Vulnerability of Transportation Networks to Traffic-Signal Tampering

Aron Laszka
University of California, Berkeley

Bradley Potteiger
Vanderbilt University

Yevgeniy Vorobeychik
Vanderbilt University

Saurabh Amin
Massachusetts Institute of Technology

Xenofon Koutsoukos
Vanderbilt University

ABSTRACT

Traffic signals were originally standalone hardware devices running on fixed schedules, but by now, they have evolved into complex networked systems. As a consequence, traffic signals have become susceptible to attacks through wireless interfaces or even remote attacks through the Internet. Indeed, recent studies have shown that many traffic lights deployed in practice have easily exploitable vulnerabilities, which allow an attacker to tamper with the configuration of the signal. Due to hardware-based failsafes, these vulnerabilities cannot be used to cause accidents. However, they may be used to cause disastrous traffic congestions. Building on Daganzo's well-known traffic model, we introduce an approach for evaluating vulnerabilities of transportation networks, identifying traffic signals that have the greatest impact on congestion and which, therefore, make natural targets for attacks. While we prove that finding an attack that maximally impacts congestion is NP-hard, we also exhibit a polynomial-time heuristic algorithm for computing approximately optimal attacks. We then use numerical experiments to show that our algorithm is extremely efficient in practice. Finally, we also evaluate our approach using the SUMO traffic simulator with a real-world transportation network, demonstrating vulnerabilities of this network. These simulation results extend the numerical experiments by showing that our algorithm is extremely efficient in a microsimulation model as well.

Keywords

Transportation network, attack-resilience, cyber-physical system, tampering attack, traffic model, cyber-security

1. INTRODUCTION

The evolution of traffic signals from standalone hardware devices to complex networked systems has provided us with many benefits, such as reducing wasted time and environmental impact. However, it has also exposed traffic signals to cyber-attacks. While traditional hardware systems were susceptible only to attacks based on direct physical access, modern systems are vulnerable to attacks through wireless interfaces or even to remote attacks through the Internet. A recent case study by Ghena et al. analyzed the security of traffic infrastructure in cooperation with a road agency located in Michigan [9]. This agency operates around a hundred traffic signals, which are all part of the same wireless network, but the signals at every intersection operate independently of the other intersections. The study found three major weaknesses in the traffic infrastructure: lack of encryption for the network, lack of secure authentication due to the use default usernames and passwords on the devices, and vulnerability to known exploits.

Even if every weakness discovered by the investigation of [9] were corrected, it is extremely difficult to prevent all future software vulnerabilities. In addition to the general difficulty of this task, traffic signals pose further challenges, such as long system lifetime and complicated software-upgrade procedures. Consequently, ensuring that there will not be any opportunities for attack during the lifetime of a system is practically impossible, and we must consider the impact of successful attacks.

Due to hardware-based failsafes, compromising a traffic signal does not allow an attacker to set the signal into an unsafe configuration, which could lead to traffic accidents. However, compromising a signal does enable tampering with its schedule, which allows an attacker to

cause disastrous traffic congestions. In order to increase the resilience of transportation networks to tampering attacks, we must first be able to assess how vulnerable a given network is, that is, we must be able to estimate the potential impact of tampering attacks. Since this impact depends on both the transportation network and the schedules of the uncompromised signals in a non-trivial way, vulnerability assessment is a challenging problem.

Building on Daganzo's well-known traffic model [7], we propose an approach for evaluating the vulnerability of a transportation network to traffic signal tampering attacks. We provide theoretical results on the computational complexity of our approach, and introduce an efficient heuristic algorithm for practical application. We show that the proposed algorithm is extremely efficacious using numerical results based on large numbers of randomly generated networks, which mimic real-world road networks. Finally, we evaluate our approach on an actual transportation network using SUMO, a micro-model traffic simulator.

The remainder of this paper is organized as follows. In Section 2, we discuss the traffic and attacker models, and formulate the problem of evaluating vulnerability. In Section 3, we study computational complexity and introduce our heuristic algorithm. In Sections 4 and 5, we evaluate our approach using numerical results based on random networks and simulations based on a real-world network. In Section 6, we give a brief overview of the related work. Finally, in Section 7, we offer concluding remarks and outline directions future work.

2. MODEL

In this section, we introduce the traffic and attacker models on which our approach is built, and then define the vulnerability of transportation networks. A list of symbols used in this paper can be found in Table 1.

2.1 Cell Transmission Model

To model traffic, our approach employs Daganzo's cell transmission model for transportation networks. Here, we provide a summary of this traffic model; for a more detailed discussion, we refer the reader to [7, 8, 19].

In the cell transmission model, the road network is divided into *cells*, which represent homogeneous road segments, and time is divided into uniform *intervals*. The length of a road segment corresponding to a cell is equal to the distances traveled in light traffic by a typical vehicle in one time interval. Each cell has three parameters:

- N_i^t is the maximum number of vehicles that can be present in cell i at time t,
- Q_i^t is the maximum number of vehicles that can flow into or out of cell i during time interval t,
- and δ_i^t is the ratio between the free-flow speed and the backward propagation speed of cell i at time t (see [19] for a detailed explanation). This constant is used to quantify how the speed of traffic decreases as the cell becomes congested, and can model traffic phenomena such as shockwaves.

Table 1: List of Symbols

Symbol	Description
Cell Transmission Model	
x_i^t	number of vehicles in cell i at time t
y_{ij}^t	number of vehicles moving from cell i to cell j at time t
Q_i^t	maximum number of vehicles that can flow into or out of cell i during time interval t
δ_i^t	ratio between free-flow speed and backward propagation speed of cell i at time t
N_i^t	maximum number of vehicles in cell i at time t
$\Gamma(i)$	set of successor cells to cell i
$\Gamma^{-1}(i)$	set of predecessor cells to cell i
d_i^t	demand (inflow) at source cell i during time interval t
Signalized Intersection Model	
p_{ki}^t	inflow proportion from from cell k to signalized intersection i
$\mathcal{S}$	set of signalized intersections
Attacker Model	
B	attacker's budget
$\mathcal{A}$	attack reconfiguring cells in $\hat{\mathcal{S}} \subseteq \mathcal{S}$ to inflow proportions $\hat{p}_{ki}$
$T(\mathcal{A})$	total travel time resulting from attack $\mathcal{A}$

Every cell is connected to one or more other cells (i.e., cells that correspond to consecutive road segments or road segments that are joined by an intersection are connected). The set of cells from which vehicles can move into cell i is called the set of predecessor cells, denoted by $\Gamma^{-1}(i)$; similarly, the set of cells to which vehicles can move from cell i is called the set of successor cells, denoted by $\Gamma(i)$.

At a given time t, the state of the transportation network is given by the vector $\boldsymbol{x}^t$, where the value x_i^t is the number of vehicles in cell i. The traffic model defines how the state of the network $\boldsymbol{x}^t$ evolves over time from an initial state $\boldsymbol{x}^0 = (0, \ldots, 0)'$. In each time interval, for every pair of connected cells i and k, the number of vehicles y_{ik}^t moving from cell i to cell k is determined by the state of both cells (see below).

Based on their connections, the cells can be divided into five types: ordinary cells, diverging cells, merging cells, source cells, and sink cells. Next, we describe how the state x_i^t of each cell i evolves.

Ordinary Cells.

Ordinary cells have only one successor cell and one predecessor cell. For every ordinary cell i and time interval t, the state evolves as follows:

$$x_i^t = x_i^{t-1} + y_{ki}^{t-1} - y_{ij}^{t-1}, \tag{1}$$

where $k \in \Gamma^{-1}(i)$ and $j \in \Gamma(i)$ (note that since cell i is ordinary, k and j are uniquely defined).

The flow y_{ki}^t between ordinary cells k and i is

$$y_{ki}^t = \min\{x_k^t,\ \min\{Q_i^t, Q_k^t\},\ \delta_i^t(N_i^t - x_i^t)\}. \tag{2}$$

According to the above equation, the flow is limited by three terms:

- x_k^t, since the number of vehicles leaving cell k cannot be greater than the number of vehicles present in cell k;
- Q_i^t and Q_k^t by definition;
- and $\delta_i^t(N_i^t - x_i^t)$, which limits the flow as cell i becomes congested.

By setting y_{ki}^t equal to the minimum of these terms, we assume that drivers do not stop without a traffic reason (until they reach their destination).

Diverging Cells.

Diverging cells differ from ordinary cells in that they have multiple successor cells. For every diverging cell i and time interval t,

$$x_i^t = x_i^{t-1} + y_{ki}^{t-1} - \sum_{j \in \Gamma(i)} y_{ij}^{t-1}, \tag{3}$$

where $k \in \Gamma^{-1}(i)$ (note that since cell i is diverging, k is unique).

The inflow y_{ki}^t is the same as in the case of ordinary cells. The outflows y_{ij}^t are subject to the following constraints:

$$\forall j \in \Gamma(i):\ y_{ij}^t \le \min\left(Q_j^t,\ \delta_j^t(N_j^t - x_j^t)\right) \tag{4}$$

$$\sum_{j \in \Gamma(i)} y_{ij}^t \le \min\left(x_i^t,\ Q_i^t\right). \tag{5}$$

Notice that these constraints are analogous to the ones for ordinary cells, with the exception that the limits posed by x_i^t and Q_i^t apply to the sums of the outflows.

Merging Cells.

Merging cells differ from ordinary cells in that they have multiple predecessor cells. For every merging cell i and time interval t,

$$x_i^t = x_i^{t-1} + \left(\sum_{k \in \Gamma^{-1}(i)} y_{ki}^{t-1}\right) - y_{ij}^{t-1}, \tag{6}$$

where $j \in \Gamma(i)$ (note that since cell i is ordinary, j is unique).

The outflow y_{ij}^t is the same as in the case of ordinary cells. The inflows y_{ki}^t are subject to the following constraints:

$$\forall k \in \Gamma^{-1}(i):\ y_{ki}^t \le \min\left(x_k^t,\ Q_k^t\right) \tag{7}$$

$$\sum_{k \in \Gamma^{-1}(i)} y_{ki}^t \le \min\left(Q_i^t,\ \delta_i^t(N_i^t - x_i^t)\right). \tag{8}$$

Again, notice that these constraints are natural extensions of the ones for ordinary cells, except that the limits posed by Q_i^t and $\delta_i^t(N_i^t - x_i^t)$ apply to the sums of the inflows.

Sink and Source Cells.

Sink cells have infinite capacity and allow infinite input flows (i.e., $N_i^t = \infty$ and $Q_i^t = \infty$ for every sink cell i and time interval t); hence, the input flow of sink cell i is $y_{ki}^t = \min\{x_k^t, Q_k^t\}$.

Source cells have infinite capacity but allow only finite output flows (i.e., $N_i^t = \infty$ for every source cell i and time interval t). For every source cell i and time interval t, $x_i^t = x_i^{t-1} + d_i^{t-1} - y_{ij}^{t-1}$, where $j \in \Gamma(i)$ (note that j is unique) and d_i^t is the demand (inflow) at cell i in time interval t (i.e., d_i^t is the number of vehicles entering traffic at cell i in time interval t).

2.1.1 Signalized Intersections

To account for signal control at intersections, we follow Daganzo's proposition [8] and introduce the time-dependent parameter p_{ki}^t controlling the inflow proportions of merging cells. Then, for every signalized merging cell $i \in \mathcal{S}$ and time interval t, the inflows must also satisfy

$$\forall k \in \Gamma^{-1}(i):\ y_{ki}^t \le p_{ki}^t Q_i^t \tag{9}$$

$$\forall k \in \Gamma^{-1}(i):\ y_{ki}^t \le p_{ki}^t \delta_i^t(N_i^t - x_i^t), \tag{10}$$

where $\sum_{k \in \Gamma^{-1}(i)} p_{ki}^t = 1$.

2.1.2 Solving the Traffic Model

In order to solve the traffic model, that is, to find $\boldsymbol{x}^t$ for every $t > 0$, we use Ziliaskopoulos's linear programming approach [19], and formulate the following program:

$$\min \sum_t \sum_i x_i^t \tag{11}$$

subject to Equations (1) to (10), where $x_i^t \in \mathbb{R}_{\ge 0}$ for every time $t > 0$ and cell i, $y_{ki}^t \in \mathbb{R}_{\ge 0}$ for every time $t > 0$ and connected cells k and i, and the number of time intervals is chosen so that $\boldsymbol{x}^t$ reaches $(0, \ldots, 0)$ by the last time interval. Note that we assume fractional x_i^t values, since we are interested in a macro solution, not individual vehicles.

Now, observe that the objective of the above linear program is the sum of the number of vehicles traveling

(i.e., number of vehicles on the road) over time, which is clearly equal to the *total travel time* of all the vehicles. In other words, the above solution assumes that vehicles will travel efficiently (i.e., in a way that minimizes their travel time) given that they have to abide the constraints of the traffic model, including the inflow proportions dictated by the traffic signals. As a consequence, we can use the value of the above linear program – which can be computed efficiently for a given instance – as a measure of network congestion.

2.2 Attacker Model

Next, we introduce our attacker model, which defines the attacker's action space and goal. In our approach, we model attackers who can compromise some of the traffic signals and tamper with their configuration (i.e., schedule), thereby dramatically increasing the total travel time in the transportation network. Furthermore, we consider only relatively short-term scenarios, in which the parameters of the cells and the default (i.e., unattacked) schedules of the traffic signals are constant. Hence, for the remainder of this paper, we will omit the superscript ^{t} from Q_i^t, N_i^t, δ_i^t, and p_{ki}^t.

2.2.1 Action Space

We assume that the attacker is resource bounded, which means that it can compromise at most $B \leq |\mathcal{S}|$ intersections at the same time. Hence, the attacker's action choice is to select a subset of at most B cells from the signalized cells $\mathcal{S}$ and reconfigure the traffic signals at the selected cells. In other words, an attack $\mathcal{A}$ consists of a set $\hat{\mathcal{S}}$ of signalized cells and a set of new inflow proportions $\hat{p}_{ki}$ for the cells in $\hat{\mathcal{S}}$. Formally, an attack $\mathcal{A}$ is a pair

$$\left(\hat{\mathcal{S}}, \left\{\hat{p}_{ki} \;\middle|\; \forall i \in \hat{\mathcal{S}}, k \in \Gamma^{-1}(i)\right\}\right), \tag{12}$$

where $\hat{\mathcal{S}} \subseteq \mathcal{S}$ and $\hat{p}_{ki} \in [0, 1]$.

Due to the attacker's budget constraint, an attack is feasible only if

$$|\hat{\mathcal{S}}| \leq B. \tag{13}$$

Furthermore, we also assume that due to hardware-based failsafes, the signals at an intersection can be reconfigured only to a valid setting. Consequently, the inflow proportions of a feasible attack must sum up to 1 for each merging cell. Formally, an attack $\mathcal{A}$ has to abide the constraint

$$\forall i \in \hat{\mathcal{S}} : \sum_{k \in \Gamma^{-1}(i)} \hat{p}_{ki} = 1. \tag{14}$$

2.2.2 Goal

We assume a worst-case attacker, whose goal is to minimize the network's utility, that is, to maximize the total travel time. For a given attack $\mathcal{A}$, let us denote by $T(\mathcal{A})$ the total travel time computed from the traffic model for the attacked network. In other words, for an attack $\mathcal{A} = \left(\hat{\mathcal{S}}, \{\hat{p}_{ki} | \ldots\}\right)$, let $T(\mathcal{A}) = \sum_t \sum_i x_i^t$ where x_i^t constitute the solution of the traffic model with the inflow proportions of the cells in $\hat{\mathcal{S}}$ replaced by the values $\hat{p}_{ki}$. Then, we can express the attacker's problem as

$$\max_{\mathcal{A}=\left(\hat{\mathcal{S}}, \{\hat{p}_{ki} | \ldots\}\right)} T(\mathcal{A}) \tag{15}$$

subject to

$$|\hat{\mathcal{S}}| \leq B \tag{16}$$

$$\forall i \in \hat{\mathcal{S}} : \sum_{k \in \Gamma^{-1}(i)} \hat{p}_{ki} = 1 \tag{17}$$

where $\hat{\mathcal{S}} \subseteq \mathcal{S}$ and $\hat{p}_{ki} \in [0, 1]$.

2.3 Network Vulnerability

Based on the traffic and attacker models introduced in the preceding sections, we can define the vulnerability of a transportation network in an intuitive way as follows.

Definition 1. The *vulnerability of a transportation network* to traffic-signal tampering attacks is

$$\frac{T(\mathcal{A}) - T}{T}, \tag{18}$$

where $\mathcal{A}$ is the worst-case attack given by our attacker model and T is the total travel time of the network with the default configurations of the traffic signals.

Besides quantifying the vulnerability of a network, our approach also enables us to identify critical traffic signals, which have the greatest impact on traffic congestion and which, therefore, make natural targets for attacks.

Definition 2. A traffic signal (i.e., merging cell) $s \in \mathcal{S}$ is *critical* if $s \in \hat{\mathcal{S}}$ for a worst-case attack $\mathcal{A}$.

Identifying these critical signals is beneficial, since it allows us to locate the most vulnerable elements of a network, which should be strengthened first to increase the resilience of a network. For example, if we have a limited security budget which permits us to replace only a subset of the traffic signals with more secure ones, then we should start with the critical signals $\hat{\mathcal{S}}$.

3. COMPUTATIONAL COMPLEXITY AND HEURISTIC ALGORITHM

To compute the vulnerability of a transportation network, we first have to solve the attacker's problem, that is, we have to find a worst-case attack. However, this problem is challenging, as the number of feasible attacks is an exponential function of B. Consequently,

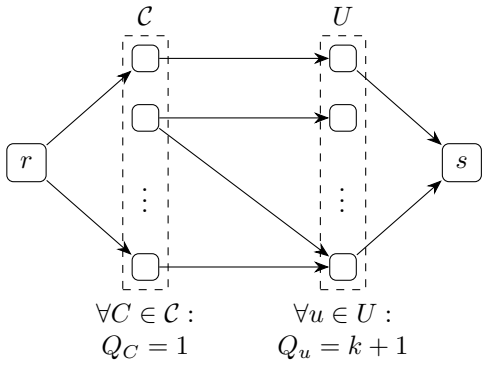

Figure 1: Illustration for the proof of Theorem 1.

an exhaustive search for finding the worst-case attack is impractical, since its running time can quickly become prohibitively high. To meet this challenge, we focus our analysis on the computational aspects of our approach.

3.1 Computational Complexity

We begin our analysis by showing that the attacker's problem (i.e., finding a worst-case attack) is computationally hard. First, we formulate a decision version of the attacker's problem as follows.

Definition 3. Attacker's Decision Problem: Given a transportation network, a budget B, and a threshold travel time T^*, determine if there exists an attack $\mathcal{A}$ satisfying the budget constraint such that $T(\mathcal{A}) > T^*$.

We show that the above problem is computationally hard by reducing a well-known NP-hard problem, the Set Cover Problem, to the above problem.

Definition 4. Set Cover Problem: Given a base set U, a collection $\mathcal{C}$ of subsets of U, and a number k, determine if there exists a subcollection $\mathcal{C}' \subseteq \mathcal{C}$ of at most k subsets such that every element of U is contained by at least one subset in $\mathcal{C}'$.

The following theorem establishes the computational complexity of the attacker's problem.

THEOREM 1. *Attacker's Decision Problem is NP-hard.*

PROOF. Given an instance of the Set Cover Problem (i.e., a set U, a collection $\mathcal{C}$ of subsets, and a number k), we construct an instance of the Attacker's Decision Problem as follows:

- let the transportation network be the following (see Figure 1 for an illustration):
  - there is one source cell r, with $Q_r = k + 1$, $d_r^1 = k + 1$, and $d_r^t = 0$ for $t > 1$;
  - there is one sink cell s;
  - for every element $u \in U$, there is a merging cell u;
  - for every subset $C \in \mathcal{C}$, there is a diverging cell C;
  - each diverging cell C is connected to every merging cell $u \in C$;
  - for every cell i, $N_i = k + 1$ and $\delta_i = 1$;
  - for every merging cell u, $Q_u = k + 1$;
  - for every diverging cell C, $Q_C = 1$;
- let the attacker's budget be $B = |U|$;
- let the threshold travel time be $T^* = 3(k + 1)$.

Clearly, the above reduction can be carried out in time that is polynomial in the size of the Set Cover Problem instance.

It remains to show that the above instance of the Attacker's Decision Problem has a solution $\mathcal{A}$ if and only if the given instance of the Set Cover Problem has a solution $\mathcal{C}'$. Before we proceed to prove this equivalence, notice that the values Q_r, N_i and δ_i for every cell i, and Q_u for every merging cell u will not play any role, since they are high enough to allow any traffic to pass through. Furthermore, since $B = |U|$, the attacker will be able to reconfigure every traffic signal; hence, the attacker's problem is simply to pick the values $\hat{p}_{Cu}$ for every $u \in C$.

First, suppose that there exists a set cover $\mathcal{C}'$ of size at most k. Then, we construct an attack as follows: for every merging cell u, choose one diverging cell C from $\mathcal{C}'$ that is connected to u (if there are multiple, then choose an arbitrary one), and let $\hat{p}_{Cu} = 1$. We have to show that the total travel time in the transportation network is greater than $3(k + 1)$ after the attack. Since the distance between the source cell and the sink cell is 3 hops and there are $k + 1$ vehicles, all the vehicles must move one step closer to the sink in every time interval in order for the total travel time to be at most $3(k + 1)$. However, from the source cell, the vehicles may only move to the cells in $\mathcal{C}'$; otherwise, they would get "stuck" at one of the diverging cells that are not in $\mathcal{C}'$. Consequently, in the second time interval, at most k of the $k + 1$ vehicles may move on, which means that the total travel time has to be greater than $3(k + 1)$.

Second, suppose that there does not exist a set cover $\mathcal{C}'$ of size at most k. Then, we have to prove that there cannot exist an attack which increases the total travel time to more than $3(k+1)$. Firstly, we show that there exists an optimal attack which assigns either 0 or 1 to every $\hat{p}_{Cu}$. To prove this, consider an attack in which there is a merging cell v with a $\hat{p}_{Cv}$ value other than 0 or 1. If none of its predecessor cells C has a positive $\hat{p}_{Cw}$ value for some other merging cell w, then the assignment for v can clearly be changed to 0 and 1 values without changing the total travel time. Next, suppose that one (or more) of the predecessor cells C of the merging cell has a positive $\hat{p}_{Cw}$ value for some other merging cell w. Then, the total travel time maximizing assignment is clearly one which assigns $\hat{p}_{Cv} = 1$ to a predecessor cell

C for which $\sum_{u \in C} \hat{p}_{Cu}$ is maximal, since this "wastes" the most "merging capacity." Thus, for the remainder of the proof, it suffices to consider only attacks where every $\hat{p}_{Cu}$ value is either 0 or 1.

Now, consider an optimal attack $\mathcal{A}$ against the transportation network, and let $\mathcal{C}^*$ be the set of diverging cells C for which there exists a merging cell u such that $\hat{p}_{Cu} = 1$. Clearly, $\mathcal{C}^*$ forms a set cover of U since for every element u, there is a subset $C \in \mathcal{C}^*$ such that $u \in C$ (i.e., C is connected to u). From our initial supposition, it follows readily that the cardinality of set $\mathcal{C}^*$ must be at least $k+1$. However, this also implies that the total travel time after the attack is equal to $3(k+1)$: in the second time interval, all $k+1$ vehicles may move forward to the diverging cells in set $\mathcal{C}^*$; in the third time interval, all the vehicles may again move forward to the merging cells (since every cell in $\mathcal{C}$ has at least one "enabled" connection); and all the vehicles may leave the network by the next interval through the sink cell. Since the total travel time after an optimal attack $\mathcal{A}$ is equal to $T^* = 3(k+1)$, the attacker's problem does not have a solution. Therefore, the constructed instance of the Attacker's Decision Problem has a solution if and only if the given instance of the Set Cover Problem has one, which concludes our proof. □

3.2 Heuristic Algorithm

Since the attacker's problem is *NP*-hard, we cannot hope for a polynomial-time algorithm that always finds a worst-case attack (unless $P = NP$). Hence, to provide an alternative to the computationally infeasible exhaustive search, we turn our attention to designing an efficient heuristic algorithm.

The attacker's problem can be viewed as the composition of two problems: finding a subset $\hat{\mathcal{S}}$ of signalized intersections and finding new inflow proportions $\hat{p}_{ki}$ for the cells in $\hat{\mathcal{S}}$. For finding a subset $\hat{\mathcal{S}}$, we propose to use a greedy heuristic, which starts with an empty set and adds new cells to it one-by-one, always picking the one that leads to the greatest increase in travel time. Finding new inflow proportions $\hat{p}_{ki}$ is especially challenging, since the set of possible choices is continuous. However, we observe that in most networks, the worst-case configuration is an "extreme" one, which assigns proportion $\hat{p}_{ki} = 1$ to one predecessor cell k and proportion $\hat{p}_{ji} = 0$ to every other predecessor cell j. [1] Hence, for every new cell i added to the set of attacked intersections, we propose to search over the possible extreme configurations by iterating over the predecessors of cell i. Based on the above propositions, we formulate Algorithm 1.

It is fairly easy to see that the running time of Algorithm 1 is $O\left(B \cdot |\mathcal{S}| \cdot \left(\max_{s \in \mathcal{S}} |\Gamma^{-1}(s)|\right) \cdot \text{computing } T\right)$. Since $B \leq |\mathcal{S}|$ and we can compute $T(\mathcal{A})$ for any attack $\mathcal{A}$ using a linear program, it follows readily that the running time of the algorithm is upper bounded by a polynomial function of the input size (i.e., size of the transportation network and B).

[1] In fact, this property holds for *every* network that we have encountered so far. Proving analytically that this property holds for all networks is left for future work.

Algorithm 1 Polynomial-Time Heuristic Algorithm for Finding an Attack

```
A ← (∅, ∅)
for b = 1, ..., B do
    for s ∈ S do
        for k ∈ Γ^{-1}(s) do
            A' ← A ∪ ({s}, {p̂_ks = 1, ∀j ≠ k : p̂_js = 0})
            if T(A') ≥ T(A*) then
                A* ← A'
            end if
        end for
    end for
    A ← A*
end for
Output A
```

4. NUMERICAL RESULTS

In this section, we present numerical results on the heuristic algorithm proposed in the previous section. We compare the heuristic algorithm to an exhaustive-search algorithm, which always finds the worst-case attack (i.e., optimal from the attacker's perspective), but has exponential running time. We study two performance metrics: the travel times resulting from attacks found by the algorithms and the running times of the algorithms.

4.1 Setup

In order for the comparison to be reliable, we have to evaluate the algorithms on a large number of transportation networks. To obtain these networks, we use the Grid model with Random Edges (GRE) to generate random network topologies [15], which closely resemble real-world transportation networks. For a detailed description of this model, we refer the reader to [15, 14].

We set both the width and height of the generated grids to be 4, and let the bottom-left corner be a source and the upper-right corner be a sink. For the parameters controlling the randomness of the generation, we use the values from [15], which were derived from measurements on actual road networks from Europe and the USA. We let the inflow at the source cell be $d^0 = 8$, $d^1 = 12$, $d^2 = 8$, and $d^t = 0$ for $t \geq 3$. For every other cell i, we let the parameters be $Q_i = 6$, $\delta_i = 1.0$, and $N_i = 10$. Finally, we let every merging cell be a signalized intersection, and set the inflow proportions to be uniform over the

predecessors of each intersection.

Due to the randomness of the generation, some of the generated networks pose trivial problems for the attacker, since they allow the sink to be simply cut from the source using the attacker's budget. To make our comparison fair (and pessimistic), we discard these instances, and only use the non-trivial ones. This leaves us with 264 and 122 networks mimicking road networks from the USA and Europe, respectively.

Finally, note that the attacker's action space is continuous since an inflow proportion $\hat{p}_{ki}$ can take any real value from $[0, 1]$. Consequently, to perform an exhaustive search, we must quantize the attacker' action space. For the numerical results, we restricted the proportions to values from $(0, 1/3, 2/3, 1)$ since more fine-grained quantizations did not lead to higher travel times.

4.2 Travel Times

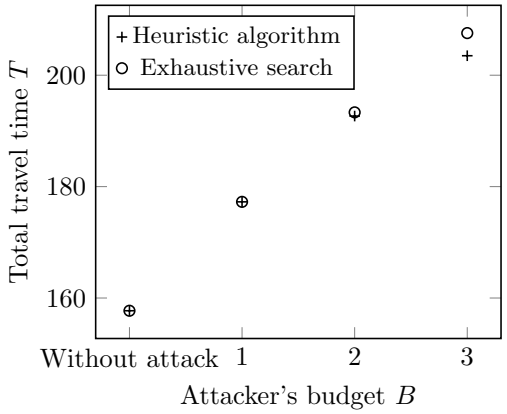

Figure 2: Travel times resulting from attacks found by the heuristic algorithm and by exhaustive search, for randomly generated networks mimicking road networks of the USA.

Figures 2 and 3 show travel times resulting from attacks found by the heuristic algorithm and by exhaustive search, as well as travel times without an attack. Note that the plotted values are averages taken over large numbers of random networks, which were generated using parameters mimicking road networks of the USA for Figure 2 and road networks of Europe for Figure 3. The figures show that the heuristic algorithm performs very well, as the average difference to the exhaustive search remains below 3.4% in all cases.

4.3 Running Times

Figures 4 and 5 show the running times of the heuristic algorithm and the exhaustive search. Again, note that the plotted values are averages taken over large numbers of random networks. As expected, the figures show that the running time of exhaustive search

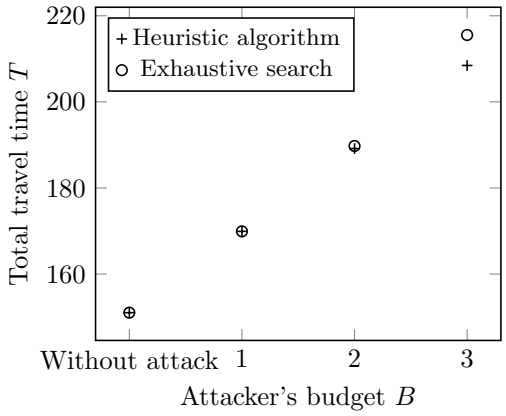

Figure 3: Travel times resulting from attacks found by the heuristic algorithm and by exhaustive search, for randomly generated networks mimicking road networks of Europe.

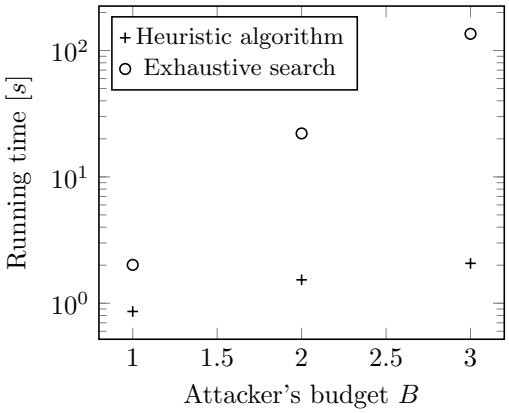

Figure 4: Running times of the heuristic algorithm and the exhaustive search, for randomly generated networks mimicking road networks of the USA.

grows exponentially, and it is multiple orders of magnitude higher than that of the heuristic algorithm even for $B = 3$. Higher values of B are not plotted, as the prohibitively high running time of the exhaustive algorithm prevented us from evaluating the algorithms on a sufficiently large number of networks.

5. SIMULATION RESULTS

So far, we have studied the vulnerability of traffic networks using Daganzo's cell-transmission model, which can be viewed primarily as a macro model. Now, we take a micro-modeling approach, and study the vulnerability of a real-world road network using simulations. The network topology and traffic data used in these experiments is available at `http://aronlaszka.com/data/`

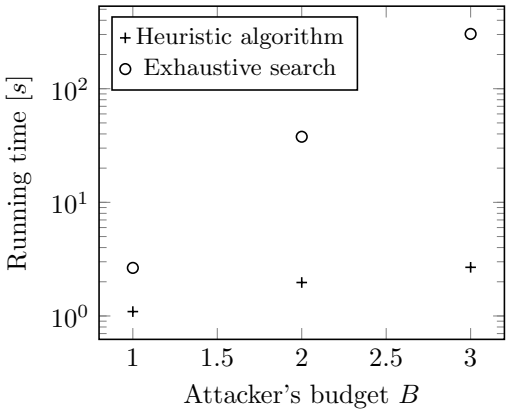

Figure 5: Running times of the heuristic algorithm and the exhaustive search, for randomly generated networks mimicking road networks of Europe.

`laszka2016vulnerability.zip`.

5.1 Setup

Figure 6: Topology of the real-world transportation network used in the simulations. Possible targets for an attack are marked by red disks.

To perform the simulations, we employ SUMO (Simulation of Urban MObility) [2], a well-known and widely-used micro simulator [12, 3]. We retrieved a map of the road network around Vanderbilt University campus from OpenStreetMap [3] (see Figure 6). We selected five major intersections around the campus as possible targets $\mathcal{S}$ for an attack (marked by red disks on Figure 6). The default configurations for these traffic signals were selected to minimize total travel time without considering an attack.

For the supply of vehicles passing through the road network, we generated four traffic scenarios:

- morning: primarily moving from outside of the area to internal destinations (i.e., morning commute), with some traffic between internal points;
- midday: primarily moving between internal points;
- afternoon: primarily moving from internal points to outside of the area (i.e., afternoon commute), with some traffic between internal points;
- nighttime: mostly random traffic traffic.

Finally, we measured the average travel time over all the vehicles instead of their total travel time in this experiment. Since the number of vehicles can differ greatly between traffic scenarios, this facilitates the comparison of the scenarios.

5.2 Varying Budget

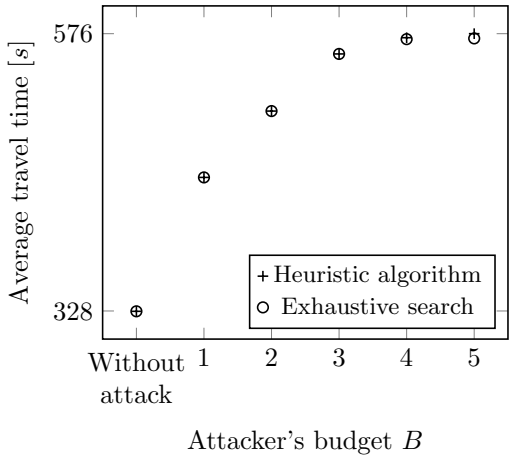

Figure 7: Travel times resulting from attacks found by the heuristic algorithm and by exhaustive search for the road network around Vanderbilt University in the afternoon scenario.

Figure 7 shows the travel times resulting from attacks found by the heuristic algorithm and by exhaustive search, as well as the travel time without any attacks. In this experiment, we used the afternoon scenario. Again, the heuristic algorithm performs exceptionally well, the difference being less than 0.8% to the exhaustive search in terms of the resulting travel time. Due to space limitations, we do not plot the running times of the algorithms for this experiment. The running time of the whole experiment was 8 hours, with the same quantization for the exhaustive search as in the previous section.

5.3 Varying Traffic Scenarios

Finally, Figure 8 shows the travel times with heuristic attack and without attack for various scenarios. In this experiment, we fixed the attacker's budget to $B = 3$. The figure shows that the vulnerability of the trans-

[2] `http://sumo.dlr.de/wiki/Main_Page`
[3] `https://www.openstreetmap.org/`

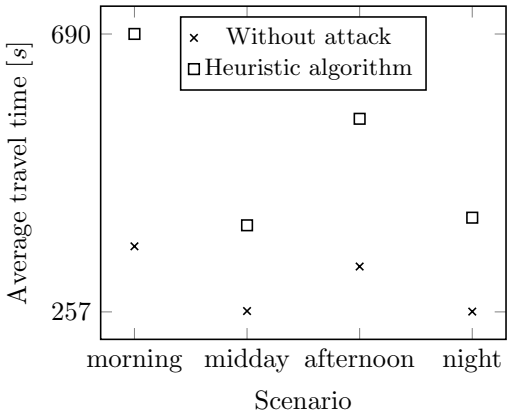

Figure 8: Travel times with heuristic attack and without attack for various traffic scenarios on the road network around Vanderbilt University.

portation network varies between 51% (midday scenario) and 92% (morning scenario).

6. RELATED WORK

In this section, we give a brief overview of the related work on the vulnerability of transportation networks. Due to space limitations, we omit reviewing other, less related areas, such as the vast literature on traffic modeling and assignment [6, 13] and vulnerability analyses from the complex-networks community [1].

A number of research efforts have studied the vulnerability of transportation networks to natural disasters and attacks. However, to the best of our knowledge, our paper is the first one to consider traffic-signal tampering attacks against general transportation networks.

Reilly et al. consider the vulnerability of freeway control systems to attacks on the sensing and control infrastructure [16]. They present an in-depth analysis on the takeover of a series of onramp-metering traffic lights using a methodology based on finite-horizon optimal control techniques and multi-objective optimization.

Sullivan et al. study short-term disruptive events, such as partial flooding, and propose an approach that employs various link-based capacity-disruption values [18]. The proposed approach can be used to identify and rank the most critical links and to quantify transportation network robustness (i.e., inverse vulnerability).

Scott et al. propose a comprehensive, system-wide approach for identifying critical links and evaluating network performance [17]. Using three hypothetical networks, the authors demonstrate that their approach yields different highway planning solutions than traditional approaches, which rely on volume/capacity ratios to identify congested or critical links.

Bell introduces a two-player non-cooperative game between a network user, who seeks to minimize expected travel cost, and an adversary, who chooses link performance scenarios to maximize the travel cost [4, 5]. The Nash equilibrium of this game can be used to measure network performance when users are pessimistic and, hence, may be used for cautious network design.

Jenelius proposes a methodology for vulnerability analysis of road networks and considers the impact of road-link closures [10]. The author considers different aspects of vulnerability, and explores the dichotomy between system-wide efficiency and user equity.

Jenelius and Mattson introduce an approach for systematically analyzing the robustness of road networks to disruptions affecting extended areas, such as floods and heavy snowfall [11]. Their methodology is based on covering the area of interest with grids of uniformly shaped and sized cells, where each cell represents the extent of an event. The authors apply their approach to the Swedish road network, and find that the impact of area-covering disruptions are largely determined by the internal, outbound, and inbound travel demands of the affected area itself.

Alpcan and Buchegger investigate the resilience aspects of vehicular networks using a game-theoretic model, in which defensive measures are optimized with respect to threats posed by intentional attacks [2]. The game is formulated in an abstract manner, based on centrality values computed by mapping the centrality values of the car communication network onto the road topology. The authors consider multiple formulations based on varying assumptions on the players' information, and evaluate their models using numerical examples.

7. CONCLUSION & FUTURE WORK

We introduced an approach for evaluating transportation-network vulnerability, provided computational-complexity results and an efficient heuristic algorithm, and evaluated our approach on both randomly-generated and real-world networks. The primary application of our approach is assessing the vulnerability of a given transportation network and traffic-signal configuration, which is a key step in designing resilient networks and signal configurations. Furthermore, our approach also identifies critical signals, which have the highest impact on congestion. Identifying these critical signals enables the optimal planning and deployment of defensive countermeasures and resources.

Our paper constitutes the necessary first step towards more resilient transportation networks. In future work, we will extend our results in multiple directions. Firstly, we will study how to configure traffic signals in a resilient way, so that even if some of the signals are compromised and tampered with, the default configuration of the uncompromised signals ensures relatively conges-

tion-free traffic flow. We will propose efficient algorithms for finding a resilient configuration, and demonstrate that resilience can be achieved without substantially increasing travel time in the no-attack case. Secondly, we will analyze what makes a traffic signal an attractive target by studying the characteristics of critical signals. We will consider basic graph-theoretic metrics (e.g., node degree), characteristics of the traffic flowing through the intersection, and centrality metrics (e.g., betweenness centrality).

Acknowledgements

This work was supported in part by FORCES (Foundations Of Resilient CybEr-Physical Systems), which receives support from the National Science Foundation (NSF award numbers CNS-1238959, CNS-1238962, CNS-1239054, CNS-1239166), by the Air Force Research Laboratory under Award FA8750-14-2-0180, and by Sandia National Laboratories.

8. REFERENCES

[1] R. Albert, H. Jeong, and A.-L. Barabási. Error and attack tolerance of complex networks. *Nature*, 406(6794):378–382, 2000.

[2] T. Alpcan and S. Buchegger. Security games for vehicular networks. *IEEE Transactions on Mobile Computing*, 10(2):280–290, 2011.

[3] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz. SUMO – Simulation of Urban MObility: An overview. In *Proceedings of the 3rd International Conference on Advances in System Simulation (SIMUL)*, pages 63–68, 2011.

[4] M. G. Bell. A game theory approach to measuring the performance reliability of transport networks. *Transportation Research Part B: Methodological*, 34(6):533–545, 2000.

[5] M. G. Bell, U. Kanturska, J.-D. Schmöcker, and A. Fonzone. Attacker–defender models and road network vulnerability. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 366(1872):1893–1906, 2008.

[6] N. Bellomo and C. Dogbe. On the modeling of traffic and crowds: A survey of models, speculations, and perspectives. *SIAM Review*, 53(3):409–463, 2011.

[7] C. F. Daganzo. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B: Methodological*, 28(4):269–287, 1994.

[8] C. F. Daganzo. The cell transmission model, part II: Network traffic. *Transportation Research Part B: Methodological*, 29(2):79–93, 1995.

[9] B. Ghena, W. Beyer, A. Hillaker, J. Pevarnek, and J. A. Halderman. Green lights forever: Analyzing the security of traffic infrastructure. In *Proceedings of the 8th USENIX Workshop on Offensive Technologies (WOOT)*, pages 1–10, 2014.

[10] E. Jenelius. *Large-scale road network vulnerability analysis*. PhD thesis, KTH, 2010.

[11] E. Jenelius and L.-G. Mattsson. Road network vulnerability analysis of area-covering disruptions: A grid-based approach with case study. *Transportation research part A: policy and practice*, 46(5):746–760, 2012.

[12] D. Krajzewicz, G. Hertkorn, C. Rössel, and P. Wagner. SUMO (Simulation of Urban MObility): An open-source traffic simulation. In *Proceedings of the 4th Middle East Symposium on Simulation and Modelling (MESM)*, pages 183–187, 2002.

[13] P. Patriksson. *The traffic assignment problem: Models and methods*. Topics in Transportation Series. VSP Publishers, 1994.

[14] W. Peng, G. Dong, K. Yang, and J. Su. A random road network model and its effects on topological characteristics of mobile delay-tolerant networks. *IEEE Transactions on Mobile Computing*, 13(12):2706–2718, 2014.

[15] W. Peng, G. Dong, K. Yang, J. Su, and J. Wu. A random road network model for mobility modeling in mobile delay-tolerant networks. In *Proceedings of the 8th International Conference on Mobile Ad-hoc and Sensor Networks (MSN)*, pages 140–146. IEEE, 2012.

[16] J. Reilly, S. Martin, M. Payer, and M. Payer. On cybersecurity of freeway control systems: Analysis of coordinated ramp metering attacks. *Transportation Research, Part B*, 2014.

[17] D. M. Scott, D. C. Novak, L. Aultman-Hall, and F. Guo. Network robustness index: a new method for identifying critical links and evaluating the performance of transportation networks. *Journal of Transport Geography*, 14(3):215–227, 2006.

[18] J. Sullivan, D. Novak, L. Aultman-Hall, and D. M. Scott. Identifying critical road segments and measuring system-wide robustness in transportation networks with isolating links: A link-based capacity-reduction approach. *Transportation Research Part A: Policy and Practice*, 44(5):323–336, 2010.

[19] A. K. Ziliaskopoulos. A linear programming model for the single destination system optimum dynamic traffic assignment problem. *Transportation science*, 34(1):37–49, 2000.