Stuck in Traffic (SiT) Attacks

Mina Guirguis

Texas State University

Joint work with George Atia

77th Vehicular Technology Conference
(VTC), Dresden, Germany, June 2013

Traffic

Intelligent Transportation Systems

- V2X communication enable drivers to make better decisions:
  - Avoiding congestion
  - Balancing traffic across multiple routes
  - Cooperating with other drivers
- Already, happening implicitly to a subset of drivers:
  - Smartphone apps
- Vision: more explicit through smart traffic signs and software agents on the vehicles

Challenges

- Reliance on wireless communication
  - Attackers can interfere with/jam the signals preventing communication
- Complexity
  - Harder to understand and debug – not all drivers will follow the signs – suggestive ones!
- Studies consider communication failures as “random” noise
  - Attacks are not “random”, but are well orchestrated

Contributions

- Research questions:
  - Can ITS be exploited by attackers to cause congestion?
  - Can attackers do this in a smart way to avoid detection?

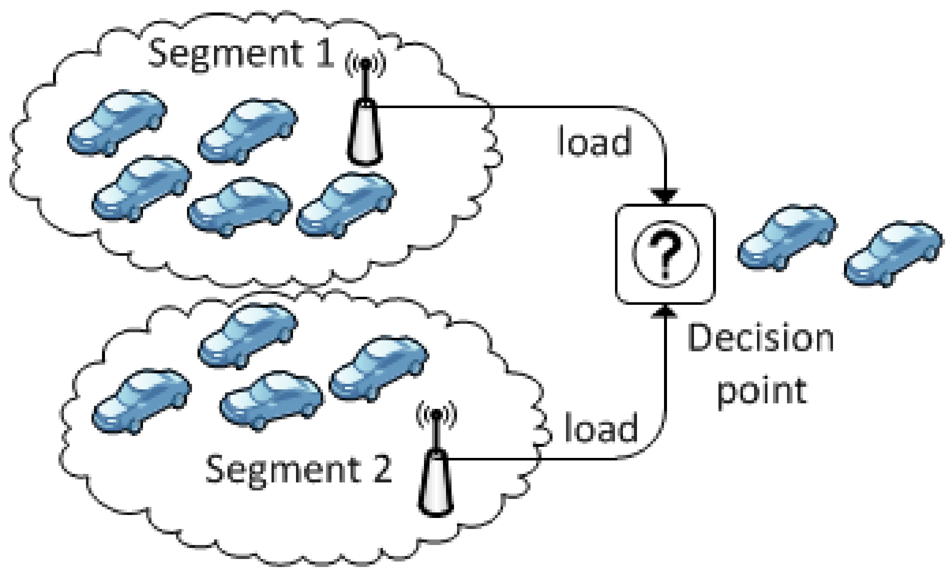
- Contributions:
  - Develop a general framework to identify stealthy attacks – minimize cost and maximize damage
  - Expose SiT attacks that decides *which* signal to interfere with and *when*
  - Attack policies identified outperform other attack policies (e.g., DoS, random and myopic)

Talk outline

- Motivation
- An MDP framework
- Results
- Conclusions

ITS: balancing traffic

- ITS goal: balance traffic across road segments
  - Segment – part of an infrastructure controlled by a Road Side Unit (RSU)
  - Decision Point – a point in which drivers make informed decisions
- How:
  - Vehicles on each segment report to their RSU to get an estimate of the load
  - Decision point influence the choice made by incoming traffic to balance traffic



The model

- Discrete-time system of n segments, indexed by time k
- Number of vehicles on segment *i* at time *k*:

$$q_k(i) \quad = \quad q_{k-1}(i) + \underbrace{\alpha_{k-1}}_{\text{admission ratio}}(i)\underbrace{\lambda_k}_{\text{Arrival rate}} - \underbrace{\beta_k}_{\text{Service rate}}(i).$$

- Traffic optimization function:

$$\alpha_k(i) = f\left(q_{k-1}(1), q_{k-1}(2), \ldots q_{k-1}(n)\right)$$

SiT attacks

- Goal: unbalance traffic causing congestion
- How:
  - Attacker jams some signals from vehicles to the RSU by action u
  - RSUs get incorrect estimates

$$\hat{q}_k(i) = h(q_k(i), u_k)$$

Attack action

  - Decision point does not reflect true conditions

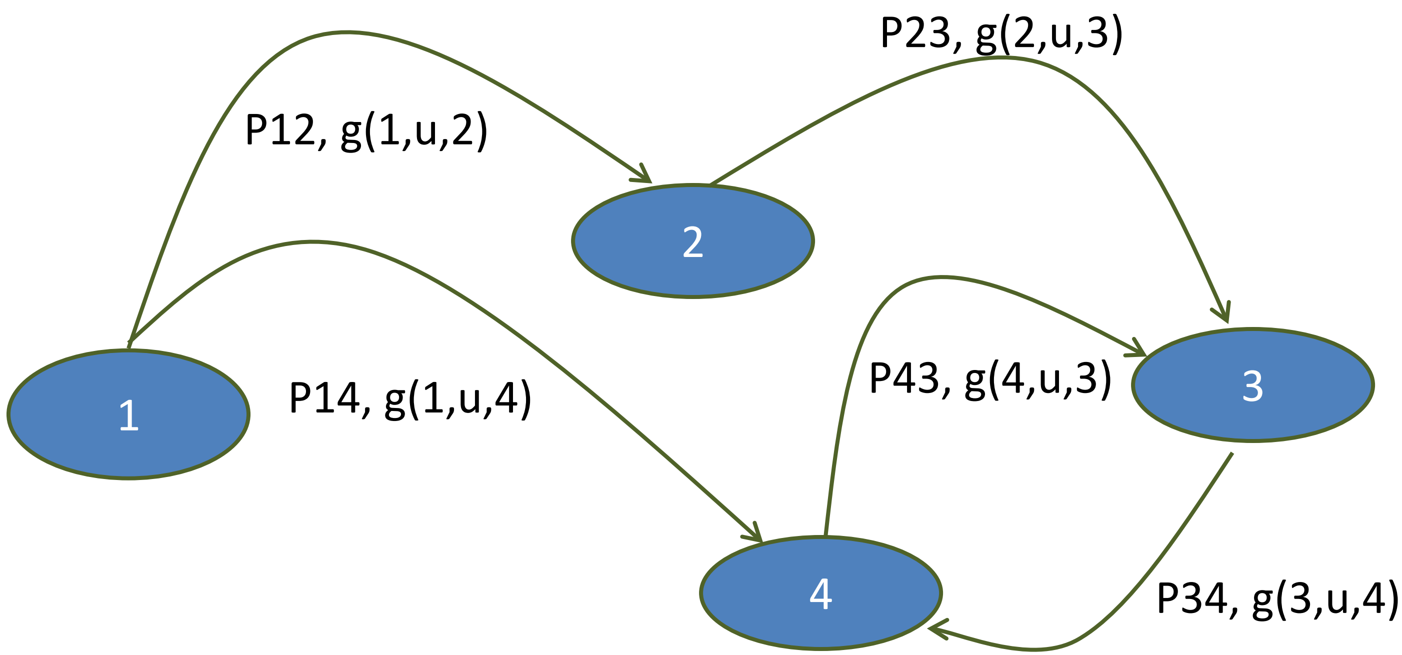
$$\alpha_k(i) = f\left(\hat{q}_{k-1}(1), \hat{q}_{k-1}(2), \ldots \hat{q}_{k-1}(n)\right)$$

  - Incoming vehicles make wrong decisions

Markov Decision Process

- The state at time k:
  - Number of cars on each segment
  - Decision info displayed to drivers
- State transitions
  - Randomness from the arrival probability distribution
  - Attack actions (no attacks, attack 1 segment, attack 2 segments, etc…)
- Rewards
  - Damage: unbalance in traffic
  - Cost: price incurred when a segment is attacked

Illustration



- Pij: probability of transition from state i to state j
- g(i,u,j): reward under action u
- Policy: selecting an action u for every state

Bellman's equation

- To obtain an optimal policy, attacker solves:

$$J^*(i) \quad = \quad \max_{u \in U(i)} \sum_{j=1}^{n} p_{ij(u)(g(i,u,j)+\alpha J*(j))}$$

$$\begin{aligned}
J^*(i) &= optimal\ cost-to-go\ for\ state\ i \\
U(i) &= controls\ available\ from\ state\ i \\
g(i,u,j) &= immediate\ reward\ from\ state\ i\ to\ state\ j\ under\ control\ u \\
\alpha &= discount\ factor
\end{aligned}$$

Bellman's equation

- To obtain an optimal policy, attacker solves:

$$J^*(i) \quad = \quad \max_{u \in U(i)} \sum_{j=1}^{n} p_{ij(u)(g(i,u,j)+\alpha J^*(j))}$$

- Immediate reward reflects tradeoffs between the damage inflicted and the cost of the attack

$$g(i, u, j) \quad = \quad Damage(i, j) - Cost(u)$$

Bellman's equation

- To obtain an optimal policy, attacker solves:

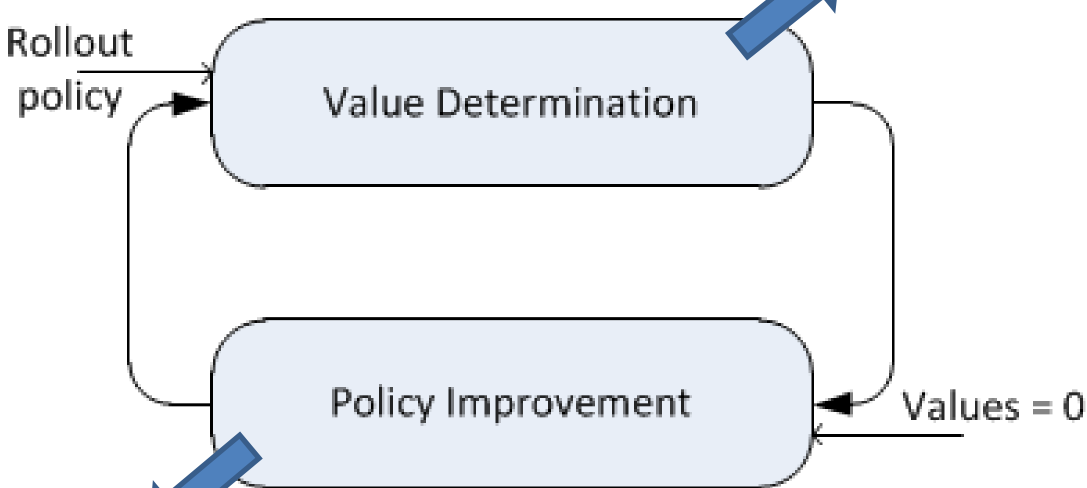
$$J^*(i) \quad = \quad \max_{u \in U(i)} \sum_{j=1}^{n} p_{ij(u)(g(i,u,j)+\alpha J*(j))}$$

- To find the optimal policy
  - Value iteration
  - Policy iteration

Policy iteration

- Exact policy iteration

$$J^{\mu}(i) \quad = \quad \sum_{j=1}^{n} p_{ij}(u)(g(i,u,j) + \alpha J^{\mu}(j))$$



$$\bar{\mu}(i) \quad = \quad \max_{u \in U(i)} \sum_{j=1}^{n} p_{ij}(u)(g(i,u,j) + \alpha J^{\mu}(j))$$

- There is a curse!

The curse of dimensionality!

- Finding optimal policies is difficult!

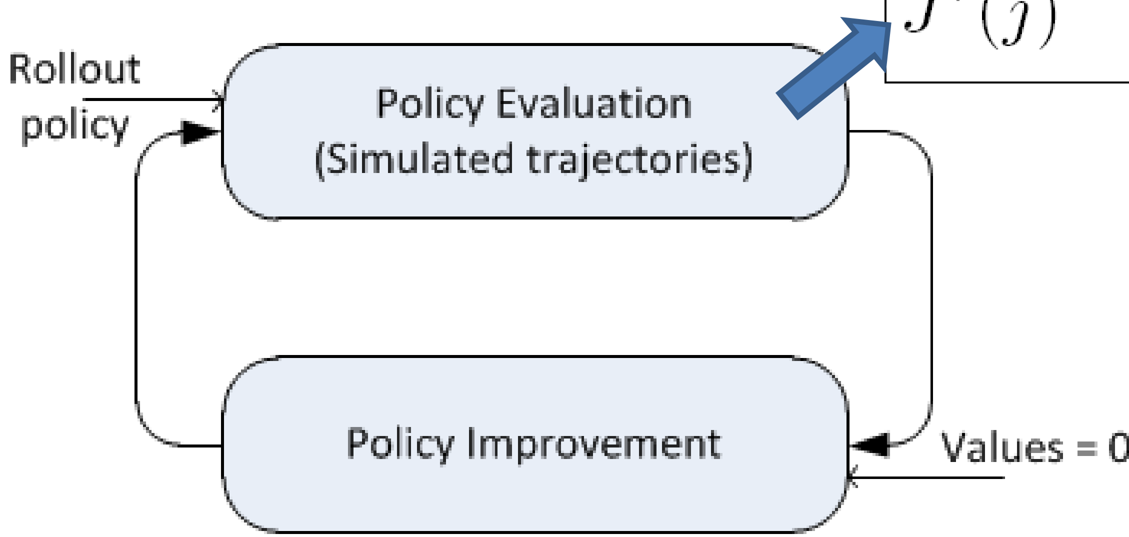
- Sample example
  – A 2 segment setup with 100 vehicles has 100^8 state space
  – Cannot solve a system of linear equations!

- Need to look for approximations!

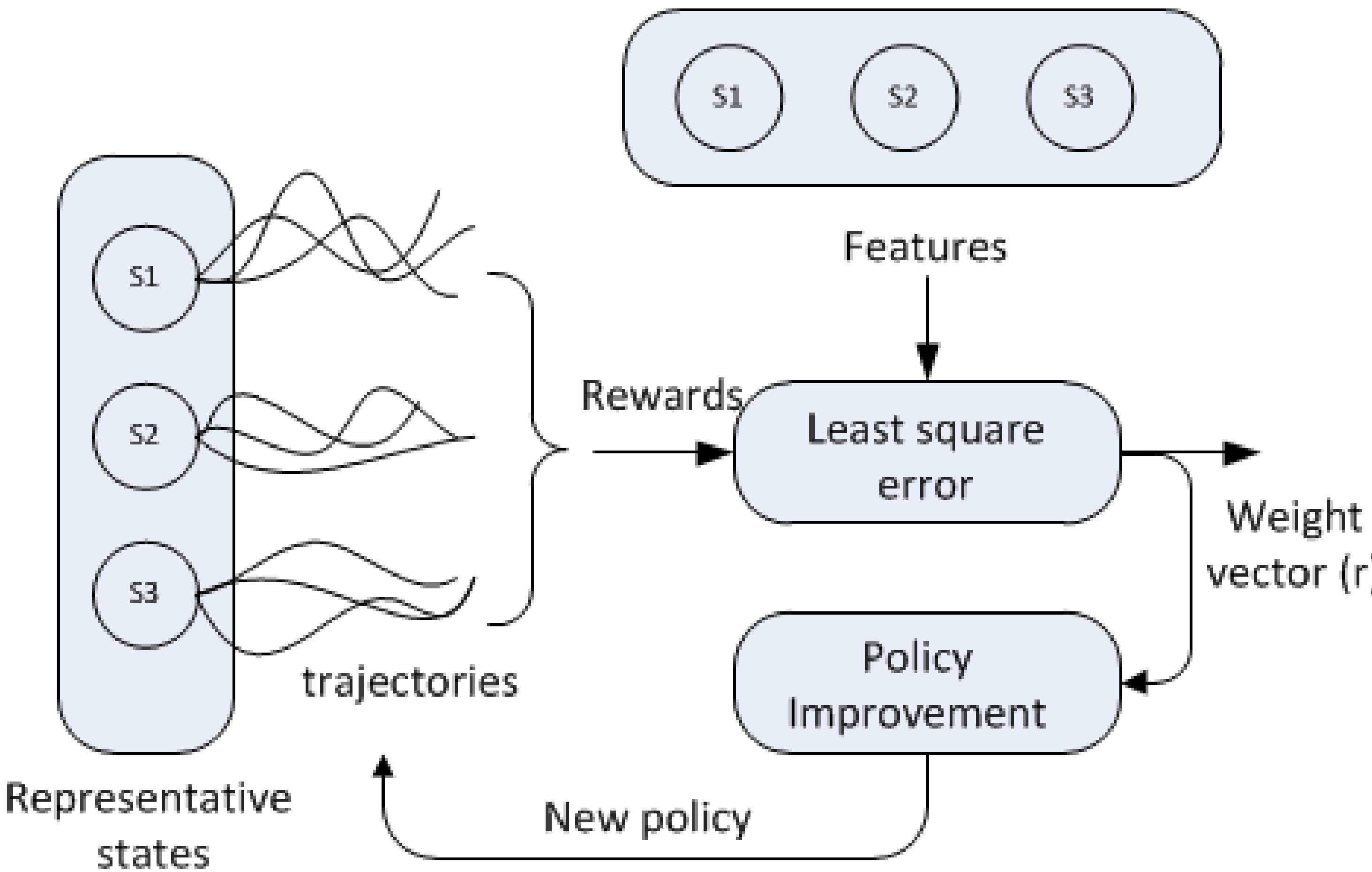
Approximate policy iteration

- Replace the optimal values with a parametric cost-to-go function
  - Obtain the approximate cost-to-go from simulations
  - Characterize every state by s features
  - r is a weight vector, one for each feature

$$J^*(j) \quad \approx \quad \tilde{J}(j,r)$$



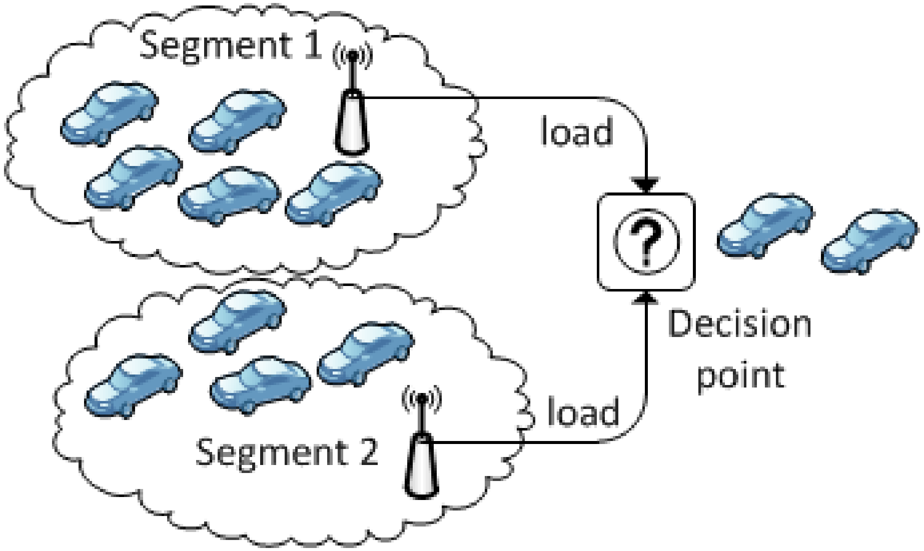
API algorithm



Talk outline

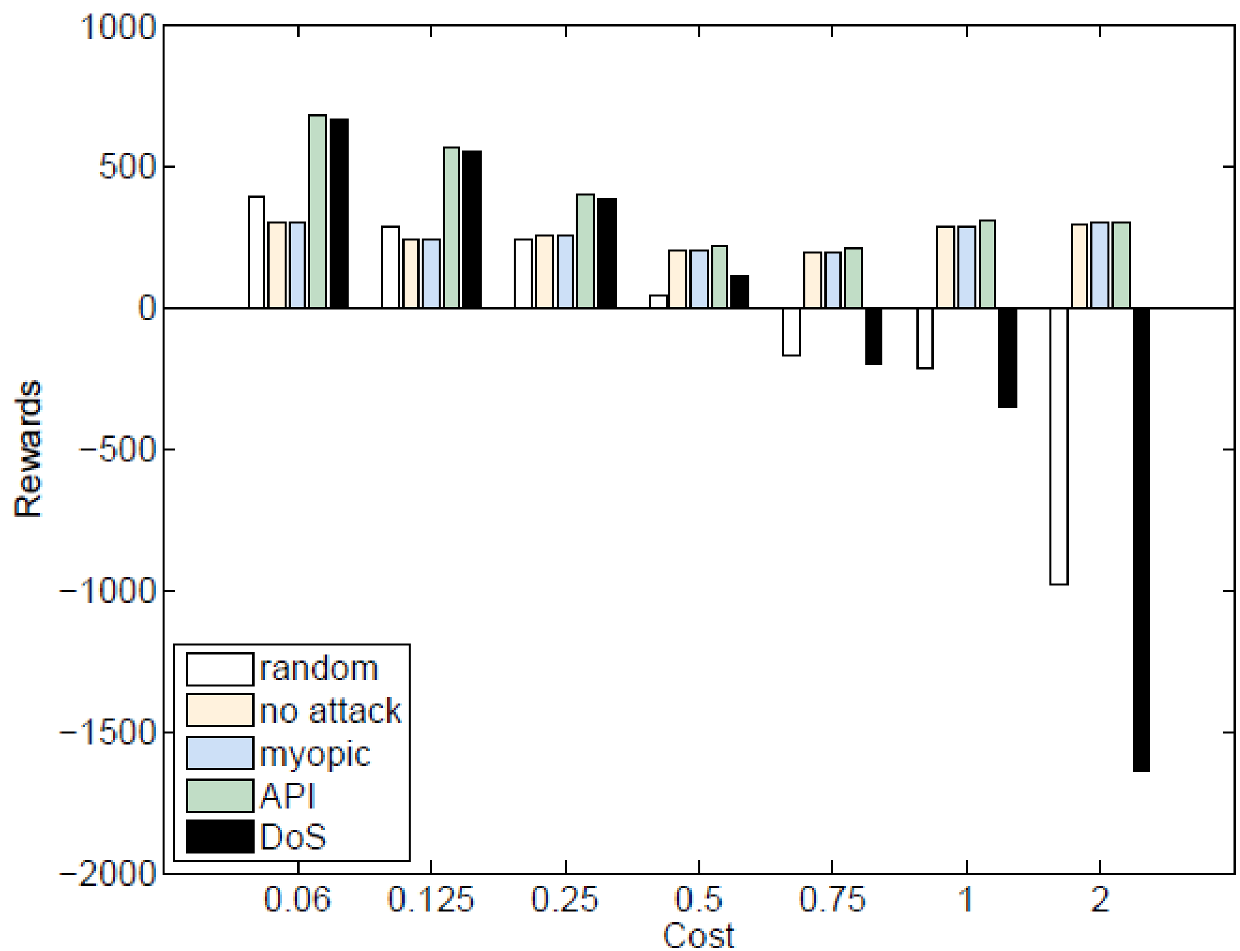
- Motivation
- An MDP framework
- Results
- Conclusions

Experimental setup

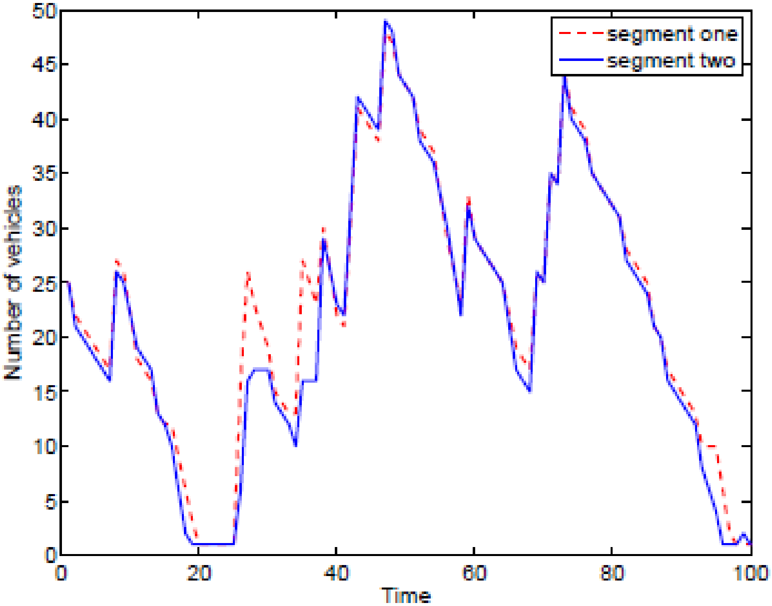


- The setup
  - Two segments
  - Arrival distribution
    - 3 prob. 0.3
    - 8 prob. 0.6
    - 30 prob. 0.1
  - Service rate fixed to 5 v/time
- A SiT attack affects 50% of vehicles
  - Damage: $|q_k(1) - q_k(2)|$
  - Cost: $C_T \times 0.5 \times q_k$

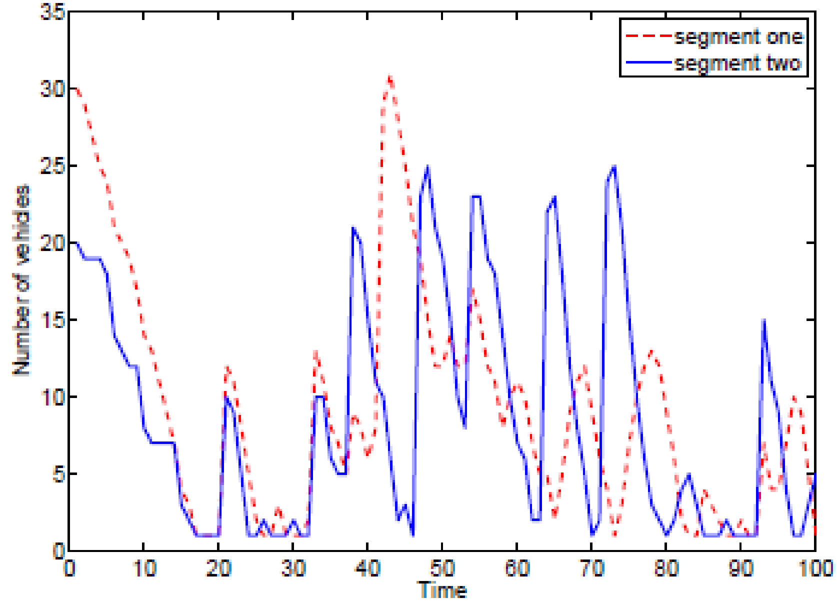
System 1: two identical segments



System 2: different segments



System 1



System 2

	No Attack	Attack Seg1	Attack Seg2
System 1	**35%**	**45%**	**20%**
System 2	**4%**	**20%**	**76%**

Conclusions

- Developed a framework to identify stealthy attacks that cause congestion – SiT attacks
  - Demonstrated their potency in comparison to other attack policies (DoS, random, myopic)
  - Adapt to system parameters while balancing between current and future rewards
- As the degree of uncertainty increases, the policies obtain perform better
- Important to investigate the safety of ITS as they are developed

Stuck in Traffic (SiT) Attacks

Thank you!

This work is supported by NSF