

# Multi-Agent Reinforcement Learning for Assessing False-Data Injection Attacks on Transportation Networks

Taha Eghesad✳, Sirui Li❖, Yevgeniy Vorobeychik †, Aron Laszka✳

✳ Pennsylvania State University

❖ Massachusetts Institute of Technology

† Washington University In St. Louis



**PennState**

23rd International Conference On Autonomous Agents And Multi-Agent Systems (AAMAS)

Auckland, NZ, May 6-10, 2024

# WHY TRANSPORTATION SERVICES ARE IMPORTANT?

- Provide access to:
  - Education
  - Healthcare
  - Emergency Services
- Contribute to:
  - Economic growth
  - Logistic services
  - Delivery of essential goods

- Disruptions can lead to:
  - Financial losses
  - Physical damage
  - Bodily harm



# VULNERABILITY OF TRANSPORTATION NETWORKS



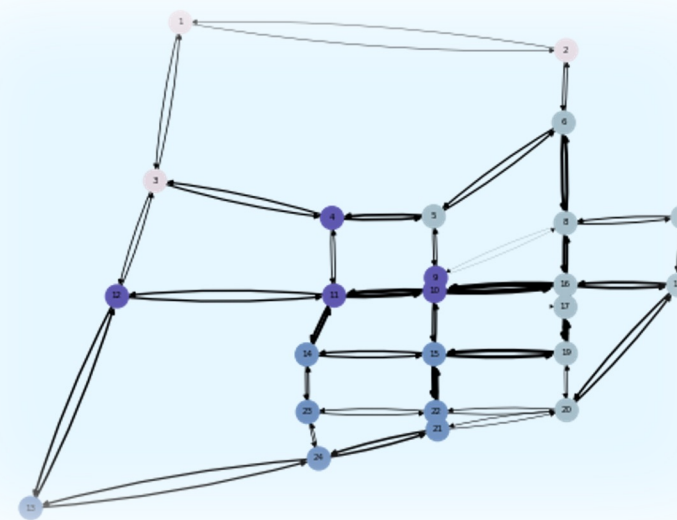
- SMS Disinformation
- Traffic **Sign** Manipulation
- Traffic **Signal** Manipulation
- False Data Injection in Navigation Applications



# TRANSPORTATION NETWORK MODEL



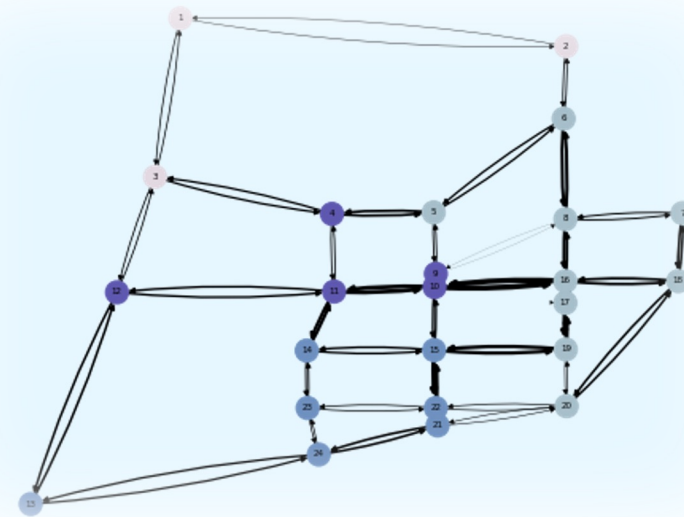
- A **directed graph**  $G = \langle V, E \rangle$  defines the transportation network's roads and intersections
- **Congestion Model**
  - Each road has a given free-flow travel time
  - The more vehicles on a given road, the higher the actual travel time
- **At each intersection**, drivers take the **shortest path to their destination** based on a navigation application



Sioux Falls, SD

# FALSE DATA INJECTION (THREAT) MODEL

- The attacker has a budget to perturb perceived travel times
  - The attacker perturbs perceived travel times at each step
  - The drivers take a longer path due to perceived congestion
- 
- **Strong threat model:**  
The attacker has full observation of the network
    - Vehicle locations
    - Vehicle destinations



Sioux Falls, SD

# PROBLEM FORMULATION

- **Assessing the extent of the damage** is the prerequisite for defense
  - An attack oracle can be used to **generate worst-case** attacks for detection and mitigation schemes
- False data injection attacks may happen **over a time horizon**
- **Uncertainty** of the environment
- The attacker can manipulate observed congestion in a navigation application
  - Restricted to a **fixed budget**
  - Able to manipulate **any road link**
  - Aiming to cause **worst-case impact**
- Find a policy, mapping from **network state** to **perturbations**, that maximize **total travel time**
- Leading to: **Markov Decision Process (MDP)** formulation

$$MDP = \langle S, A, R, T \rangle$$

$S \mapsto$  state space

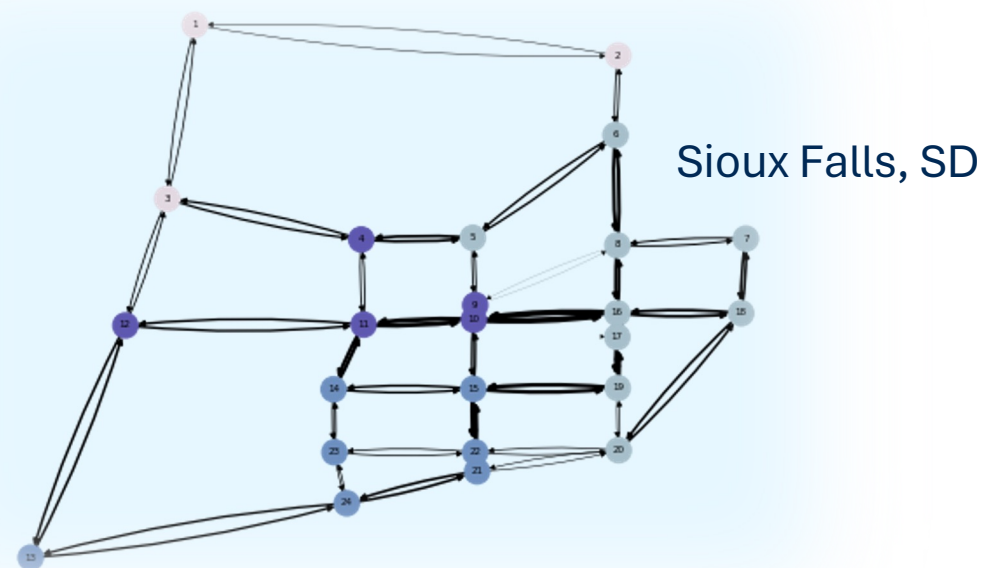
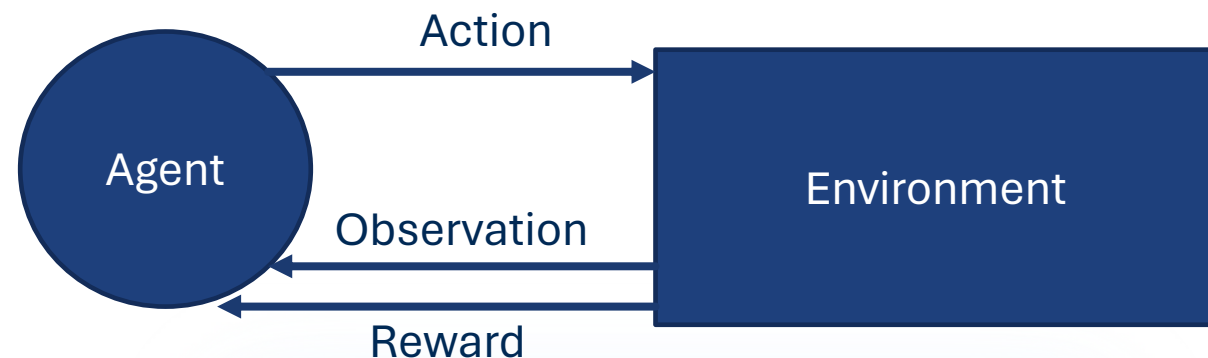
$A \mapsto$  action space

$R(s, a) \mapsto$  rewarding rule

$T(s, a) \mapsto$  transition rule

# REWARD, ACTION, AND STATE SPACE

- Objective
  - Goal: maximize total travel time
  - Reward:  $r^t =$  number of vehicles in traffic
  
- Action Space
  - Perturb observed edge travel times restricted to a budget
  - Action space:  $|a^t|_1 \leq B$  and  $a_e^t \geq 0$
  
- State Space
  - Vehicle locations and destinations



# DEEP REINFORCEMENT LEARNING AS ATTACK ORACLE



- Reinforcement Learning

optimize  $\pi(o^t) \mapsto a^t$

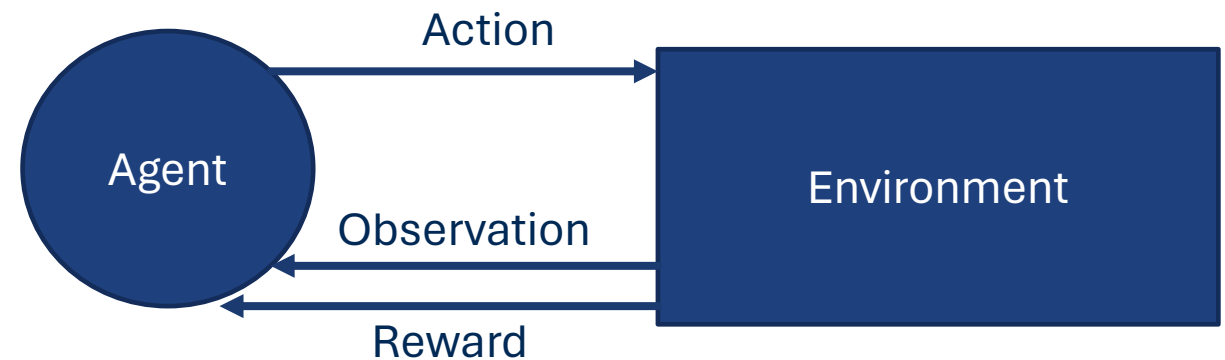
$\max \mathbb{E}[\sum_{\tau=0}^{\infty} \gamma^{\tau} \cdot r^{t+\tau} | \pi]$

- **Critic:**  $Q(o^t, a^t) \leftarrow r^t + \max_{a'} Q(o^{t+1}, a')$

- Updated by gradient descent, reducing Mean Squared Bellman Error

- **Actor:**  $\pi(o^t) \leftarrow \operatorname{argmax}_{a'} Q(o^{t+1}, a')$

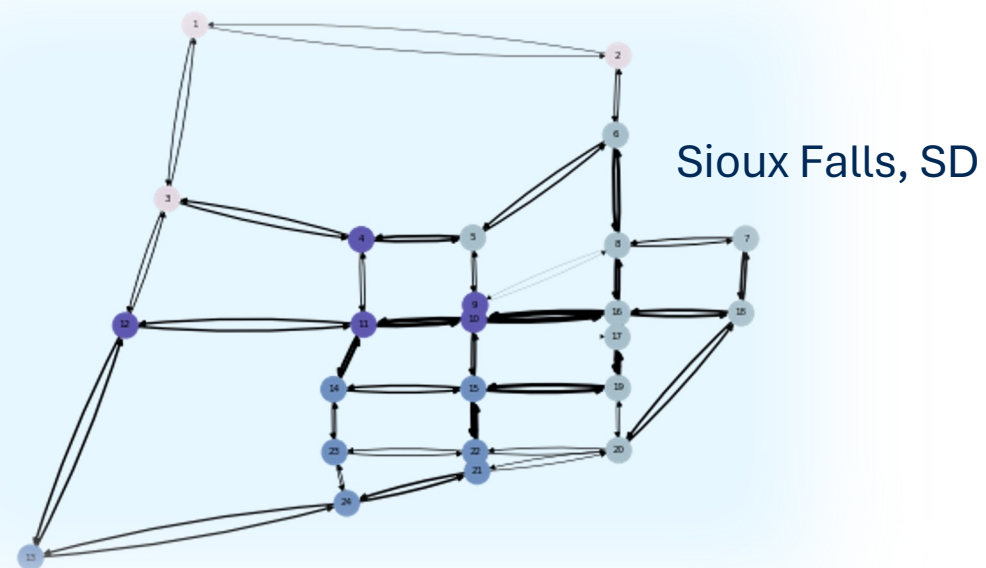
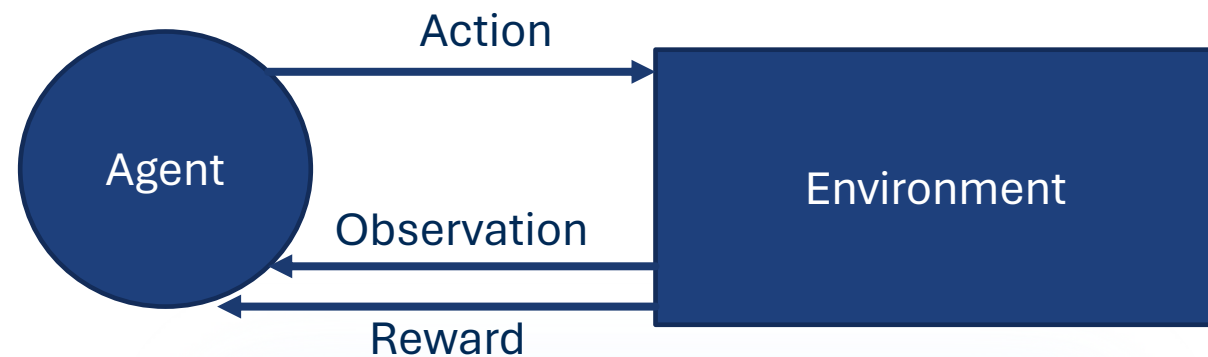
- Updated with gradient ascent, increasing  $Q$





# FEATURE EXTRACTION

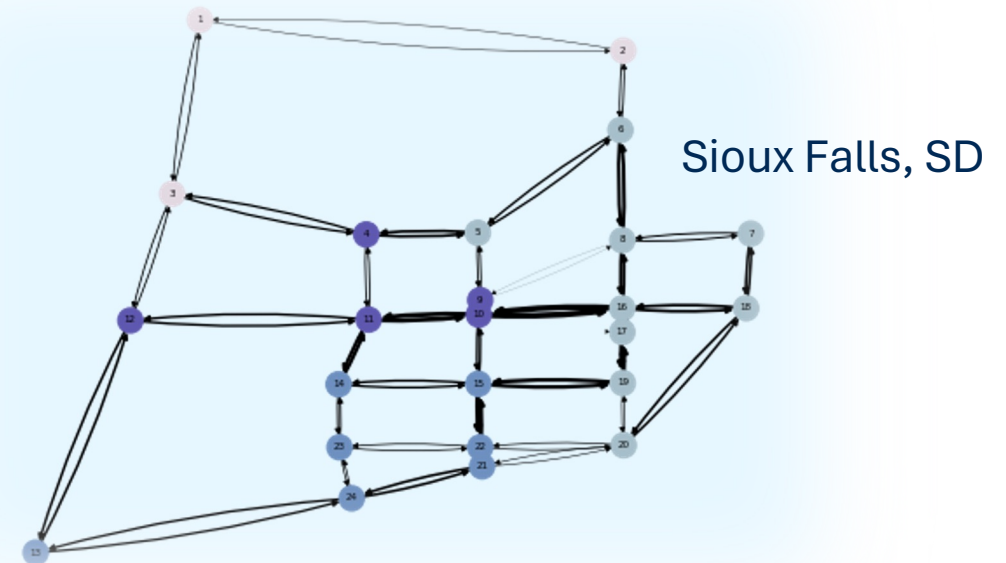
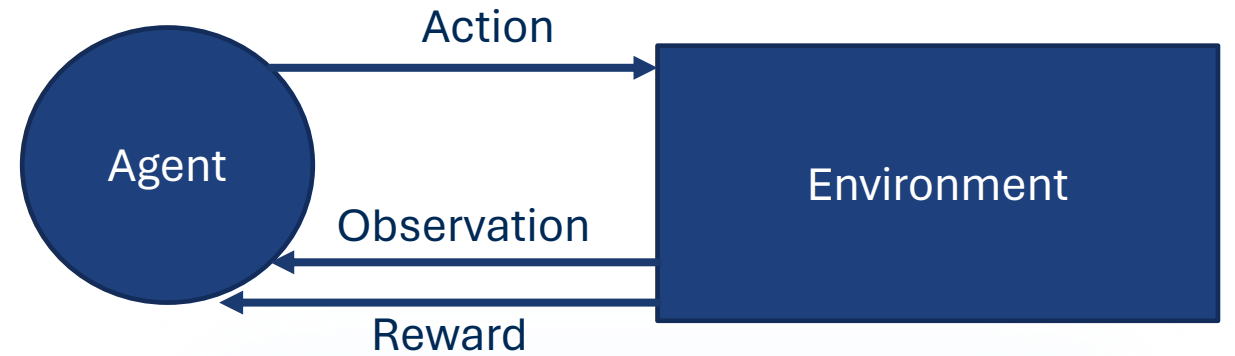
- Features for edge  $e$ 
  1. Number of vehicles that are at an intersection with an unperturbed shortest path to the destination that passes through  $e$
  2. Number of vehicles that are on an edge but will take  $e$  as the shortest path
  3. Number of vehicles that are at an intersection that will immediately take  $e$  as their shortest path without perturbation
  4. Number of vehicles currently on  $e$
  5. Sum of remaining travel times of vehicles currently on edge  $e$
- State represented as  $|E| \times 5$  vector



# ASSESSMENT THROUGH DEEP REINFORCEMENT LEARNING



- The attacker could output perturbations for **hundreds of city roads**
- General-purpose reinforcement learning algorithms (e.g., DDPG) are **infeasible** even for a small city
  - 24 nodes and 76 edges in Sioux Falls
  - Enormous action/observation space
- It requires **millions of samples** collected from the environment
- We need a robust and feasible attack oracle



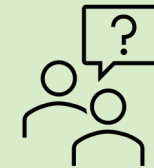
# HIERARCHICAL MULTI-AGENT REINFORCEMENT LEARNING



- The idea:
  - We can **divide** the network into smaller components
  - **Low-level** RL agents are assigned to each component
  - A **high-level** RL agent coordinates the low-level agents

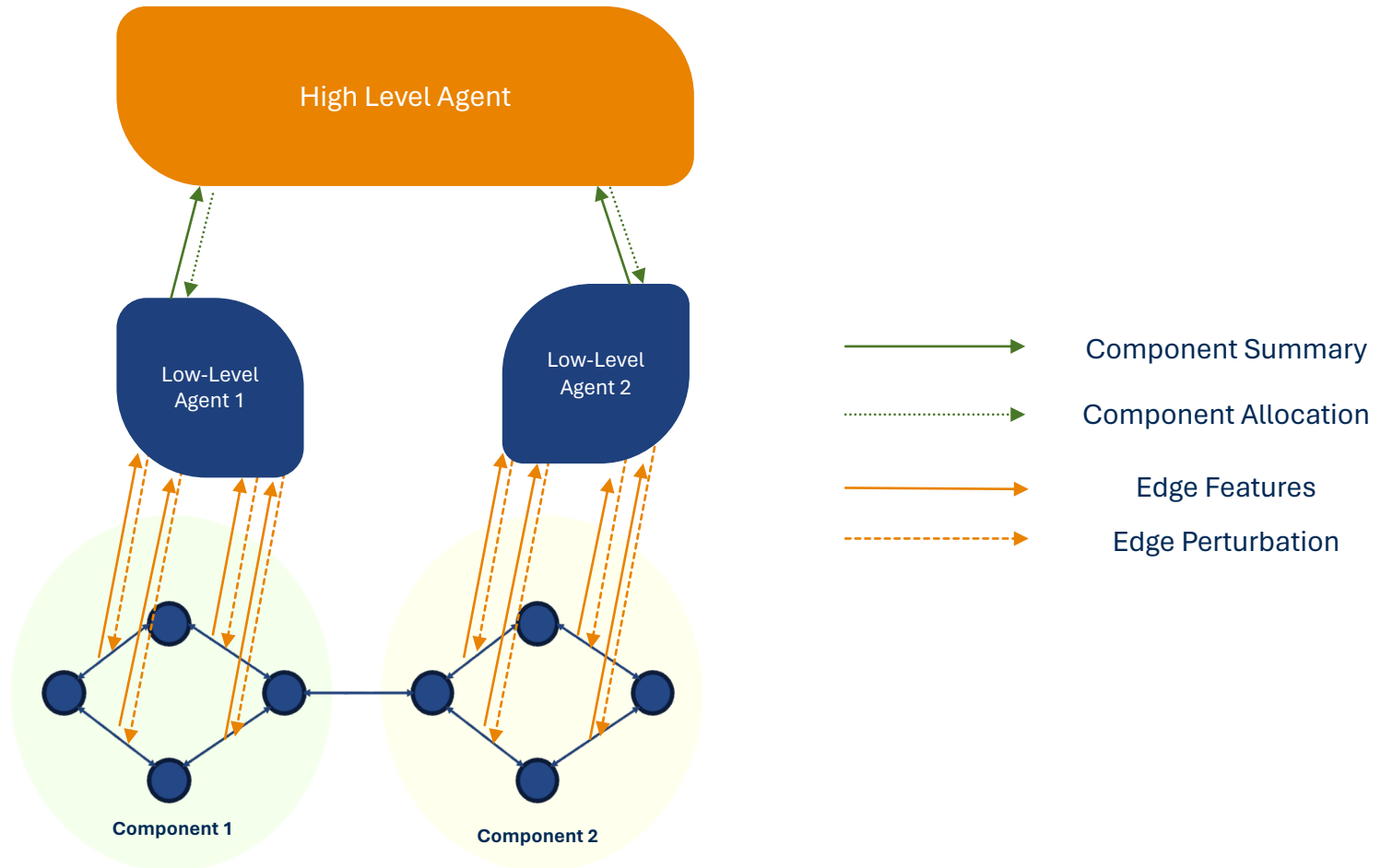


- Why a high-level coordinator?
  - The total perturbations are restricted by a budget
  - Low-level agents compete over the budget



- The high-level agent allocates the perturbation budget to the component agents
- The low-level agents distribute allocated perturbation budgets to road links

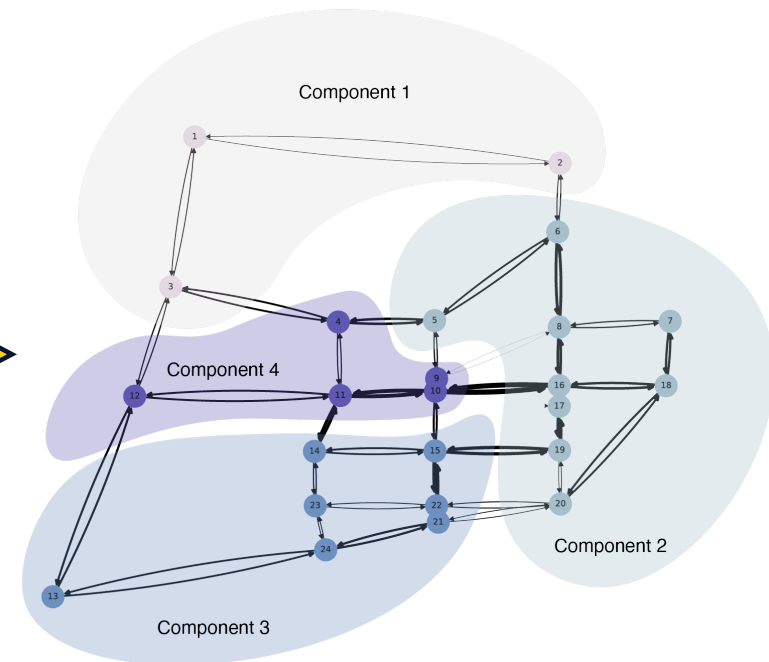
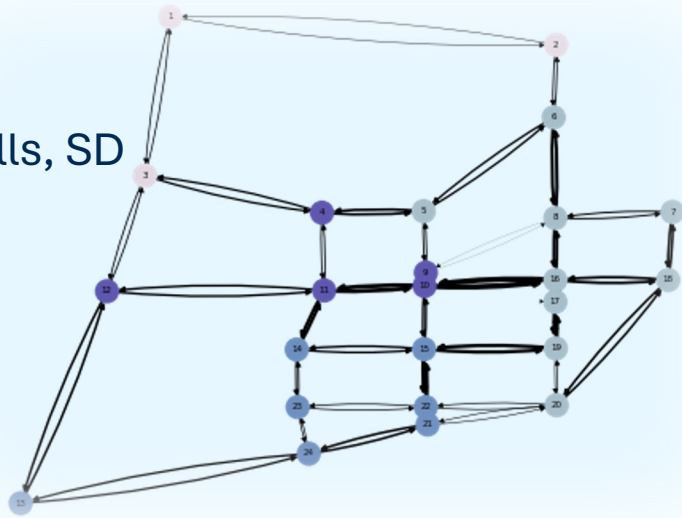
# HIERARCHICAL APPROACH



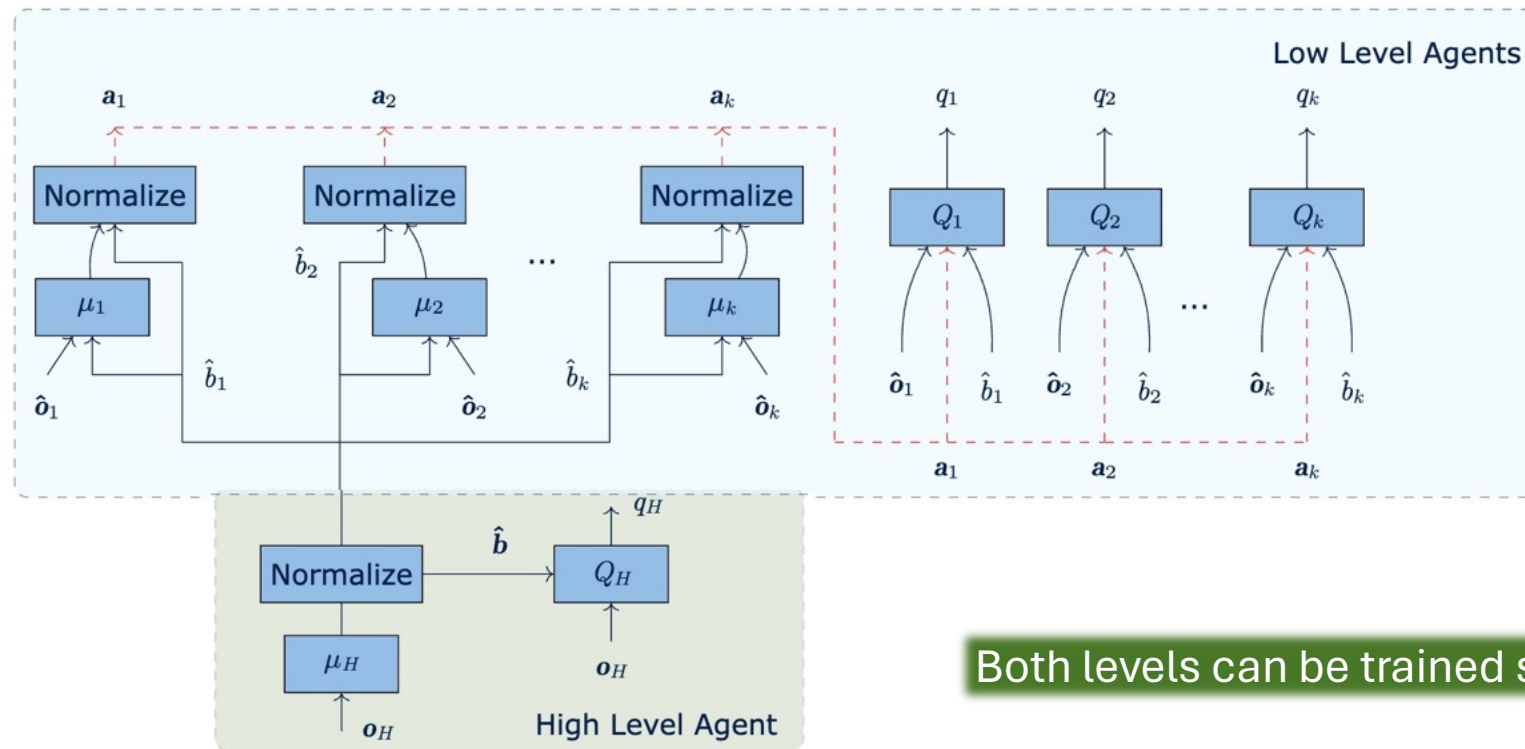
# NETWORK DECOMPOSITION

Decompose the network based on **K-means clustering** by edge distance (without congestion)

Sioux Falls, SD



# DISTRIBUTED LEARNING



Both levels can be trained simultaneously.

# EXPERIMENTAL SETUP

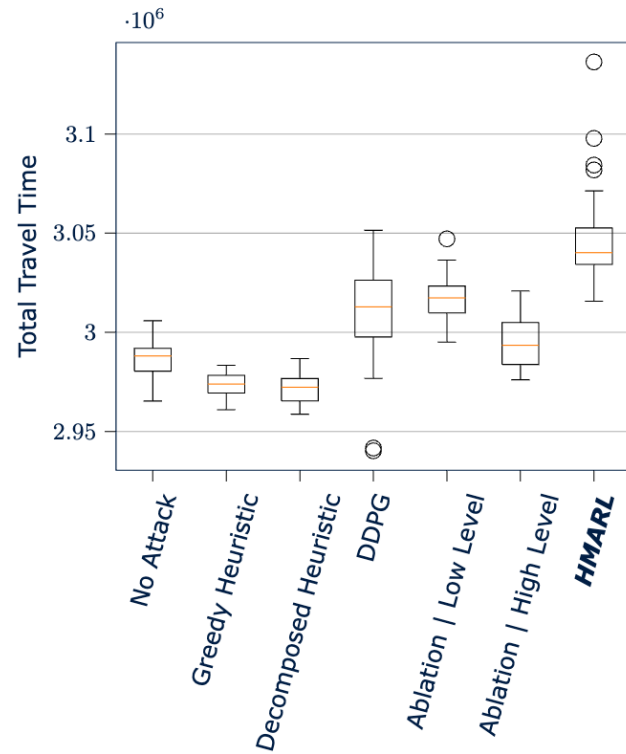


- Baselines
  - **Proportional High Level:** Allocates budget to each component based on proportion of vehicles in the component
  - **Greedy Low Level:** Perturbs edges by proportion of vehicles that pass through that edge
  - Random
  - DDPG without decomposition
- Hyperparameter search
  - Grid search

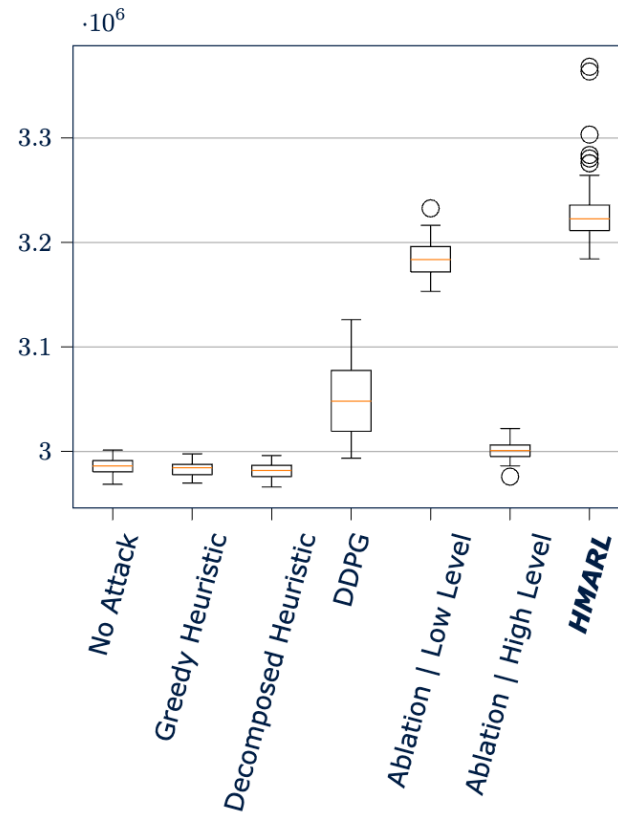
# EVALUATIONS



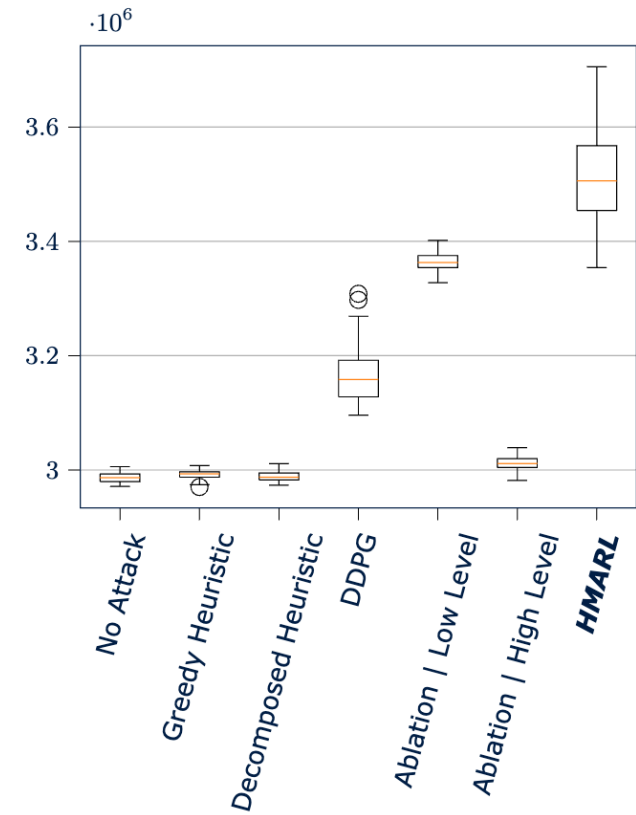
## Budget $B = 5$



## Budget $B = 10$



## Budget $B = 15$





# CONCLUSION



- We discussed the importance of **resiliency** of transportation networks
- We discussed how transportation networks are **vulnerable** to various attacks.
- We introduced a **model of false-data attacks against navigation** in transportation networks
- We proposed a computational method based on **multi-agent reinforcement learning** to assess against worst-case attacks
- We demonstrated the **effectiveness** of our framework on the Sioux Falls, SD benchmark network
- We showed that a **worst-case attack can increase total travel time by up to 50%**



THANKS FOR YOUR ATTENTION

Dr. Aron Laszka

[alaszka@psu.edu](mailto:alaszka@psu.edu)