Vulnerability of Fixed-Time Control of Signalized Intersections to Cyber-Tampering

Amin Ghafouri, Waseem Abbas, Yevgeniy Vorobeychik, and Xenofon Koutsoukos
Institute for Software Integrated Systems, Vanderbilt University, USA

Abstract— **Recent experimental studies have shown that traffic management systems are vulnerable to cyber-attacks on sensor data. This paper studies the vulnerability of fixed-time control of signalized intersections when sensors measuring traffic flow information are compromised and perturbed by an adversary. The problems are formulated by considering three malicious objectives: 1) *worst-case network accumulation*, which aims to destabilize the overall network as much as possible; 2) *worst-case lane accumulation*, which aims to cause worst-case accumulation on some target lanes; and 3) *risk-averse target accumulation*, which aims to reach a target accumulation by making the minimum perturbation to sensor data. The problems are solved using bilevel programming optimization methods. Finally, a case study of a real network is used to illustrate the results.**

I. INTRODUCTION

Recent experimental studies claim that about 200,000 vulnerable traffic control sensors are installed in important cities around the world such as New York, San Francisco, London, and Melbourne [3]. This indicates the presence of cyber-threats to traffic management systems, since such systems directly use the data measured by the vulnerable sensors. In order to diminish these threats and design resilient systems, the vulnerability of traffic control systems to cyber-tampering of these sensors must be analyzed as an initial step.

In the traffic management literature, queueing networks are often used to model the movement of traffic [2], [12]. For the traffic control purposes, various signal control policies are defined based on the queue length information such as *max-pressure* [14], [16], which is a feedback control policy, and *fixed-time control* [11], which operates the signal in fixed periodical cycles independent of the traffic state. Although feedback control policies for signalized intersections have advantages in terms of stabilizing the traffic flows, 90 percent of all traffic signals in the US follow fixed-time control policy [8].

Fixed-time control considers deterministic vehicle flows subject to conservation constraints, constraints on saturation flows, and simultaneous turn movements. The formulation of fixed-time control policy leads to a characterization of feasible demands and fixed-time control with minimum cycle length to accommodate the feasible demands [13]. In this direction, Muralidharan et al. showed that under fixed-time control there is a unique periodic trajectory, which is globally asymptotically stable, that is, every trajectory converges to this periodic trajectory [11]. From the periodic trajectory one can easily calculate possible performance measures such as delay, travel time, amount of service time wasted, and progression quality.

Owing to the rising strategic risks of cyber-attacks, exploiting vulnerabilities of transportation systems to cyber-attacks has been an active area of research. For instance, recently Cerrudo has shown that wireless sensors can be spoofed to manipulate traffic light timing [3]. Similarly, in [5], Ghena et al. analyze the security of traffic infrastructure in cooperation with a road agency located in Michigan. The study reports three major weaknesses in the traffic infrastructure: lack of encryption for the network, lack of secure authentication, and vulnerability to known exploits. Furthermore, Laszka et al. have recently proposed an approach for evaluating vulnerabilities of the transportation network by identifying traffic signals with the greatest impact on congestion [9]. They also present that the problem of finding an optimal attack to maximize the congestion is computationally hard, thereby, proposing a polynomial-time heuristic algorithm for computing approximately optimal attacks. Nevertheless, no vulnerability analysis of fixed-time control policy has been done for transportation networks.

In this paper, we study the vulnerability of fixed-time control when a malicious adversary compromises some sensors and perturbs the data corresponding to the traffic flow information. The attacker launches this integrity attack either by directly compromising sensors or by gaining control over the communication network. The tampered data can lead to inefficient scheduling of traffic signals, and in some extreme cases, it can lead to disastrous congestions. In this direction, we formulate three attack problems: 1) *Worst-case network accumulation*, which aims to destabilize the overall network as much as possible; 2) *Worst-case lane accumulation*, which aims to cause worst-case accumulation on some target lanes; and 3) *Risk-averse target accumulation*, which aims to reach a target accumulation by making the minimum perturbation. We formulate these problems as bilevel programs, in which one optimization problem is embedded within the other. Bilevel programs are intrinsically hard to solve, and even the simplest instance, the linear-linear case, is known to be strongly NP-hard [7]. The existing algorithms for solving bilevel programs include branch-and-bound, extreme point, complementary pivot, descent methods, penalty function, and trust-region [4]. We solve the problems using existing implementations of branch-and-bound. Further, we present a case study of vulnerability analysis of a real road network

Emails: amin.ghafouri@vanderbilt.edu, waseem.abbas@vanderbilt.edu, yevgeniy.vorobeychik@vanderbilt.edu, xenofon.koutsoukos@vanderbilt.edu

segment in the city of Nashville.

The remainder of this paper is organized as follows. Section II defines the system model. In Section III, we present the attacker model and formulate the problems. In Section IV, we discuss how the problems can be solved. Sevtion V presents the case study of vulnerability analysis of a real road network. Finally, we conclude the paper in Section VI with a discussion and future work.

II. System Model

A. Network Model

We use the network model presented in [13] with minor modifications in notation. Consider a network of roads modeled as a directed graph with road links being edges $i \in \mathcal{L}_{all}$ and intersections being nodes $n \in \mathcal{N}$. A link can be either an internal link ($i \in \mathcal{L}$) that goes from its start node to its end node, an entry link ($i \in \mathcal{L}_{ent}$) that has no start node, or an exit link $i \in \mathcal{L}_{exit}$ that has no end node.

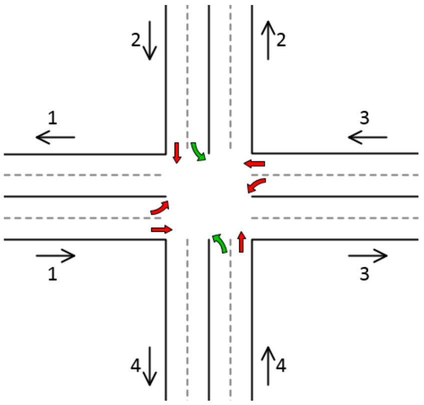
A movement (i, j) describes an intention to travel from a link i to a link j. Let the *flow* corresponding to movement (i, j) be denoted by $f(i, j)$. This means the rate of vehicles intending to leave link i and enter link j per sample period is $f(i, j)$. Flow conservation imposes the following constraint on all $i \in \mathcal{L}$,

$$\sum_{h \in In(i)} f(h, i) = \sum_{j \in Out(i)} f(i, j) \qquad (1)$$

where $In(i)$ and $Out(i)$ are the sets of upstream and downstream links connected to i. This represents the same concept as the formulation presented in [13], with routing proportions being implicit in the formulation of each flow $f(i, j)$.

Intersections are modeled as nodes and traffic signals are placed at every node to limit the set of permitted movements. Defining a *phase* as a pair of links with $j \in Out(i)$ and $i \in \mathcal{L} \cup \mathcal{L}_{ent}$, *saturation flow* of phase (i, j) is denoted by $c(i, j)$. This means that if phase (i, j) is activated, up to $c(i, j)$ vehicles can move from i to j per sample period.

At an intersection n, certain subsets of phases may be simultaneously activated, which is defined as a *stage*. Let $I(n)$ and $O(n)$ denote the set of links entering and leaving intersection n. As shown in Fig. 1, each stage is represented by an intersection control matrix $S_n = \{S_n(i, j), i \in I(n), j \in O(n)\}$ with entries $S_n(i, j) = 1$, if the phase (i, j) is activated, or 0 otherwise. A collection of intersection control matrices S_n, one for each intersection, can be combined into the single network control matrix S, with $S(i, j) = 1$ if for some intersection n, $i \in I(n)$, $j \in O(n)$, and $S_n(i, j) = 1$; otherwise $S(i, j) = 0$. The matrix S can be put in block-diagonal form with the intersection matrices S_n along the diagonal and all other entries zero. The set of all network control matrices S is denoted by $\mathbb{S}$, which is a finite set of 0, 1 matrices.



$$S = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 1 & 0 \\ 3 & 0 & 0 & 0 & 0 \\ 4 & 1 & 0 & 0 & 0 \end{array}$$

Fig. 1: The eight phases of a standard intersection and the control matrix S corresponding to the stage (NW, SE).

TABLE I: Flow data used in the example

From	To	Flow	From	To	Flow
1	6	2	8	13	2
	4	2		11	2
3	14	8	10	7	4
	6	4		13	2
5	2	2	12	9	2
	14	4		7	4
7	4	6	14	11	6
	2	2		9	6

B. Fixed-time Control

Fixed-time control is a collection of network control matrices $S^1, ..., S^k$, and corresponding durations $\lambda_{S^1}, ..., \lambda_{S^k}$, expressed in fractions of a cycle length T [13]. Let L be a fixed lost time per cycle. The minimum cycle length T is defined as $T = \frac{L}{1-(\sum \lambda_S)}\tau$, where τ is the sample rate in seconds. Suppose $F = \{f(i, j)\}$ is a fixed flow matrix. The following linear program (LP) solves the fixed-time control problem

$$\begin{aligned} \min & \sum_{S \in \mathbb{S}} \lambda_S \\ \text{s.t.} & \sum_{S \in \mathbb{S}} \lambda_S c(i, j) S(i, j) \geq f(i, j), \text{all } (i, j) \qquad (2) \\ & \lambda_S \geq 0, \text{ all } \forall S \in \mathbb{S} \end{aligned}$$

Denote by λ^* the minimum value of (2). Flow matrix F is feasible if and only if $\lambda^* < 1$ [13]. The fixed-time LP (2) is easily solvable since it decomposes into small linear programs, one per intersection.

C. Example

Figure 2 presents a network of 2 intersections with 16 phases. Suppose vehicles flow through the network as the flow data shown in Table I. Consider four stages (NS,SN), (WE,EW), (NE,SW), and (WN,ES) for each intersection. More specifically, define the stages $\varphi_1 = \{(3, 14), (7, 4)\}$, $\varphi_2 = \{(1, 6), (5, 2)\}$, $\varphi_3 = \{(3, 6), (7, 2)\}$, and $\varphi_4 = \{(1, 4), (5, 14)\}$ for the first intersection, and $\varphi_5 = \{(14, 11), (10, 7)\}$, $\varphi_6 = \{(12, 9), (8, 13)\}$, $\varphi_7 = \{(14, 9), (10, 13)\}$, and $\varphi_8 = \{(12, 7), (8, 11)\}$ for the second intersection.

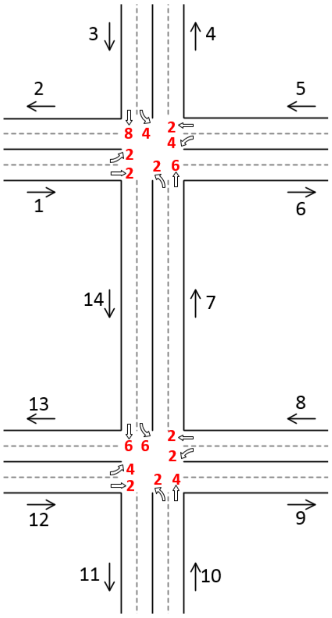


Fig. 2: An example of a network with 2 intersections

TABLE II: Fixed-time durations

Stage	φ_1	φ_2	φ_3	φ_4	φ_5	φ_6	φ_7	φ_8
Duration	.25	.062	.125	.125	.25	.083	.25	.166

Suppose the capacities for all the phases of the first and second intersection are respectively, 32 and 24, and let $\lambda = (\lambda_{\varphi_1}, \lambda_{\varphi_2}, \lambda_{\varphi_3}, \lambda_{\varphi_4}, \lambda_{\varphi_5}, \lambda_{\varphi_6}, \lambda_{\varphi_7}, \lambda_{\varphi_8})$. Solving the LP (2) for each intersection, the fixed-time durations are obtained as shown in Table II. Consequently, for the first and second intersections we obtain $\lambda = \sum_1^4 \lambda_{\varphi_i} = 0.5625$ and $\lambda' = \sum_5^8 \lambda_{\varphi_i} = 0.75$, respectively. Assuming $L = 1$, if the same cycle length is required for the entire network, it is computed as $T = \frac{1}{1-\max(\lambda,\lambda')}\tau = 4\tau$, where τ is the sample rate in seconds.

III. ATTACKER MODEL

In this section, we provide a formulation for attacker models that could result in congestion on road networks implementing fixed-time control policy. We assume that the attacker knows the network model, fixed-time algorithm, implementation, and can thus compute the optimal schedule.

a) Action Space: The attacker compromises some of the sensors measuring flows and perturbs their data. Formally, it selects a subset $\tilde{Q}$ of sensors and perturbs their flow values to $\tilde{F}$. Note that we assume the attacker cannot directly change the schedule. But, this can be done indirectly through perturbing the sensor data. This assumption is realistic since it complies with the real-life cyber-attacks launched in the previous experimental studies [3].

b) Objective: To define the attacker's objective, we first define the notion of a movement being unstable.

Definition 1: Unstable Movement: A movement (i, j) is unstable if its service rate, i.e., $\sum \lambda_S c(i,j) S(i,j))$, is lower than its flow rate $f(i, j)$.

We assume the adversary's objective is to make some movements unstable, which in turn leads to the network becoming unstable. More specifically, we consider the following different strategies for the adversary:

1) *Worst-case network accumulation* which aims to destabilize the overall network as much as possible;
2) *Worst-case lane accumulation* which aims to cause worst-case accumulation on some target lanes;
3) *Risk-averse target accumulation* which aims to reach a target accumulation by making the minimum perturbation.

c) Constraints: We assume the attacker is resource-bounded, which means that there exists a budget B such that the number of compromised sensors $|\tilde{Q}|$ is less than or equal to B, i.e., $|\tilde{Q}| \leq B$. Further, we assume the sensor data and the resulting schedules can only be changed to valid values since otherwise the attack can easily be detected. This means that first, the flow conservation (1) must be satisfied, and second, the schedule obtained using perturbed data must be feasible, i.e., $\lambda^* < 1$. We formulate the attacker problems assuming traffic signals are timed according to the optimal fixed-time schedule.

A. Worst-Case Network Accumulation Attack

The attacker's goal here is to destabilize the network as much as possible and to cause the worst possible traffic congestion. An attack $\mathcal{A}$ has two components of selecting a subset of sensors $\tilde{Q}$ and choosing flow perturbation values $\tilde{F}$. The problem is formally defined below.

Problem 1: Worst-case Network Accumulation Attack: Given a network of signalized intersections and a budget B, find a worst-case attack $\mathcal{A} = (\tilde{Q}, \tilde{F})$ such that it minimizes the service rate of the entire network.

This problem can be formulated as the bilevel program below.

$$\begin{aligned}
\max_{\tilde{Q},\tilde{F}} \quad & \sum_{ij} \max(0, (f_{ij} - \sum_S \tilde{\lambda}_S c_{ij} S_{ij})) \\
\text{s.t.} \quad & \tilde{\lambda}_S \in \text{FT}(\tilde{F}) \\
& \sum \tilde{\lambda}_S < 1 \\
& \sum_h \tilde{f}(h,i) = \sum_j \tilde{f}(i,j) \\
& |\tilde{Q}| \leq B \\
& \tilde{f}(i,j) \geq 0, \ \text{all } (i,j)
\end{aligned} \tag{3}$$

In the formulation above, the term $f_{ij} - \sum_S \tilde{\lambda}_S c_{ij} S_{ij}$, describes the difference between the flow and the service rate. The malicious attacker is only concerned with the positive values for this difference, since negative difference means extra service time, which indeed results in no accumulation. Therefore, the max function is used in the objective function to avoid the negative differences. The first constraint represents the inner-level problem, where FT($\tilde{F}$) corresponds to the fixed-time LP (2) with flow matrix $\tilde{F}$ as its input. The other constraints represent the feasibility of schedule, flow conservation, and attacker's budget respectively.

B. Worst-Case Lane Accumulation Attack

In a targeted attack, the attacker's goal is to maximize the accumulation rate of a particular lane, or similarly to minimize its corresponding service rates, as much as possible.

Problem 2: Worst-Case Lane Accumulation Attack: Given a network of signalized intersections, budget B, and a target lane l^a, find an attack $\mathcal{A} = (\tilde{Q}, \tilde{F})$ that minimizes the service rate of movements corresponding to the lane l^a.

This problem is formulated as

$$\begin{aligned} \min_{\tilde{Q},\tilde{F}} \quad & \sum_j \sum_S \tilde{\lambda}_S c(l^a, j) S(l^a, j) \\ \text{s.t.} \quad & \tilde{\lambda}_S \in \text{FT}(\tilde{F}) \\ & \sum \tilde{\lambda}_S < 1 \\ & \sum_h \tilde{f}(h, i) = \sum_j \tilde{f}(i, j) \\ & |\tilde{Q}| \le B \\ & \tilde{f}(i, j) \ge 0, \text{ all } (i, j) \end{aligned} \tag{4}$$

The objective function is defined as the sum of the service rates of all movements starting from l^a. By minimizing this function, the target lane l^a will have a minimum service time. Note that similar to the previous case, the attacker is restricted by the feasibility of schedule, flow conservation, and budget constraint.

C. Risk-Averse Target Accumulation Attack

A risk-averse attacker has the strategy of reaching a target accumulation rate while minimizing the perturbations. That is, the difference between the perturbed and actual flow values (i.e., $\|\tilde{F} - F\|$) must be minimal.

Problem 3: Risk-Averse Target Accumulation Attack: Given a network of signalized intersections, find an attack $\mathcal{A} = (\tilde{Q}, \tilde{F})$ that leads to an unstable service rate of $\{\alpha_{ij}\}$ for some set of target movements $Q^a = \{(i, j)\}$, via causing a minimum perturbation $\|\tilde{F} - F\|$. This problem is formulated as the optimization problem below.

$$\begin{aligned} \min_{\tilde{Q},\tilde{F}} \quad & \|\tilde{F} - F\|_\infty \\ \text{s.t.} \quad & \tilde{\lambda}_S \in \text{FT}(\tilde{F}) \\ & \sum_S \tilde{\lambda}_S c(i, j) S(i, j) \le \alpha_{ij}, \forall (i, j) \in Q^a \\ & \sum \tilde{\lambda}_S < 1 \\ & \sum_h \tilde{f}(h, i) = \sum_j \tilde{f}(i, j) \\ & |\tilde{Q}| \le B \\ & \tilde{f}(i, j) \ge 0, \text{ all } (i, j) \end{aligned} \tag{5}$$

Note that any other desired norm function can also be used in the objective function.

IV. VULNERABILITY ANALYSIS

In this section, we present solution and evaluation methods followed by an example.

A. Solution

The three problems described above are all strongly NP-hard, but can be solved with the using of integer programming and decomposition algorithms [4] [15]. Although the computational results and finer details of these algorithms have been suppressed here due to space limitations, we discuss some preprocessing steps carried out in order to be able to use known algorithms for solving bilevel programs.

1) Preprocessing: In order to handle the max function in the objective function of the first problem, one can convert the problem to a bilevel mixed-integer quadratic program (BMIQP) as follows. In the objective function, for each term of the form $\max(0, (f_{ij} - \sum_S \tilde{\lambda}_S c_{ij} S_{ij}))$, we introduce an auxiliary binary variable $y_{ij} \in \{0, 1\}$, and add the constraint $f_{ij} - \sum_S \tilde{\lambda}_S c_{ij} S_{ij} \le M y_{ij}$, where M is a sufficiently large constant. Then, we replace the previous objective function with:

$$\begin{aligned} \max_{\tilde{F}} \quad & \sum_{ij} (y_{ij} f_{ij} - y_{ij} \sum_S \tilde{\lambda}_S c_{ij} S_{ij}) \\ \text{s.t.} \quad & f_{ij} - \sum_S \tilde{\lambda}_S c_{ij} S_{ij} \le M y_{ij} \end{aligned} \tag{6}$$

In order to solve the risk-averse attacker problem, we rewrite the objective function $\min \|\tilde{F} - F\|_\infty$ as

$$\begin{aligned} \min \quad & y \\ \text{s.t.} \quad & -y\mathbf{1} \le \tilde{F} - F \le y\mathbf{1} \end{aligned} \tag{7}$$

where $\mathbf{1}$ is a vector of ones.

2) Solver: The problems are solved using methods for solving bilevel mixed integer programs. Existing algorithms in the literature include branch-and-bound, cutting planes, etc. [4]. We use the optimization solver Gurobi to solve the attacker problems [6]. We use the MATLAB toolbox YALMIP to invoke Gurobi's bilevel solver [10]. Also, note that because of the worst-case nature of the first and second problems, the optimal value of corrupted flow has to be as small as possible, and thus, the solver can skip considering different values of $\tilde{F}$ and only try extreme values.

B. Evaluation

1) Metrics: In order to quantify the vulnerability to worst-case network accumulation attack, we define the network vulnerability as follows:

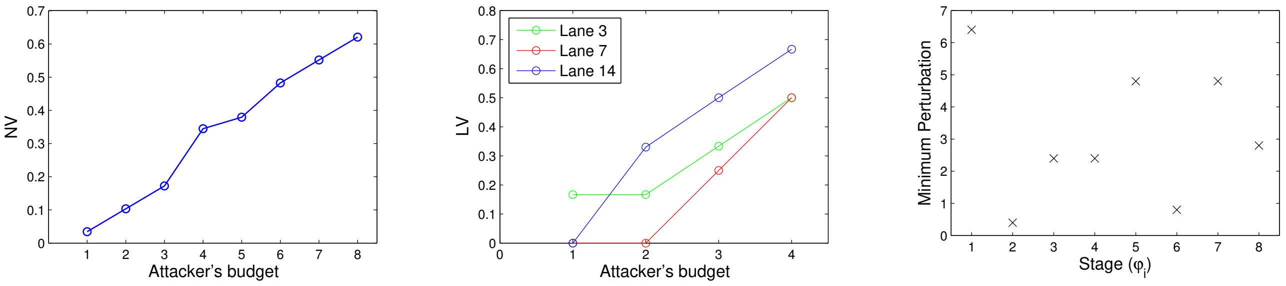
Definition 2: Network Vulnerability: The vulnerability of a network to cyber-tampering is

$$NV = \frac{\text{Accumulation Rate}}{\text{Total Flow}} \tag{8}$$

In the definition above, accumulation rate is the total difference between traffic flow and service rate, i.e., $\sum_{ij} \max(0, (f_{ij} - \sum_S \tilde{\lambda}_S c_{ij} S_{ij}))$, and total flow is the sum of all flow values, i.e., $\sum_{ij} f(i, j)$. The value of network vulnerability represents the relative traffic congestion caused by an attack. We also define lane vulnerability as follows:

Definition 3: Lane Vulnerability: The vulnerability of a lane to cyber-tampering is

$$LV = \frac{\text{Lane Accumulation Rate}}{\text{Lane Total Flow}} \tag{9}$$



(a)

(b)

(c)

Fig. 3: (a) Network vulnerability as a function of attacker's budget in the case of worst case network attack. (b) Lane vulnerability as a function of attacker's budget in the case of worst case lane attacks. (c) Risk-averse target attack. The attacker's goal is to reduce the service time fractions to 0.05 by making the minimum perturbation.

where similarly, lane accumulation is the difference between flow and service rate of the lane, and lane flow is the sum of its corresponding flow values.

2) Critical Sensor: Besides quantifying the vulnerability of network and lanes, we define critical sensors, which have the highest effect on congestion, as follows:

Definition 4: Critical Sensor: A sensor is critical with respect to an attacker's strategy, if it is included in the worst-case attack.

Identifying the critical sensors allows us to locate the most vulnerable elements of a network, which should be strengthened first to increase the network's resilience. For instance, if there is a security budget that permits us to replace only a subset of the sensors with more secure ones, then we should start with replacing the critical sensors.

C. Example

We now study the attacker problems for the network of Fig. 2. We first solve the worst-case accumulation rate problem (3). The results are shown in Fig. 3a as a function of the attacker's budget B. As the budget increases (i.e., the attacker is able to compromise more sensors), the accumulation rate increases as well. Also, the results indicate that by controlling only 4 sensors, the attacker can decrease the total service time by up to 35%.

The worst-case lane accumulation problem is solved similarly. Fig. 3b shows the results for the lanes 3, 7, and 14 as targets according to different budgets. Finally, for the case of risk-averse target accumulation attacks, assume the attacker's objective is to find the minimum perturbation that leads to the target service rate of 0.05, which is indeed unstable for any stage. Fig. 3c shows the minimum perturbation for each stage.

V. CASE STUDY

We analyze the vulnerability of a real road network segment in the city of Nashville, TN. The area spans between 1st Ave, 8th Ave, Demonbreun St, and Charlotte Ave. The network under consideration comprises 15 intersections (12 four-way and 3 three-way), and 104 movements. In order to perform vulnerability analysis, we use real traffic history data provided by Tennessee Department Of Transportation (TDOT) [1]. For lanes with no available data, we estimate their demands using data from their adjacent lanes. Also, since our dataset only provides demands for unidirectional movements, we estimate bidirectional demands considering flow conservation constraints. We assume that fixed-time schedule is computed based on hourly demand, with the total demand being approximately 15000 vehicles per hour.

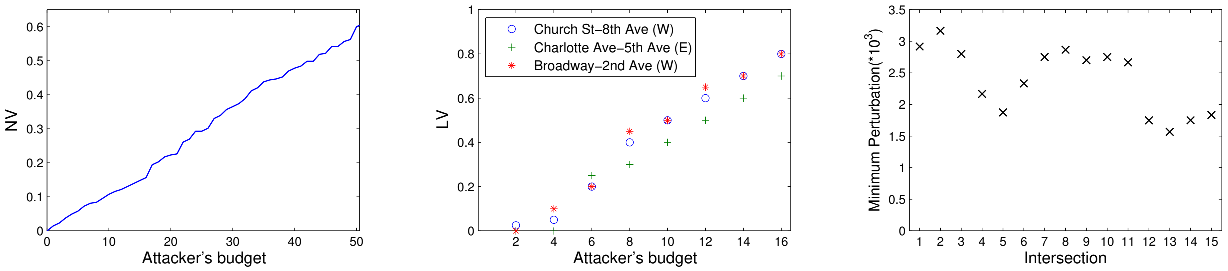
TABLE III: Sensor measurements with the highest frequency of being attacked.

Sensor measurement	Frequency
Charlotte Ave-8th Ave (WE)	98%
Broadway-8th Ave (NW)	97%
Charlotte Ave-8th Ave (SE)	95%
Demonbreun St-8th Ave (NE)	95%
Charlotte Ave-5th Ave (WE)	94%
Charlotte Ave-3rd Ave (NE)	94%
Broadway-8th Ave (WE)	91%
Broadway-5th Ave (WE)	83%

Figure 5a presents the results for the worst-case network accumulation problem. The results indicate that by compromising roughly 21 sensors, which is 20% of the total sensors, the attacker can cause an accumulation of up to 4000 vehicles per hour. Table III shows the sensors that appeared most frequently in the worst-case attacks scenarios.

Next, we solve the worst-case lane attack problem for some target lanes. The results are shown in Fig. 5b for some different budgets. The data shows that on average, it is easier to cause a disastrous congestions on Broadway-2nd Ave than the other two lanes.

Finally, we solve the risk-averse target attack problem. As the target, we assume the attacker has the goal of reducing the service rate of an intersection by at least 50%. The results are shown in Fig. 5c for all 15 intersections. The second and thirteenth intersections (i.e., Charlotte Ave-5th Ave and Demonbreun St-3rd Ave) need the highest and lowest perturbations respectively.



(a) (b) (c)

Fig. 5: (a) Network vulnerability as a function of attacker's budget in the case of worst-case network attack. (b) Lane vulnerability as a function of attacker's budget in the case of worst-case lane attacks. (c) Minimum perturbation needed to reduce the service time of each intersection by at least 50%, in the case of risk-averse target accumulation attack.

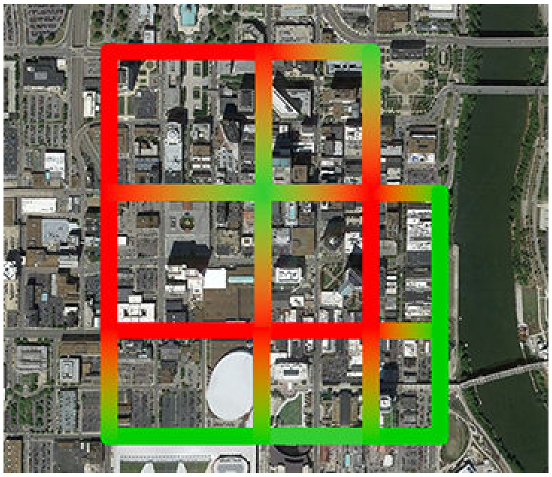


Fig. 4: Traffic heatmap of case study after one hour has passed from a worst-case network accumulation attack. Green represents normal traffic and red represents congested traffic.

VI. CONCLUSIONS

We studied the vulnerability of fixed-time control of signalized intersections when sensors measuring traffic flow information are perturbed by an adversary. As the threat model, we considered an attacker that has the objective of congesting the road network. We formulated three attacker problem and solved them using bilevel programming optimization methods. We found that fixed-time control is vulnerable to cyber-attacks and by compromising only a small number of sensors, an attacker can create severe network congestion. Our approach also identified critical sensors, which have the highest impact on congestion. We illustrated our approach by analyzing the vulnerability of a real road network.

This paper forms the initial step towards more resilient traffic control systems. We aim to extend our results in two directions: first, to design a resilient fixed-time control of signalized intersections so that even if some of the sensors are tampered with, a relatively congestion-free traffic flow is still ensured; and second, to perform the vulnerability analysis of feedback control policies to cyber-tampering.

VII. ACKNOWLEDGMENTS

This work is supported in part by the National Science Foundation (CNS-1238959) and the Air Force Research Laboratory (FA 8750-14-2-0180). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of NSF or AFRL.

REFERENCES

[1] Tennessee Department of Transportation. Tennessee Traffic History, 2014. Available: http://www.tdot.tn.gov/applications/traffichistory. [Accessed: 4/20/2016].

[2] R. E. Allsop. Estimating the traffic capacity of a signalized road junction. *Transportation Research*, 6(3):245–255, 1972.

[3] C. Cerrudo. An emerging us (and world) threat: Cities wide open to cyber attacks. *Securing Smart Cities*, 2015.

[4] B. Colson, P. Marcotte, and G. Savard. An overview of bilevel optimization. *Annals of operations research*, 153(1):235–256, 2007.

[5] B. Ghena, W. Beyer, A. Hillaker, J. Pevarnek, and J. A. Halderman. Green lights forever: analyzing the security of traffic infrastructure. In *8th USENIX Workshop on Offensive Technologies (WOOT 14)*, 2014.

[6] I. Gurobi Optimization. Gurobi optimizer reference manual, 2015.

[7] P. Hansen, B. Jaumard, and G. Savard. New branch-and-bound rules for linear bilevel programming. *SIAM Journal on scientific and Statistical Computing*, 13(5):1194–1217, 1992.

[8] P. Koonce, L. Rodegerdts, K. Lee, S. Quayle, S. Beaird, C. Braud, J. Bonneson, P. Tarnoff, and T. Urbanik. Traffic signal timing manual. Technical report, 2008.

[9] A. Laszka, B. Potteiger, Y. Vorobeychik, S. Amin, and X. Koutsoukos. Vulnerability of transportation networks to traffic-signal tampering.

[10] J. Löfberg. Yalmip: A toolbox for modeling and optimization in matlab. In *Computer Aided Control Systems Design, 2004 IEEE International Symposium on*, pages 284–289. IEEE, 2004.

[11] A. Muralidharan, R. Pedarsani, and P. Varaiya. Analysis of fixed-time control. *Transportation Research Part B: Methodological*, 73:81–90, 2015.

[12] M. Papageorgiou, C. Diakaki, V. Dinopoulou, A. Kotsialos, and Y. Wang. Review of road traffic control strategies. *Proceedings of the IEEE*, 91(12):2043–2067, 2003.

[13] P. Varaiya. Max pressure control of a network of signalized intersections. *Transportation Research Part C: Emerging Technologies*, 36:177–195, 2013.

[14] P. Varaiya. The max-pressure controller for arbitrary networks of signalized intersections. In *Advances in Dynamic Network Modeling in Complex Transportation Systems*, pages 27–66. Springer, 2013.

[15] L. N. Vicente and P. H. Calamai. Bilevel and multilevel programming: A bibliography review. *Journal of Global optimization*, 5(3):291–306, 1994.

[16] T. Wongpiromsarn, T. Uthaicharoenpong, Y. Wang, E. Frazzoli, and D. Wang. Distributed traffic signal control for maximum network throughput. In *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, pages 588–595. IEEE, 2012.