Vulnerability of Transportation Networks to Traffic-Signal Tampering

Aron Laszka\textsuperscript{1}, Bradley Potteiger\textsuperscript{2}, Yevgeniy Vorobeychik\textsuperscript{2}, Saurabh Amin\textsuperscript{3}, Xenofon Koutsoukos\textsuperscript{2}

\textsuperscript{1}University of California, Berkeley
\textsuperscript{2}Vanderbilt University
\textsuperscript{3}Massachusetts Institute of Technology
Evolution of Transportation Networks
Evolution of Transportation Networks

Intelligent Transportation

- reducing wasted time and environmental impact, increasing road safety, etc.
## Evolution of Traffic Control

<table>
<thead>
<tr>
<th></th>
<th>Traditional</th>
<th>Intelligent</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Traffic control devices</strong></td>
<td>standalone hardware</td>
<td>complex networked systems of sensors and controllers</td>
</tr>
<tr>
<td><strong>Traffic signal timing</strong></td>
<td>configured at the time of deployment</td>
<td>adapt to local or global traffic situation</td>
</tr>
<tr>
<td><strong>Traffic flow</strong></td>
<td>varies freely with traffic demand</td>
<td>optimized to minimize, e.g., wasted time and environmental impact</td>
</tr>
<tr>
<td><strong>Vulnerabilities</strong></td>
<td>direct attacks based on physical access</td>
<td>attacks through wireless interfaces or remote attacks over the Internet</td>
</tr>
</tbody>
</table>
Vulnerabilities in Traffic Signals

Case study by University of Michigan [1]

• In cooperation with a road agency located in Michigan, which operates around a hundred traffic signals
• Intersections are part of the same network, but operate individually
• Major weaknesses:
  • wireless communication is **unencrypted**
  • controllers are vulnerable to **known exploits**
  • devices use **default usernames** and **passwords**

Attacks Based on Traffic Signal Tampering

- Due to hardware-based failsafes, these vulnerabilities cannot be used directly to cause traffic accidents.
- However, they may be used to cause disastrous traffic congestions, which can effectively cripple a transportation network.

How vulnerable are transportation networks to such attacks?
Vulnerability Assessment

Transportation network + Model = vulnerability metric

- Attacker Model
- Signalized Intersection Model
- Traffic Model

- critical intersections
1. Traffic Model: Daganzo’s Cell Transmission Model

- Well-known and simple approach for modeling traffic flow
- Discrete: **time** is divided into **intervals**, while **roads** are divided into **cells**

\[
y_{ij} = \min(x_i, Q, \delta(N - x_j))
\]

- Traffic flow is limited by the capacity and the congestion level of the successor cell
2. Signalized Intersection Model

- Intersection:
  cell with multiple predecessors

- Signalized intersection:
  inflow proportions are controlled by the signal schedule

\[ y_{ij} \leq p_{ij} \times \min(Q, \delta(N - x_j)) \]

\[ \sum_i p_{ij} = 1 \]
3. Attacker Model

- **Action space**
  - **budget limit**: attacker can compromise at most $B$ intersections
  - **tampering**: attacker can change the schedule (i.e., inflow proportions $p_{ij}$) of every compromised intersection $j$
  - **failsafes**: the attacker can select only valid schedules (i.e., the inflow proportions must add up to one: $\sum_i p_{ij} = 1$)

- **Goal**
  - **worst-case**: attacker minimizes the network’s utility by maximizing its congestion

- We quantify congestion as the **total travel time** $T$ of the vehicles that enter the transportation network
Vulnerability and Critical Intersections

Vulnerability of a transportation network:

\[ \frac{T(A) - T}{T} \]

- \( T \): total travel time without attack
- \( T(A) \): total travel time resulting from a worst-case attack

Critical intersections:
an intersection is **critical** if it is an element of a worst-case attack
Computational Complexity

**Theorem:** Given a transportation network, an attacker budget $B$, and a threshold travel time $T^*$, determining whether there exists an attack $\mathcal{A}$ satisfying the budget constraint such that $T(\mathcal{A}) > T^*$ is NP-hard.

- We cannot hope to find polynomial-time algorithms for evaluating the vulnerability of a transportation networks against signal-tampering attacks.
Heuristic Algorithm for Finding an Attack

- Combination of two principles:
  - **outer search:** 
    greedy heuristic for selecting the set of intersections to target
  - **inner search:** 
    for each new intersection \( j \), exhaustive search over extreme configurations (i.e., \( p_{ij} = 1 \) for some \( i \))

- Running time: polynomial in the size of the input

---

Algorithm 1 Polynomial-Time Heuristic Algorithm for Finding an Attack

\[
\begin{align*}
\mathcal{A} &\leftarrow (\emptyset, \emptyset) \\
&\text{for } b = 1, \ldots, B \text{ do} \\
&\quad \text{for } s \in S \text{ do} \\
&\quad \quad \text{for } k \in \Gamma^{-1}(s) \text{ do} \\
&\quad \quad \quad \mathcal{A}' \leftarrow \mathcal{A} \cup \{s\}, \{\hat{p}_{ks} = 1, \forall j \neq k : \hat{p}_{js} = 0\} \\
&\quad \quad \text{if } T(\mathcal{A}') \geq T(\mathcal{A}^*) \text{ then} \\
&\quad \quad \quad \mathcal{A}^* \leftarrow \mathcal{A}' \\
&\quad \quad \text{end if} \\
&\quad \text{end for} \\
&\text{end for} \\
&\mathcal{A} \leftarrow \mathcal{A}^* \\
&\text{end for} \\
\text{Output } \mathcal{A}
\end{align*}
\]
Numerical Evaluation

- Random road networks:
  
  * Grid model with Random Edges (GRE) [2]
    
    - grid with **randomly** chosen horizontal/vertical edges **removed** and diagonal edges **added**
    
    - resulting networks are **very similar** to **real-world** road networks with respect to various metrics (e.g., road density, shortest-paths)
  
  - Generated 300 random networks
    
    - resembling either European or US cities
  
  - Performed an **exhaustive search** and the **heuristic algorithm** on each network

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’s action space. For the numerical results, we restricted the proportions to values from $(0, \frac{1}{3}, \frac{2}{3}, 1)$ since more fine-grained quantizations did not lead to higher travel times.

### 4.2 Travel Times

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 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/](http://aronlaszka.com/data/).
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's 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

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 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/.
Micro-Model Based Simulations

How well does the algorithm perform in a micro model?

• SUMO simulator
  (Simulation of Urban MObility)
  • widely-used microscopic simulator
  • traffic demand: placing individual vehicles on the road network and setting their trajectories
  • traffic light schedule: modeled explicitly by SUMO

• Total travel time $T(\mathcal{A})$: total travel time output by SUMO
Example Transportation Network

- Transportation network
  - area around Vanderbilt University campus
  - from OpenStreetMap
- Traffic scenarios
  1. morning commute
  2. midday
  3. afternoon commute
  4. nighttime

(all data available on the first author’s homepage)
Travel Times in the Afternoon Scenario

![Graph showing travel times with and without attack]

The graph illustrates the average travel time in seconds for different attacker's budgets. The x-axis represents the attacker's budget, and the y-axis shows the average travel time. The graph compares the travel times resulting from attacks found by the heuristic algorithm and by exhaustive search, as well as the travel time without any attacks.

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, the heuristic algorithm performs exceptionally well, with a difference 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.

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 transportation network is less than 0.8% difference in every case.
Comparison of Scenarios

vulnerability varies between **51%** (midday scenario) and **92%** (morning scenario)
Ongoing Work: Resilient Traffic Signal Configuration

- **Resilient configuration:**
even if some of the traffic signals are compromised and reconfigured, the default configuration of the remaining signals ensures acceptable traffic flow

- **Tradeoff:**
  
  resilience ↔ efficiency
  
  travel time after attack ↔ travel time without attack

  *Can we increase resilience without a significant sacrifice of efficiency?*
Numerical Example

• Example network:

• Pareto optimal configurations:
Numerical Example

• Example network:

• Pareto optimal configurations:
Numerical Example

- Example network:

- Pareto optimal configurations:

  ![Diagram of a network with targetable intersections](image)

  **15:1 tradeoff**

  ![Graph showing travel time after attack vs. normal travel time](image)
Conclusion & Future Work

• Approach and algorithm for evaluating the vulnerability of transportation networks

• Evaluation based on a large number of random networks and a real-world road network

• Future work: what makes a traffic signal critical?
  • what metrics are related to vulnerability and criticality
    (e.g., characteristics of the traffic flowing through the intersection, graph-theoretic metrics, such as centrality)
Thank you for your attention!

Questions?