. This leads to the question of whether or not the costs of θ_g types always improve when one adds security to the game. This question is answered in the positive for series-parallel graphs using known results, but is not true in general.

Proposition 3.5. *Suppose that latency functions are nondecreasing, nonnegative, and continuous functions. Then if G is series-parallel, θ_g types haves weakly less latency at any security equilibrium than at a Nash flow. That is, if (f_g, f_b, a_s) is a security equilibrium and f^* is a Nash flow, then*

$$C_g^*(f_g, f_b, a_s) \leq C^*(f^*)$$

Proof. The proof is an application of Lemma 1 from Cole, Lianeas, and Nikolova (2018). Since the equilibrium flow f^* and the security equilibrium flow f have the same value, there is a path P such that for all $e \in P$, $f^*(e) \geq f(e)$ and $f^*(e) > 0$. Thus we have

$$C_g^*(f) \leq \sum_{e \in P} C_e(f(e)) \leq \sum_{e \in P} C_e(f^*(e)) = C^*(f^*)$$

□

The previous result uses almost no structure of equilibria in the proof. Indeed, it is only using the properties of flows on series parallel graphs and the fact that θ_g types are latency minimizers. This begs the question of which topologies cannot induce any negative externalities for those not subject to fines. Before exploring this, I give a simple result that characterizes weakly dominated edges for the network operator.

Lemma 3.6. *Denote by $P(e_i)$ as the collection of paths that pass through edge e_i. Suppose that for two edges e_1 and e_2 that $P(e_1) \subset P(e_2)$. If $a_s(e_1) > 0$ at a security equilibrium (f_g, f_b, a_s), then $f_b(p) = 0$ for any path $p \in P(e_2) \backslash P(e_1)$. Furthermore, if there is a security equilibrium (f_g, f_b, a_s) in which $a_s(e_1) > 0$, then there is also a security equilibrium $(f_g^{'}, f_b^{'}, a_s^{'})$ such that $a_s^{'}(e_2) > 0$ and in which the payoffs to θ_g and θ_b types are unchanged.*

Proof. The proof is simple. Suppose at an equilibrium (f_g, f_b, a_s) that $a_s(e_1) > 0$. Since $P(e_2)$ contains $P(e_1)$, it must be that $f_b(p) = 0$ for any path $p \in P(e_2) \backslash P(e_1)$ since a_s must be a weak best response for security. For the second statement, simply note that shifting all of the probability from edge e_1 to edge e_2 for security must not change their payoffs by the first part of the lemma and the fact that (f_g, f_b, a_s) is assumed to be an equilibrium with $a_s(e_1) > 0$. Note that this shifting of security must cannot change the payoffs for any paths used by θ_b types, and can only increase the cost of paths not used by θ_b types. Thus θ_b types choose not to switch paths after the change in security, keeping f_g fixed. Finally, the shift in security causes no change in latency to θ_g types. Hence, $f_g^{'} = f_g$ and $f_b^{'} = f_b$. □

One sees here that one can restrict attention to security equilibria in which the network operator uses only undominated edges. That is, edges $e_i \in E$ such that there is no $e_j \in E$ with $P(e_i)P(e_j)$. Using this fact, I begin to look into which network topologies are such that θ_g types cannot be worse off at a security equilibrium for any assignment of cost functions in the class that I consider.

The next example shows that on the familiar Braess graph, θ_g types can be no worse off for any choice of cost functions.

Example 3.7. Suppose G is the Braess graph depicted below. Then for any security equilibrium (f_g, f_b, a_s) and any equilibrium flow f^* of the game without security,

$$C_g^*(f_g, f_b, a_s) \leq C^*(f^*)$$

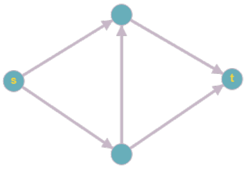

Fig. 1. Braess Graph

Note that this graph has only two undominated edges, both of which lie on the crossover path from s to t. The inefficiencies in the Braess graph can occur when there are agents using both parallel $s - t$ paths, as well as a positive mass using the crossover path. This is a standard example. Note in this model that if θ_g types are using both of these parallel paths, corollary 3.2 implies that both of the undominated edges carry a positive expected fine. This increased cost disincentivizes those subject to the fines, the θ_b types, from inducing this inefficiency. While some mass of agents may still use the crossover path, this configuration of undominated edges guarantees that the latency on a shortest path cannot deteriorate relative to the equilibrium without fines. That is to say, θ_g types cannot be worse off.

A natural question is whether or not the inclusion of security can ever hurt those that are not subject to the fines. The answer is yes, the risk of fines can cause those subject to them to take paths that increase the latency of all shortest paths. The counterexample is simple. All we need to do is modify the Braess graph above so that there exist undominated edges that do not lie on the crossover path.

Example 3.8. Consider the figure below. In this graph, all edges are undominated.

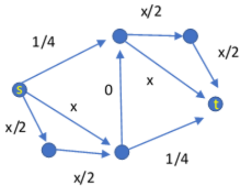

Fig. 2. Modified Braess Graph

Fix $w(\theta_g) = 0.2$, $w(\theta_b) = 0.8$, and $\kappa = \frac{1}{10}$. At the equilibrium without fines, the latency experienced by all agents is $\frac{1}{2}$. Now, there is a security equilibrium in which each edge with constant cost $\frac{1}{4}$ has expected fine $\frac{\kappa}{2}$. At this security equilibrium, a mass of 0.1 θ_g types will traverse each of the upper and lower sections of the graph individually, and not cross over the middle edge. A mass of 0.2 θ_b types will choose to cross over, while a mass of 0.3 θ_b types will follow the upper and lower paths individually and not cross over. This leads to a the latency on fastest paths to be $\frac{3}{10} + \frac{1}{4} = \frac{11}{20} > \frac{1}{2}$.

This example is useful in that if the graph in example 3.8 is embedded in a particular way in some graph G, then we know that we can assign a vector of cost functions to G that result in θ_g types being worse off after the addition of fines. This is made more formal in the next lemma. Before moving on to the lemma, I provide a definition of an $s - t$ paradox from Chen, Diao, and Hu (2016).

Definition 4. We call G an $s - t$ paradox if $G = P_1 \cup P_2 \cup P_3$ is the union of three paths P_1, P_2, and P_3 such that:

- P_1 is an $s - t$ path going through distinct vertices a, u, v, b in this order
- P_2 is an $a - v$ path with $V(P_2) \cap V(P_1) = \{a, v\}$
- P_3 is a $u - b$ path with $V(P_3) \cap V(P_1) = \{u, b\} and V(P_3) \cap V(P_2) = \emptyset$

Lemma 3.9. *Suppose that a graph G contains an $s - t$ paradox $\hat{G}$ with corresponding paths P_1, P_2, P_3 and vertices a, u, v, b. Suppose that G has the following properties:*

- *In addition to P_1, there is another $a - u$ path P_4 that is edge-disjoint from P_1 and such that $V(P_4) \cap V(P_2) = \{a\}$*
- *In addition to P_1, there is another $v - t$ path P_5 that is edge-disjoint from P_1 and such that $v(P_5) \cap V(P_3) = b$*
- *There exist three pairwise edge-disjoint $s - a$ paths and three pairwise edge-disjoint $b - t$ paths.*

Then there is an assignment of latency functions, type-weights, and fine level such that there is an equilibrium of the security game (f_g, f_b, a_s) with

$$C_g^*(f_g, f_b, a_s) > C^*(f^*)$$

Proof. The proof uses the fact that the graph in example 3.8 is a embedded in G, as well as the fact that there are no bottlenecks in key locations. Select three edge-disjoint $s - a$ paths and three edge-disjoint $b - t$ paths. Also determine the relevant paradox paths P_1, P_2, P_3, P_4, and P_5 as above. For any edge not on the aforementioned collection of paths, give a latency function of excessively high constant cost so that these edges will never be used at an equilibrium. What we're left with is essentially a segment that resembles example 3.8 as well as three paths at each end of the segment that allow us to distribute the flow of θ_b types enough so that they do not incur security. The result then follows from example 3.8. □

The conditions from the previous lemma are extremely narrow. Future work will be devoted to tightening this condition and pinning down network topologies in which θ_g types do not experience any negative externalities from the presence of security.

Up until this point, I've considered whether or not the inclusion of security can improve latency for θ_g type agents. One natural question is just how much of an improvement can be made. The next example shows that θ_g types can experience arbitrarily less latency than at the equilibrium without security.

Example 3.10. Let $m \in \mathbb{N}$, $m \geq 2$. Consider the graph given by m-1 parallel links. Suppose that $m - 2$ of these links have constant cost m, and the final link has cost function $C_e(x) = x$. Let $w(\theta_g) = \frac{1}{m}$ and $w(\theta_b) = \frac{m-1}{m}$. The Nash equilibrium in the usual selfish routing game is for all agents to to route over the link with cost function x and obtain a cost of 1. Now consider the security game over the same graph with $\kappa = m - \frac{2}{m}$. Then we can find an equilibrium of the security game in which there are $\frac{1}{m}$ θ_b types on each edge, $\frac{1}{m}$ θ_g types on the edge with cost function x, and security chooses the edge with cost function x with certainty. Then all θ_b types incur cost m, and θ_g types incur cost $\frac{2}{m}$. Allowing m to tend to infinity yields the result that θ_g types can be arbitrarily better off.

Note that this required the mass of θ_g types to be small in order to achieve large improvements in latency for them. It is another application of Lemma 1 of Cole, Lianeas, and Nikolova (2018)[10] that θ_g types can be no better off in a security equilibrium than at an equilibrium of a routing game on the same graph with mass equal to $w(\theta_g)$ (when θ_g types are playing by themselves).

4 CONCLUSION AND FUTURE DIRECTIONS

We have seen above that in several cases, adding security to the game cannot hurt the θ_g types. The primary question that remains is whether or not this is always the case. In principle, one could work toward placing bounds on how much the welfare of θ_g types can differ between equilibria of the two games. These bounds will depend on the relative weights on types as well as the properties of the graph and cost functions. There are many ways to go about this. Perhaps the most promising is to use the fact that the equilibrium flows considered in this game are ϵ-approximate Nash equilibria as defined in Christodoulou et al. (2011)[3]. These equilibria have their own price of anarchy values (the worst-case ratio of social welfare at a Nash equilibrium and a social optimum). While these values take into account the welfare of the entire flow, it's possible that one can still leverage these results based on the value of ϵ and $w(\theta_g)$.

One could also allow security to choose the level of fine κ associated with being caught in order to maximize revenue. As a note, a solution to the problem of choosing a randomization over edges and a κ would not exist if one were to allow for $\kappa \in [0, \infty)$. The reasoning is because there is a positive lower bound to the amount of flow security is guaranteed to catch since they can simply form an s-t cut of the graph G. Then security obtains at least $\frac{\kappa w(\theta_b)}{|C|}$ with certainty, where $|C|$ is the number of edges in the cut. Thus there would be no solution if one were to allow unbounded κ. This is likely the reason network pricing models often include elastic demand or place restrictions on the edges which can be priced. The question of maximizing revenue for $\kappa \in [0, \bar{\kappa}]$ is interesting however. Comparative statics involving the mass of θ_b passing over secure edges at equilibrium would be quite helpful in solving this problem.

Another shortcoming of this model is that I have assumed the presence of security does not slow down θ_g agents at all. This was done for simplicity, but anybody who has waited to be screened at an airport or DUI checkpoint knows that screening agents takes time. On the other hand, ticket inspectors on trains in public transit often impose no additional latency on agents.

Finally, it would be interesting to characterize the differences between the settings of security moving as a leader and the game in its current state of simultaneous moves. Determining how much more revenue can be obtained by security as a result of moving first is interesting. Furthermore, the potential for differences in revenue to security also has implications on the latency externalities experienced by θ_g types in the two settings. Future drafts of this paper will seek to address these questions.

REFERENCES

[1] Daron Acemoglu and Asuman Ozdaglar. 2007. Competition and Efficiency in Congested Markets. *Mathematics of Operations Research* 32, 1 (Feb. 2007), 1–31.

[2] Alexandre Bayen Farhad Farokhi, Walid Krichene and Karl Johansson. 2014. When Do Potential Functions Exist in Heterogeneous Routing Games? https://www.diva-portal.org/smash/get/diva2:695605/FULLTEXT01.pdf

[3] Elias Koutsoupias George Christodoulou and Paul Spirakis. 2011. On the Performance of Approximate Equilibria in Congestion Games. *Algorithmica* 61, 1 (Sept. 2011), 116–140.

[4] Thanasis Lianeas Evdokia Nikolova Jose Correa, Cristobal Guzman and Marc Schroder. [n. d.]. Network Pricing: How to Induce Optimal Flows Under Strategic Link Operators *(EC '18).* ACM.

[5] Vincent Kreuzen Jose Correa, Tobias Harks and Jannik Matuschke. [n. d.]. Fare Evasion in Transit Networks. *Operations Research* 65, 1 ([n. d.]).

[6] C.B. McGuire Martin Beckmann and Christopher Winsten. 1956. *Studies in the Economics of Transportation.* Yale University Press, New Haven, CT.

[7] Andreu Mas-Colell. 1984. On a Theorem of Schmeidler. *Journal of Mathematical Economics* 13 (1984), 201–206.

[8] Robert Kleinberg Moshe Babaioff and Christos Papadimitrou. [n. d.]. Congestion Games with Malicious Players. In *Proceedings of the 8th ACM Conference on Electronic Commerce (EC '07).* ACM, New York, NY.

[9] Guillaume Sagnol Ralf Borndorfer, Julia Buwaya and Elmar Swarat. 2015. Network Spot-Checking Games: Theory and Application to Toll Enforcing in Transportation Networks. *Networks* 65, 4 (Feb. 2015), 312–328.

[10] Thanasis Lianeas Richard Cole and Evdokia Nikolova. [n. d.]. When Does Diversity of User Preferences Improve Outcomes in Selfish Routing?. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI '18).*

[11] Yevgeniy Dodis Richard Cole and Tim Roughgarden. 2003. Pricing Network Edges for Heterogeneous Selfish Users. In *Proceedings of the 35th Annual Symposium on Theory of Computing (STOC '07)*. New York, NY.

[12] Tim Roughgarden and Eva Tardos. [n. d.]. How Bad is Selfish Routing. In *Proceedings of the 41st Annual Symposium on Foundations of Computer Science*. IEEE Computer Society, Washington, DC.

[13] J.G. Wardrop. [n. d.]. Some Theoretical Aspects of Road Traffic Research. In *Proceedings of the Institution for Civil Engineers*. Institution for Civil Engineers, London, UK.