Detection and Optimization Problems with Applications in Smart Cities

Jing Zhang
jzh@bu.edu, http://people.bu.edu/jzh/
https://github.com/jingzbu

Division of Systems Engineering,
and Center for Information and Systems Engineering
Boston University

December 26, 2016
Current Inefficient Transportation System

(a) Traffic jam during AM peak  
(b) Traffic jam during PM peak

Source: (a) http://www.bostonherald.com/news_opinion/local_coverage/2013/05/boston_traffic_ranked_10th_worst_in_us
(b) http://bostinno.streetwise.co/2014/06/29/boston-fourth-of-july-parking-restrictions-traffic/boston-fourth-of-july-parking-restrictions-traffic-2/
What Can We Do?

How can we make transportation systems more efficient, thus contributing to the smart city agenda?

- In a **fast time-scale**: detect atypical jams so as to provide most urgent targets for the Transportation department and/or emergency services to intervene
  - Statistical Anomaly Detection

- In a **slower time-scale**: define appropriate metrics to assess the performance of the road network
  - Price of Anarchy
  - prioritize road segments for intervention so as to mitigate congestion
  - Parameter Sensitivity Analysis
What Can We Do?

How can we make transportation systems more efficient, thus contributing to the smart city agenda?

- In a **fast time-scale**:
  - detect atypical jams so as to provide most urgent targets for the Transportation department and/or emergency services to intervene
    - Statistical Anomaly Detection

- In a **slower time-scale**:
  - define appropriate metrics to assess the performance of the road network
    - Price of Anarchy
  - prioritize road segments for intervention so as to mitigate congestion
    - Parameter Sensitivity Analysis
What Can We Do?

How can we make transportation systems more efficient, thus contributing to the smart city agenda?

- **In a fast time-scale:**
  - detect atypical jams so as to provide most urgent targets for the Transportation department and/or emergency services to intervene
    - Statistical Anomaly Detection

- **In a slower time-scale:**
  - define appropriate metrics to assess the performance of the road network
    - Price of Anarchy
  - prioritize road segments for intervention so as to mitigate congestion
    - Parameter Sensitivity Analysis
What Can We Do?

How can we make transportation systems more efficient, thus contributing to the smart city agenda?

- In a **fast time-scale**:
  - detect atypical jams so as to provide most urgent targets for the Transportation department and/or emergency services to intervene
    - Statistical Anomaly Detection

- In a **slower time-scale**:
  - define appropriate metrics to assess the performance of the road network
    - Price of Anarchy
  - prioritize road segments for intervention so as to mitigate congestion
    - Parameter Sensitivity Analysis
What Can We Do?

How can we make transportation systems more efficient, thus contributing to the smart city agenda?

- In a **fast time-scale**:
  - detect atypical jams so as to provide most urgent targets for the Transportation department and/or emergency services to intervene
    - Statistical Anomaly Detection

- In a **slower time-scale**:
  - define appropriate metrics to assess the performance of the road network
    - Price of Anarchy
  - prioritize road segments for intervention so as to mitigate congestion
    - Parameter Sensitivity Analysis
Outline

Fast time-scale results (detection)
  - Theoretical
  - Numerical

Slower time-scale results (optimization)
  - Theoretical
  - Numerical

Discussions
Outline

- Fast time-scale results (detection)
  - Theoretical
  - Numerical
- Slower time-scale results (optimization)
  - Theoretical
  - Numerical
- Discussions
Outline

- Fast time-scale results (detection)
  - Theoretical
  - Numerical
- Slower time-scale results (optimization)
  - Theoretical
  - Numerical
- Discussions
Outline

- Fast time-scale results (detection)
  - Theoretical
  - Numerical
- Slower time-scale results (optimization)
  - Theoretical
  - Numerical
- Discussions
Outline

- Fast time-scale results (detection)
  - Theoretical
  - Numerical
- Slower time-scale results (optimization)
  - Theoretical
  - Numerical
- Discussions
Outline

- Fast time-scale results (detection)
  - Theoretical
  - Numerical
- Slower time-scale results (optimization)
  - Theoretical
  - Numerical
- Discussions
Outline

- Fast time-scale results (detection)
  - Theoretical
  - Numerical
- Slower time-scale results (optimization)
  - Theoretical
  - Numerical
- Discussions
A Generic Anomaly Detection Scheme

- Characterize system behavior
- Apply anomaly detection algorithm

Real-World System

Characterized as Random Time Series

Continuous Features

Discrete Features

Quantization

Time Series on Finite Alphabet

Time Series (Reference)  Time Series (Test)

Reference PL  Test PL

Threshold  Divergence

Abnormal?

YES  NO

(Optional) Find Culprits
Key Technical Contribution

- Composite Hypothesis Testing: represent uncertainty by specifying a family of possible models for each hypothesis

\[ \mathcal{H}_0 : X \sim p_0(x | \theta_0), \theta_0 \in \Theta_0 \]
\[ \mathcal{H}_1 : X \sim p_1(x | \theta_1), \theta_1 \in \Theta_1 \]

- Hoeffding’s test — a widely used test for deciding to accept \( \mathcal{H}_0 \) or not; \( \Theta_1 = \Theta_0^c \)

  - Requires a threshold, obtained using Large Deviation Theory (LDT), leading to too many false alarms
  - We develop an alternative threshold-setting approach, using weak convergence results
Key Technical Contribution

- **Composite Hypothesis Testing**: represent uncertainty by specifying a family of possible models for each hypothesis

\[
\mathcal{H}_0 : X \sim p_0(x | \theta_0), \theta_0 \in \Theta_0 \\
\mathcal{H}_1 : X \sim p_1(x | \theta_1), \theta_1 \in \Theta_1
\]

- **Hoeffding’s test** — a widely used test for deciding to accept \(\mathcal{H}_0\) or not; \(\Theta_1 = \Theta_0^c\)
  
  - Requires a threshold, obtained using Large Deviation Theory (LDT), leading to **too many false alarms**
  - We develop an alternative threshold-setting approach, using weak convergence results
    
    - Extending work by Unnikrishnan & Huang, 2013, from i.i.d. to Markovian
Key Technical Contribution

- Composite Hypothesis Testing: represent uncertainty by specifying a family of possible models for each hypothesis

  \[ \mathcal{H}_0 : X \sim p_0(x | \theta_0) , \theta_0 \in \Theta_0 \]
  \[ \mathcal{H}_1 : X \sim p_1(x | \theta_1) , \theta_1 \in \Theta_1 \]

- Hoeffding’s test — a widely used test for deciding to accept \( \mathcal{H}_0 \) or not; \( \Theta_1 = \Theta_0^c \)

  - Requires a threshold, obtained using Large Deviation Theory (LDT), leading to too many false alarms
  
  - We develop an alternative threshold-setting approach, using weak convergence results

    Extending work by Unnikrishnan & Huang, 2013, from i.i.d. to Markovian
Key Technical Contribution

- Composite Hypothesis Testing: represent uncertainty by specifying a family of possible models for each hypothesis

\[ H_0 : X \sim p_0(x | \theta_0), \theta_0 \in \Theta_0 \]
\[ H_1 : X \sim p_1(x | \theta_1), \theta_1 \in \Theta_1 \]

- Hoeffding’s test — a widely used test for deciding to accept \( H_0 \) or not; \( \Theta_1 = \Theta_0^c \)

- Requires a threshold, obtained using Large Deviation Theory (LDT), leading to too many false alarms
- We develop an alternative threshold-setting approach, using weak convergence results
  - Extending work by Unnikrishnan & Huang, 2013, from i.i.d. to Markovian
Key Technical Contribution

- **Composite Hypothesis Testing:** represent uncertainty by specifying a family of possible models for each hypothesis

  \[ H_0 : X \sim p_0 \left( x \mid \theta_0 \right), \theta_0 \in \Theta_0 \]

  \[ H_1 : X \sim p_1 \left( x \mid \theta_1 \right), \theta_1 \in \Theta_1 \]

- **Hoeffding’s test** — a widely used test for deciding to accept \( H_0 \) or not; \( \Theta_1 = \Theta_0^c \)

  - Requires a threshold, obtained using **Large Deviation Theory (LDT)**, leading to **too many false alarms**
  - We develop an alternative threshold-setting approach, using **weak convergence** results
    - Extending work by Unnikrishnan & Huang, 2013, from i.i.d. to Markovian
The Markov Model

The Markov model:
- $\Xi = \{\xi_i; \ i = 1, \ldots, N\}$: finite alphabet (states)
- Sequence of states $Y = \{Y_l; \ l = 0, 1, 2, \ldots\}$ correspond to observations about the system
- **Null hypothesis** $H$: $Y$ forms a Markov chain over $\Xi$ with transition matrix $Q = (q_{ij})_{i,j=1}^N$
The Markov Model

The Markov model:

- $\Xi = \{\xi_i; \ i = 1, \ldots, N\}$: finite alphabet (states)
- Sequence of states $Y = \{Y_l; \ l = 0, 1, 2, \ldots\}$ correspond to observations about the system
- Null hypothesis $H$: $Y$ forms a Markov chain over $\Xi$ with transition matrix $Q = (q_{ij})_{i,j=1}^N$
The Markov Model

The Markov model:

- $\Xi = \{\xi_i; i = 1, \ldots, N\}$: finite alphabet (states)
- Sequence of states $Y = \{Y_l; l = 0, 1, 2, \ldots\}$ correspond to observations about the system
- **Null hypothesis** $H$: $Y$ forms a Markov chain over $\Xi$ with transition matrix $Q = (q_{ij})_{i,j=1}^N$
The Markov Model (cont.)

**Transformed Markov model:**

- \( \mathbf{Z} = \{ Z_l = (Y_{l-1}, Y_l); l = 1, 2, \ldots \} \): pairs of consecutive states
- \( \Theta = \Xi \times \Xi = \{ \theta_{ij} = (\xi_i, \xi_j); i, j = 1, \ldots, N \} \): state space
- \( \mathcal{P}(\Theta) \): the set of Probability Laws (PLs) on \( \Theta \)
- Under \( \mathcal{H} \), \( \mathbf{Z} \) forms a Markov chain evolving on \( \Theta \), with transition matrix \( \mathbf{P} = (p_{ij})_{N^2}^{i,j=1} \)
- One can obtain \( \mathbf{P} \) directly from \( \mathbf{Q} \), \( p(\theta_{ij} | \theta_{kl}) = 1_{\{i=l\}} q_{ij} \), e.g.,

\[
\mathbf{Q} = \begin{bmatrix} 0.2 & 0.8 \\ 0.7 & 0.3 \end{bmatrix} \Rightarrow \mathbf{P} = \begin{bmatrix} 0.2 & 0.8 & 0 & 0 \\ 0 & 0 & 0.7 & 0.3 \\ 0.2 & 0.8 & 0 & 0 \\ 0 & 0 & 0.7 & 0.3 \end{bmatrix}
\]
The Markov Model (cont.)

Transformed Markov model:
- $\mathbf{Z} = \{ Z_l = (Y_{l-1}, Y_l); \ l = 1, 2, \ldots \}$: pairs of consecutive states
- $\Theta = \Xi \times \Xi = \{ \theta_{ij} = (\xi_i, \xi_j); \ i, j = 1, \ldots, N \}$: state space
- $\mathcal{P}(\Theta)$: the set of Probability Laws (PLs) on $\Theta$
- Under $H$, $\mathbf{Z}$ forms a Markov chain evolving on $\Theta$, with transition matrix $\mathbf{P} = (p_{ij})_{i,j=1}^{N^2}$
- One can obtain $\mathbf{P}$ directly from $\mathbf{Q}$, $p(\theta_{ij} | \theta_{kl}) = 1_{\{i=l\}} q_{ij}$, e.g.,

$$
\mathbf{Q} = \begin{bmatrix}
0.2 & 0.8 \\
0.7 & 0.3
\end{bmatrix} \Rightarrow \mathbf{P} = \begin{bmatrix}
0.2 & 0.8 & 0 & 0 \\
0 & 0 & 0.7 & 0.3 \\
0.2 & 0.8 & 0 & 0 \\
0 & 0 & 0.7 & 0.3
\end{bmatrix}
$$
The Markov Model (cont.)

Transformed Markov model:

- \( Z = \{Z_l = (Y_{l-1}, Y_l); \ l = 1, 2, \ldots \} \): pairs of consecutive states
- \( \Theta = \Xi \times \Xi = \{\theta_{ij} = (\xi_i, \xi_j); \ i, j = 1, \ldots, N\} \): state space
- \( \mathcal{P}(\Theta) \): the set of Probability Laws (PLs) on \( \Theta \)
- Under \( \mathcal{H} \), \( Z \) forms a Markov chain evolving on \( \Theta \), with transition matrix \( P = (p_{ij})_{i,j=1}^{N^2} \)
- One can obtain \( P \) directly from \( Q \), \( p(\theta_{ij} | \theta_{kl}) = 1_{\{i=l\}} q_{ij} \), e.g.,

\[
Q = \begin{bmatrix}
0.2 & 0.8 \\
0.7 & 0.3
\end{bmatrix}
\Rightarrow P = \begin{bmatrix}
0.2 & 0.8 & 0 & 0 \\
0 & 0 & 0.7 & 0.3 \\
0.2 & 0.8 & 0 & 0 \\
0 & 0 & 0.7 & 0.3
\end{bmatrix}
\]
The Markov Model (cont.)

**Transformed Markov model:**

- \( Z = \{ Z_l = (Y_{l-1}, Y_l); \ l = 1, 2, \ldots \} \): pairs of consecutive states
- \( \Theta = \Xi \times \Xi = \{ \theta_{ij} = (\xi_i, \xi_j); \ i, j = 1, \ldots, N \} \): state space
- \( \mathcal{P}(\Theta) \): the set of **Probability Laws (PLs)** on \( \Theta \)
- Under \( \mathcal{H} \), \( Z \) forms a Markov chain evolving on \( \Theta \), with transition matrix \( P = (p_{ij})_{i,j=1}^{N^2} \)

One can obtain \( P \) directly from \( Q \), \( p(\theta_{ij}|\theta_{kl}) = 1_{\{i=l\}}q_{ij} \), e.g.,

\[
Q = \begin{bmatrix}
0.2 & 0.8 \\
0.7 & 0.3
\end{bmatrix} \quad \Rightarrow \quad P = \begin{bmatrix}
0.2 & 0.8 & 0 & 0 \\
0 & 0 & 0.7 & 0.3 \\
0.2 & 0.8 & 0 & 0 \\
0 & 0 & 0.7 & 0.3
\end{bmatrix}
\]
The Markov Model (cont.)

Transformed Markov model:
- $$Z = \{ Z_l = (Y_{l-1}, Y_l); \ l = 1, 2, \ldots \}$$: pairs of consecutive states
- $$\Theta = \Xi \times \Xi = \{ \theta_{ij} = (\xi_i, \xi_j); \ i, j = 1, \ldots, N \}$$: state space
- $$\mathcal{P}(\Theta)$$: the set of Probability Laws (PLs) on $$\Theta$$
- Under $$\mathcal{H}$$, $$Z$$ forms a Markov chain evolving on $$\Theta$$, with transition matrix $$P = (p_{ij})_{i,j=1}^{N^2}$$
- One can obtain $$P$$ directly from $$Q$$, $$p(\theta_{ij} | \theta_{kl}) = 1_{\{i=l\}} q_{ij}$$, e.g.,

$$Q = \begin{bmatrix} 0.2 & 0.8 \\ 0.7 & 0.3 \end{bmatrix} \Rightarrow P = \begin{bmatrix} 0.2 & 0.8 & 0 & 0 \\ 0 & 0 & 0.7 & 0.3 \\ 0.2 & 0.8 & 0 & 0 \\ 0 & 0 & 0.7 & 0.3 \end{bmatrix}$$
Assumption

Assume the chain $Z$ to be aperiodic, irreducible, and positive recurrent, with stationary distribution

$$\pi = (\pi_{ij}; \ i, j = 1, \ldots, N) = (\tilde{\pi}_k; \ k = 1, \ldots, N^2),$$

where $\pi_{ij}$ denotes the probability of seeing $\theta_{ij}$.

Note that $\tilde{\pi}_1 = \pi_{11}, \ldots, \tilde{\pi}_N = \pi_{1N}, \ldots, \tilde{\pi}_{(N-1)N+1} = \pi_{N1}, \ldots, \tilde{\pi}_{N^2} = \pi_{NN}$.
Definitions

- **Relative entropy** (i.e., *divergence*) between $\nu$ and $\pi$

\[
D(\nu \parallel \pi) = \sum_{i=1}^{N} \sum_{j=1}^{N} \nu_{ij} \log \frac{\nu_{ij}}{\sum_{t=1}^{N} \nu_{it}} - \log \frac{\sum_{t=1}^{N} \nu_{it}}{\sum_{t=1}^{N} \pi_{it}}
\]

- Define the following based on the first $n$ samples in $Z$
  - empirical PL $\Gamma_n$: $\Gamma_n(\theta_{ij}) = \frac{1}{n} \sum_{l=1}^{n} 1_{Z_l = \theta_{ij}}$
  - empirical measure: $U_n = \sqrt{n}(\Gamma_n - \pi)$

- **Hoeffding’s test**
  - $\mathcal{X}_n$, the output of the test that decides to accept or to reject $\mathcal{X}$ based on the first $n$ samples in $Z$
  - $\mathcal{X}_n$ rejects $\mathcal{X}$ if $D(\Gamma_n \parallel \pi) > \eta_n$, where $\eta_n$ is a threshold
  - false alarm probability $\beta_n = P(\mathcal{X}_n = \text{reject} | \text{true null})$
Definitions

- **Relative entropy** (i.e., **divergence**) between \( \nu \) and \( \pi \)

\[
D(\nu \parallel \pi) = \sum_{i=1}^{N} \sum_{j=1}^{N} \nu_{ij} \log \frac{\nu_{ij}}{\sum_{t=1}^{N} \nu_{it} \pi_{ij} / \sum_{t=1}^{N} \pi_{it}}
\]

- Define the following based on the first \( n \) samples in \( Z \)
  - **empirical PL** \( \Gamma_n \): 
    \[
    \Gamma_n(\theta_{ij}) = \frac{1}{n} \sum_{l=1}^{n} 1\{Z_l=\theta_{ij}\}
    \]
  - **empirical measure**: 
    \[
    U_n = \sqrt{n} (\Gamma_n - \pi)
    \]
  - **Hoeffding’s test**
    \[
    H_n:\text{ the output of the test that decides to accept or to reject } \pi \text{ based on the first } n \text{ samples in } Z
    \]
    \[
    H_n \text{ rejects } \pi \text{ if } D(\Gamma_n \parallel \pi) > \eta_n, \text{ where } \eta_n \text{ is a threshold}
    \]
    \[
    \beta_n = P_{H_0} \{D(\Gamma_n \parallel \pi) > \eta_n\}\]
Definitions

- **Relative entropy** (i.e., divergence) between \( \nu \) and \( \pi \)

\[
D(\nu \| \pi) = \sum_{i=1}^{N} \sum_{j=1}^{N} \nu_{ij} \log \frac{\nu_{ij}}{\sum_{t=1}^{N} \nu_{it}} / \sum_{t=1}^{N} \pi_{it}
\]

- Define the following based on the first \( n \) samples in \( Z \)
  - **empirical PL** \( \Gamma_n \): \( \Gamma_n (\theta_{ij}) = \frac{1}{n} \sum_{l=1}^{n} 1\{Z_l = \theta_{ij}\} \)
  - **empirical measure**: \( U_n = \sqrt{n} (\Gamma_n - \pi) \)

- **Hoeffding’s test**
  - \( H_n \) is the output of the test that decides to accept or to reject \( H \) based on the first \( n \) samples in \( Z \)
  - \( H_n \) rejects \( H \) if \( D(\Gamma_n \| \pi) > \eta_n \), where \( \eta_n \) is a threshold
  - false alarm probability: \( \beta_n = P_{H_n} (D(\Gamma_n \| \pi) > \eta_n) \)
Definitions

- **Relative entropy** (i.e., divergence) between \( \nu \) and \( \pi \)

\[
D(\nu \parallel \pi) = \sum_{i=1}^{N} \sum_{j=1}^{N} \nu_{ij} \log \frac{\nu_{ij}}{\frac{\sum_{t=1}^{N} \nu_{it}}{\frac{\sum_{t=1}^{N} \pi_{it}}{}}}\]

- Define the following based on the first \( n \) samples in \( Z \)
  - **empirical PL** \( \Gamma_n \):
    \[
    \Gamma_n (\theta_{ij}) = \frac{1}{n} \sum_{l=1}^{n} 1\{Z_l = \theta_{ij}\}
    \]
  - **empirical measure**:
    \[
    U_n = \sqrt{n} (\Gamma_n - \pi)
    \]

- **Hoeffding’s test**
  - \( \mathcal{H}_n \): the output of the test that decides to accept or to reject \( \mathcal{H} \) based on the first \( n \) samples in \( Z \)
  - \( \mathcal{H}_n \) rejects \( \mathcal{H} \) if \( D(\Gamma_n \parallel \pi) > \eta_n \), where \( \eta_n \) is a threshold
  - false alarm probability:
    \[
    \beta_n = P(\mathcal{H}_n > \eta_n) | \mathcal{H} \text{ is false}
    \]
Definitions

- **Relative entropy (i.e., divergence)** between $\nu$ and $\pi$

\[
D(\nu \parallel \pi) = \sum_{i=1}^{N} \sum_{j=1}^{N} \nu_{ij} \log \left( \frac{\nu_{ij}}{\sum_{t=1}^{N} \nu_{it}} \right) \quad \frac{\nu_{ij}}{\sum_{t=1}^{N} \pi_{ij}} \sum_{t=1}^{N} \pi_{it}
\]

- Define the following based on the first $n$ samples in $Z$
  - **empirical PL $\Gamma_n$**: $\Gamma_n (\theta_{ij}) = \frac{1}{n} \sum_{l=1}^{n} 1\{Z_l = \theta_{ij}\}$
  - **empirical measure**: $U_n = \sqrt{n} (\Gamma_n - \pi)$

- **Hoeffding’s test**
  - $\mathcal{H}_n$: the output of the test that decides to accept or to reject $\mathcal{H}$ based on the first $n$ samples in $Z$
  - $\mathcal{H}_n$ rejects $\mathcal{H}$ iff $D(\Gamma_n \parallel \pi) > \eta_n$, where $\eta_n$ is a threshold
  - **false alarm probability**: $\beta_n = \mathbb{P}_{\mathcal{H}} \{D(\Gamma_n \parallel \pi) > \eta_n\}$
Definitions

- **Relative entropy** (i.e., divergence) between $\nu$ and $\pi$

\[
D(\nu \parallel \pi) = \sum_{i=1}^{N} \sum_{j=1}^{N} \nu_{ij} \log \frac{\nu_{ij}}{\pi_{ij}} \sum_{t=1}^{N} \nu_{it} \pi_{ij} \sum_{t=1}^{N} \pi_{it}
\]

- Define the following based on the first $n$ samples in $Z$
  - **empirical PL** $\Gamma_n$: $\Gamma_n(\theta_{ij}) = \frac{1}{n} \sum_{l=1}^{n} 1\{Z_l = \theta_{ij}\}$
  - **empirical measure**: $U_n = \sqrt{n} (\Gamma_n - \pi)$

- **Hoeffding's test**
  - $\mathcal{H}_n$: the output of the test that decides to accept or to reject $\mathcal{H}$ based on the first $n$ samples in $Z$
  - $\mathcal{H}_n$ rejects $\mathcal{H}$ iff $D(\Gamma_n \parallel \pi) > \eta_n$, where $\eta_n$ is a threshold
  - **false alarm probability**: $\beta_n = \mathbb{P}_{\mathcal{H}} \{D(\Gamma_n \parallel \pi) > \eta_n\}$
**Definitions**

- **Relative entropy** (i.e., *divergence*) between $\nu$ and $\pi$
  
  $$D(\nu \parallel \pi) = \sum_{i=1}^{N} \sum_{j=1}^{N} \nu_{ij} \log \frac{\nu_{ij}}{\nu_{it} \pi_{ij} / \sum_{t=1}^{N} \pi_{it}}$$

- Define the following based on the first $n$ samples in $\mathbf{Z}$
  - **empirical PL $\Gamma_n$:** $\Gamma_n (\theta_{ij}) = \frac{1}{n} \sum_{l=1}^{n} 1\{Z_l = \theta_{ij}\}$
  - **empirical measure:** $U_n = \sqrt{n} (\Gamma_n - \pi)$

- **Hoeffdind’s test**
  - $\mathcal{H}_n$: the output of the test that decides to accept or to reject $\mathcal{H}$ based on the first $n$ samples in $\mathbf{Z}$
  - $\mathcal{H}_n$ rejects $\mathcal{H}$ iff $D(\Gamma_n \parallel \pi) > \eta_n$, where $\eta_n$ is a **threshold**
  - **false alarm probability:** $\beta_n = \mathbb{P}_{\mathcal{H}_n} \{D(\Gamma_n \parallel \pi) > \eta_n\}$
Definitions

- **Relative entropy** (i.e., divergence) between $\nu$ and $\pi$

$$D(\nu \| \pi) = \sum_{i=1}^{N} \sum_{j=1}^{N} \nu_{ij} \log \frac{\nu_{ij}}{\frac{\sum_{t=1}^{N} \nu_{it}}{\frac{\sum_{t=1}^{N} \pi_{ij}}{\sum_{t=1}^{N} \pi_{it}}}}$$

- Define the following based on the first $n$ samples in $Z$
  - **empirical PL $\Gamma_n$**: $\Gamma_n(\theta_{ij}) = \frac{1}{n} \sum_{l=1}^{n} 1\{Z_l = \theta_{ij}\}$
  - **empirical measure**: $U_n = \sqrt{n} \left( \Gamma_n - \pi \right)$

- **Hoeffding’s test**
  - $\mathcal{H}_n$: the output of the test that decides to accept or to reject $\mathcal{H}$ based on the first $n$ samples in $Z$
  - $\mathcal{H}_n$ rejects $\mathcal{H}$ iff $D(\Gamma_n \| \pi) > \eta_n$, where $\eta_n$ is a threshold
  - **false alarm probability**: $\beta_n = P_{\mathcal{H}} \left\{ D(\Gamma_n \| \pi) > \eta_n \right\}$
Existing Results

- Hoeffding’s test satisfies asymptotic Neyman-Pearson optimality
- Sanov’s theorem yields an approximation for $\eta_n$

$$\eta_n^{sv} \approx -\frac{1}{n} \log (\beta_n)$$

- Existing work involves refinements of the LDT results, but such refinements are typically hard to compute

--- earlier work

Existing Results

- Hoeffding’s test satisfies asymptotic Neyman-Pearson optimality
- **Sanov’s theorem** yields an approximation for $\eta_n$

$$\eta_{sv} \approx -\frac{1}{n} \log (\beta_n)$$

--- earlier work


- Existing work also involves refinements of the LDT results, but such refinements are typically hard to compute

Existing Results

- Hoeffding’s test satisfies asymptotic Neyman-Pearson optimality
- **Sanov’s theorem** yields an approximation for $\eta_n$

$$\eta_{sv} \approx -\frac{1}{n} \log (\beta_n)$$

--- *earlier work*


- Existing work also involves refinements of the LDT results, but such refinements are typically hard to compute

Existing Results

- Hoeffding’s test satisfies asymptotic Neyman-Pearson optimality
- **Sanov’s theorem** yields an approximation for $\eta_n$

$$\eta_{sv} \approx -\frac{1}{n} \log (\beta_n)$$


- Existing work also involves refinements of the LDT results, but such refinements are typically hard to compute
Existing Results

- Hoeffding’s test satisfies asymptotic Neyman-Pearson optimality
- **Sanov’s theorem** yields an approximation for $\eta_n$

\[
\eta_n^{sv} \approx -\frac{1}{n} \log (\beta_n)
\]

←-- earlier work


- Existing work also involves refinements of the LDT results, but such refinements are typically hard to compute

Existing Results

- Hoeffding’s test satisfies asymptotic Neyman-Pearson optimality
- **Sanov’s theorem** yields an approximation for $\eta_n$

$$\eta_n^{sv} \approx -\frac{1}{n} \log (\beta_n) \quad \leftarrow \text{earlier work}$$


- Existing work also involves refinements of the LDT results, but such refinements are typically hard to compute

Recover Transition Matrix from Stationary Distribution

Lemma

Under the Markovian assumption, we have

$$\frac{\pi_{ij}}{\sum_{t=1}^{N} \pi_{it}} = \frac{\pi_{ij}}{\sum_{t=1}^{N} \pi_{ti}} = q_{ij}, \quad i, j = 1, \ldots, N.$$
A Central Limit Theorem (CLT)

**Theorem**

A CLT holds for $U_n = \sqrt{n} (\Gamma_n - \pi)$, that is, $U_n \xrightarrow{d.} \mathcal{N} (0, \Lambda)$ with

$$\Lambda = [\Lambda^{(i,j)}]_{i,j=1}^{N^2}$$

given by

$$\Lambda^{(i,j)} = \tilde{\pi}_i (I_{ij} - \tilde{\pi}_j) + \sum_{m=1}^{\infty} \left[ \tilde{\pi}_i (P_{ij}^m - \tilde{\pi}_j) + \tilde{\pi}_j (P_{ji}^m - \tilde{\pi}_i) \right].$$

**Taylor expansion:**

- Letting $h(\nu) \overset{def}{=} D(\nu \| \pi)$, $D(\Gamma_n \| \pi) \approx \frac{1}{2n} U_n' \nabla^2 h(\pi) U_n$

**Approximation for $\eta_n$:**

- Generate samples of $\mathcal{N}(0, \Lambda)$, hence, obtaining approximate samples of $D(\Gamma_n \| \pi)$
- Compute an Empirical Cumulative Distribution Function, denoted $F_{\text{em}}$, of $D(\Gamma_n \| \pi)$

$$\eta_{wc,n} \approx F_{\text{em}}^{-1}(1 - \beta_n; n)$$

---

Taylor expansion:

- Letting $h(\nu) \overset{def}{=} D(\nu \| \pi)$, $D(\Gamma_n \| \pi) \approx \frac{1}{2n} U_n' \nabla^2 h(\pi) U_n$

**Approximation for $\eta_n$:**

- Generate samples of $\mathcal{N}(0, \Lambda)$, hence, obtaining approximate samples of $D(\Gamma_n \| \pi)$
- Compute an Empirical Cumulative Distribution Function, denoted $F_{\text{em}}$, of $D(\Gamma_n \| \pi)$

$$\eta_{wc,n} \approx F_{\text{em}}^{-1}(1 - \beta_n; n)$$

---

Taylor expansion:

- Letting $h(\nu) \overset{def}{=} D(\nu \| \pi)$, $D(\Gamma_n \| \pi) \approx \frac{1}{2n} U_n' \nabla^2 h(\pi) U_n$

**Approximation for $\eta_n$:**

- Generate samples of $\mathcal{N}(0, \Lambda)$, hence, obtaining approximate samples of $D(\Gamma_n \| \pi)$
- Compute an Empirical Cumulative Distribution Function, denoted $F_{\text{em}}$, of $D(\Gamma_n \| \pi)$

$$\eta_{wc,n} \approx F_{\text{em}}^{-1}(1 - \beta_n; n)$$
A Central Limit Theorem (CLT)

**Theorem**

A CLT holds for $U_n = \sqrt{n} (\Gamma_n - \pi)$, that is, $U_n \xrightarrow{d.} \mathcal{N} (0, \Lambda)$ with $\Lambda = \left[ \Lambda^{(i,j)} \right]^{N^2}_{i,j=1}$ given by

$$\Lambda^{(i,j)} = \tilde{\pi}_i (I_{ij} - \tilde{\pi}_j) + \sum_{m=1}^{\infty} \left[ \tilde{\pi}_i (P_{ij}^m - \tilde{\pi}_j) + \tilde{\pi}_j (P_{ji}^m - \tilde{\pi}_i) \right].$$

- **Taylor expansion:**
  - Letting $h(\nu) \overset{def}{=} D (\nu \| \pi)$, $D (\Gamma_n \| \pi) \approx \frac{1}{2n} U'_n \nabla^2 h (\pi) U_n$

- **Approximation for $\eta_n$:**
  - Generate samples of $\mathcal{N} (0, \Lambda)$, hence, obtaining approximate samples of $D (\Gamma_n \| \pi)$
  - Compute an Empirical Cumulative Distribution Function, denoted $F_{em} (\cdot ; n)$, of $D (\Gamma_n \| \pi)$

---

13/46 Jing Zhang, Boston University
Christmas Talk at Zhejiang University: Fast
A Central Limit Theorem (CLT)

**Theorem**

A CLT holds for \( U_n = \sqrt{n} (\Gamma_n - \pi) \), that is, \( U_n \xrightarrow{d.\ n \to \infty} \mathcal{N} (0, \Lambda) \)

with \( \Lambda = [\Lambda^{(i,j)}]_{i,j=1}^{N^2} \) given by

\[
\Lambda^{(i,j)} = \tilde{\pi}_i (I_{ij} - \tilde{\pi}_j) + \sum_{m=1}^\infty \left[ \tilde{\pi}_i (P_{mj}^m - \tilde{\pi}_j) + \tilde{\pi}_j (P_{ij}^m - \tilde{\pi}_i) \right].
\]

- **Taylor expansion:**
  - Letting \( h(\nu) \overset{\text{def}}{=} D(\nu \parallel \pi) \), \( D(\Gamma_n \parallel \pi) \approx \frac{1}{2n} U'_n \nabla^2 h(\pi) U_n \)

- **Approximation for \( \eta_n \):**
  - Generate samples of \( \mathcal{N} (0, \Lambda) \), hence, obtaining approximate samples of \( D(\Gamma_n \parallel \pi) \)
  - Compute an **Empirical Cumulative Distribution Function**, denoted \( F_{em}(\cdot; n) \), of \( D(\Gamma_n \parallel \pi) \)

\[
\Rightarrow \eta_{nc} \approx F_{em}^{-1} (1 - \beta_n; n)
\]

←-- our work
A Central Limit Theorem (CLT)

**Theorem**

A CLT holds for \( U_n = \sqrt{n} (\Gamma_n - \pi) \), that is, \( U_n \xrightarrow{d.} \mathcal{N} (0, \Lambda) \)
with \( \Lambda = \left[ \Lambda^{(i,j)} \right]_{i,j=1}^{N^2} \) given by

\[
\Lambda^{(i,j)} = \tilde{\pi}_i (I_{ij} - \tilde{\pi}_j) + \sum_{m=1}^{\infty} \left[ \tilde{\pi}_i (P_{ij}^m - \tilde{\pi}_j) + \tilde{\pi}_j (P_{ji}^m - \tilde{\pi}_i) \right].
\]

- **Taylor expansion:**
  - Letting \( h(\nu) \overset{\text{def}}{=} D(\nu \| \pi) \), \( D(\Gamma_n \| \pi) \approx \frac{1}{2n} U'_n \nabla^2 h(\pi) U_n \)

- **Approximation for \( \eta_n \):**
  - Generate samples of \( \mathcal{N} (0, \Lambda) \), hence, obtaining approximate samples of \( D(\Gamma_n \| \pi) \)
  - Compute an **Empirical Cumulative Distribution Function**, denoted \( F_{em}(\cdot; n) \), of \( D(\Gamma_n \| \pi) \)

\[
\Rightarrow \eta_n^{wc} \approx F_{em}^{-1}(1 - \beta_n; n)
\]
A Central Limit Theorem (CLT)

**Theorem**

A CLT holds for $U_n = \sqrt{n} (\Gamma_n - \pi)$, that is, $U_n \xrightarrow{d.} n \to \infty \mathcal{N} (0, \Lambda)$

with $\Lambda = \left[ \Lambda^{(i,j)} \right]_{i,j=1}^{N^2}$ given by

$$\Lambda^{(i,j)} = \tilde{\pi}_i (I_{ij} - \tilde{\pi}_j) + \sum_{m=1}^{\infty} \left[ \tilde{\pi}_i (P_{ij}^m - \tilde{\pi}_j) + \tilde{\pi}_j (P_{ji}^m - \tilde{\pi}_i) \right].$$

**Taylor expansion:**

- Letting $h(\nu) \stackrel{\text{def}}{=} D(\nu \parallel \pi)$, $D(\Gamma_n \parallel \pi) \approx \frac{1}{2n} U_n' \nabla^2 h(\pi) U_n$

**Approximation for $\eta_n$:**

- Generate samples of $\mathcal{N} (0, \Lambda)$, hence, obtaining approximate samples of $D(\Gamma_n \parallel \pi)$
- Compute an **Empirical Cumulative Distribution Function**, denoted $F_{em}(\cdot ; n)$, of $D(\Gamma_n \parallel \pi)$

$$\Rightarrow \eta_n^{wc} \approx F_{em}^{-1} (1 - \beta_n ; n) \quad \leftarrow \text{our work}$$
Numerical Comparison \((N = 4)\)

**Conclusion:** \(\eta_n^{wc}\) is typically more accurate than \(\eta_n^{sv}\)
A Demonstration of CLT

- an intuition why $\eta_n^{wc}$ is typically more accurate than $\eta_n^{sv}$
- $\sqrt{n} (\overline{X} - \mu) \xrightarrow{d} N(0, \sigma^2)$

source: https://www.youtube.com/watch?v=PujollYc1_A
ROC Analysis

\[ \beta = 0.001, N = 4, n = 50 \]

\[ \begin{array}{c}
\text{False Positive Rate} \\
0.0 & 0.2 & 0.4 & 0.6 & 0.8 & 1.0 \\
\hline
\text{True Positive Rate} \\
0.0 & 0.2 & 0.4 & 0.6 & 0.8 & 1.0 \\
\end{array} \]

Figure: ROC graphs of Hoeffding’s test with different threshold estimators.
Detecting Atypical Traffic Jams Using Waze Data

The Waze datasets are provided by the Department of Innovation and Technology (DoIT) in the City of Boston

- **Jams** $J_1$: traffic slowdown info generated by Waze based on user’s location and speed; each jam consists of a set of points
  - uuid (unique jam ID), start time, end time, speed (current average speed on jammed segments), delay (delay of jam compared to free flow speed), and length (jam length)
- **Points** $J_2$: latitudes and longitudes of the points within jams
  - jam uuid and the locations (latitudes and longitudes) of the points within the jam
- **Alerts** $J_3$: traffic incidents reported by active users
  - uuid (different from the jam uuid in $J_1$), location (latitude and longitude per report), type (accident, weather hazard, road closed, etc.), start time, and end time
Detecting Atypical Traffic Jams Using Waze Data

The Waze datasets are provided by the Department of Innovation and Technology (DoIT) in the City of Boston

- **Jams** $\mathcal{J}_1$: traffic slowdown info generated by Waze based on user’s location and speed; each jam consists of a set of points
  - **uuid** (unique jam ID), **start time**, **end time**, **speed** (current average speed on jammed segments), **delay** (delay of jam compared to free flow speed), and **length** (jam length)

- **Points** $\mathcal{J}_2$: latitudes and longitudes of the points within jams
  - jam **uuid** and the locations (latitudes and longitudes) of the points within the jam

- **Alerts** $\mathcal{J}_3$: traffic incidents reported by active users
  - **uuid** (different from the jam uuid in $\mathcal{J}_1$), location (latitude and longitude per report), **type** (accident, weather hazard, road closed, etc.), **start time**, and **end time**
Detecting Atypical Traffic Jams Using Waze Data

The Waze datasets are provided by the Department of Innovation and Technology (DoIT) in the City of Boston

- **Jams** $\mathcal{J}_1$: traffic slowdown info generated by Waze based on user’s location and speed; each jam consists of a set of points
  - **uuid** (unique jam ID), **start time**, **end time**, **speed** (current average speed on jammed segments), **delay** (delay of jam compared to free flow speed), and **length** (jam length)
- **Points** $\mathcal{J}_2$: latitudes and longitudes of the points within jams
  - jam uuid and the locations (latitudes and longitudes) of the points within the jam
- **Alerts** $\mathcal{J}_3$: traffic incidents reported by active users
  - **uuid** (different from the jam uuid in $\mathcal{J}_1$), location (latitude and longitude per report), **type** (accident, weather hazard, road closed, etc.), **start time**, and **end time**
Detecting Atypical Traffic Jams Using Waze Data

The Waze datasets are provided by the Department of Innovation and Technology (DoIT) in the City of Boston

- Jams $I_1$: traffic slowdown info generated by Waze based on user’s location and speed; each jam consists of a set of points
  - uuid (unique jam ID), start time, end time, speed (current average speed on jammed segments), delay (delay of jam compared to free flow speed), and length (jam length)
- Points $I_2$: latitudes and longitudes of the points within jams
  - jam uuid and the locations (latitudes and longitudes) of the points within the jam
- Alerts $I_3$: traffic incidents reported by active users
  - uuid (different from the jam uuid in $I_1$), location (latitude and longitude per report), type (accident, weather hazard, road closed, etc.), start time, and end time
Detecting Atypical Traffic Jams Using Waze Data

The Waze datasets are provided by the Department of Innovation and Technology (DoIT) in the City of Boston

- Jams $J_1$: traffic slowdown info generated by Waze based on user’s location and speed; each jam consists of a set of points
  - uuid (unique jam ID), start time, end time, speed (current average speed on jammed segments), delay (delay of jam compared to free flow speed), and length (jam length)

- Points $J_2$: latitudes and longitudes of the points within jams
  - jam uuid and the locations (latitudes and longitudes) of the points within the jam

- Alerts $J_3$: traffic incidents reported by active users
  - uuid (different from the jam uuid in $J_1$), location (latitude and longitude per report), type (accident, weather hazard, road closed, etc.), start time, and end time
Detecting Atypical Traffic Jams Using Waze Data

The Waze datasets are provided by the Department of Innovation and Technology (DoIT) in the City of Boston

- **Jams** $J_1$: traffic slowdown info generated by Waze based on user’s location and speed; each jam consists of a set of points
  - **uuid** (unique jam ID), **start time**, **end time**, **speed** (current average speed on jammed segments), **delay** (delay of jam compared to free flow speed), and **length** (jam length)

- **Points** $J_2$: latitudes and longitudes of the points within jams
  - jam **uuid** and the **locations** (latitudes and longitudes) of the points within the jam

- **Alerts** $J_3$: traffic incidents reported by active users
  - **uuid** (different from the jam uuid in $J_1$), **location** (latitude and longitude per report), **type** (accident, weather hazard, road closed, etc.), **start time**, and **end time**
Detecting Atypical Traffic Jams Using Waze Data (cont.)

- Time series of vectors:
  - \((i, \text{uuid}[i], \text{loc}[i], \text{speed}[i], \text{delay}[i], \text{length}[i], \text{startTime}[i])\)
  - Jam index: \(i\)

- Features:
  - Continuous: \(\text{speed}[i], \text{delay}[i], \text{length}[i]\)
  - Discrete: \(\text{loc}[i]\)
Detecting Atypical Traffic Jams Using Waze Data (cont.)

- Time series of vectors:
  - $(i, \text{uuid}[i], \text{loc}[i], \text{speed}[i], \text{delay}[i], \text{length}[i], \text{startTime}[i])$
  - jam index: $i$

- Features:
  - Continuous: $\text{speed}[i], \text{delay}[i], \text{length}[i]$
  - Discrete: $\text{loc}[i]$
Detecting Atypical Traffic Jams Using Waze Data (cont.)

- Time series of vectors:
  - \((i, \text{uuid}[i], \text{loc}[i], \text{speed}[i], \text{delay}[i], \text{length}[i], \text{startTime}[i])\)
  - Jam index: \(i\)

- Features:
  - Continuous: \(\text{speed}[i], \text{delay}[i], \text{length}[i]\)
  - Discrete: \(\text{loc}[i]\)
Time series of vectors:
- \((i, \text{uuid}[i], \text{loc}[i], \text{speed}[i], \text{delay}[i], \text{length}[i], \text{startTime}[i])\)
- jam index: \(i\)

Features:
- Continuous: \(\text{speed}[i], \text{delay}[i], \text{length}[i]\)
- Discrete: \(\text{loc}[i]\)
Detecting Atypical Traffic Jams Using Waze Data (cont.)

- **Time series of vectors:**
  - \((i, \text{uuid}[i], \text{loc}[i], \text{speed}[i], \text{delay}[i], \text{length}[i], \text{startTime}[i])\)
  - jam index: \(i\)

- **Features:**
  - Continuous: \(\text{speed}[i], \text{delay}[i], \text{length}[i]\)
  - Discrete: \(\text{loc}[i]\)
Detecting Atypical Traffic Jams Using Waze Data (cont.)

- **Time series of vectors:**
  - \((i, \text{uuid}[i], \text{loc}[i], \text{speed}[i], \text{delay}[i], \text{length}[i], \text{startTime}[i])\)
  - jam index: \(i\)

- **Features:**
  - Continuous: \(\text{speed}[i], \text{delay}[i], \text{length}[i]\)
  - Discrete: \(\text{loc}[i]\)
Detecting Atypical Traffic Jams Using Waze Data (cont.)

Reference (resp., Test) data are taken as jams reported on March 9, 2016 (resp., March 16, 2016).

- Both are Wednesdays, representing typical workdays.
- 3218 reference jams, and 3882 test jams.

Figure: Location cluster centers and detected abnormal jams for a circle area (radius = 3 kilometers) around BU.
Detecting Atypical Traffic Jams Using Waze Data (cont.)

Reference (resp., Test) data are taken as jams reported on March 9, 2016 (resp., March 16, 2016).

- Both are Wednesdays, representing typical workdays.
- 3218 reference jams, and 3882 test jams

Figure: Location cluster centers and detected abnormal jams for a circle area (radius = 3 kilometers) around BU.
Detection Results

Table: Key features of the detected abnormal jams.

<table>
<thead>
<tr>
<th>index</th>
<th>start time</th>
<th>detected time</th>
<th>latitude</th>
<th>longitude</th>
<th>delay (in seconds)</th>
<th>length (in meters)</th>
<th>alert type</th>
</tr>
</thead>
<tbody>
<tr>
<td>788</td>
<td>12:25:0.302</td>
<td>12:30:0.0</td>
<td>42.361951</td>
<td>-71.117963</td>
<td>232.0</td>
<td>3568.0</td>
<td>heavy traffic</td>
</tr>
<tr>
<td>1502</td>
<td>15:35:0.072</td>
<td>15:40:0.0</td>
<td>42.356275</td>
<td>-71.119852</td>
<td>585.0</td>
<td>844.0</td>
<td>heavy traffic</td>
</tr>
<tr>
<td>2412</td>
<td>19:25:0.365</td>
<td>19:30:0.0</td>
<td>42.342549</td>
<td>-71.085011</td>
<td>643.0</td>
<td>3568.0</td>
<td>heavy traffic</td>
</tr>
<tr>
<td>3005</td>
<td>21:25:0.238</td>
<td>21:30:0.0</td>
<td>42.349125</td>
<td>-71.10778</td>
<td>168.0</td>
<td>1962.0</td>
<td>weather hazard</td>
</tr>
<tr>
<td>3094</td>
<td>21:35:0.267</td>
<td>21:40:0.0</td>
<td>42.373336</td>
<td>-71.097731</td>
<td>509.0</td>
<td>897.0</td>
<td>road closed</td>
</tr>
<tr>
<td>3126</td>
<td>21:35:0.326</td>
<td>21:40:0.0</td>
<td>42.355048</td>
<td>-71.110335</td>
<td>528.0</td>
<td>1293.0</td>
<td>heavy traffic</td>
</tr>
</tbody>
</table>

- Comparing the locations and time stamps, map the jams in the final anomaly list to the alert data $J_3$, and find that one of them was reported as “road closed,” another of them as “weather hazard,” and all the others as “heavy traffic.”

- All happened during non-peak hours and would be detected as atypical within 5 minutes from their start time.
Detection Results

Table: Key features of the detected abnormal jams.

<table>
<thead>
<tr>
<th>index</th>
<th>start time</th>
<th>detected time</th>
<th>latitude</th>
<th>longitude</th>
<th>delay (in seconds)</th>
<th>length (in meters)</th>
<th>alert type</th>
</tr>
</thead>
<tbody>
<tr>
<td>788</td>
<td>12:25:0.302</td>
<td>12:30:0.0</td>
<td>42.361951</td>
<td>-71.117963</td>
<td>232.0</td>
<td>3568.0</td>
<td>heavy traffic</td>
</tr>
<tr>
<td>1502</td>
<td>15:35:0.072</td>
<td>15:40:0.0</td>
<td>42.356275</td>
<td>-71.119852</td>
<td>585.0</td>
<td>844.0</td>
<td>heavy traffic</td>
</tr>
<tr>
<td>2412</td>
<td>19:25:0.365</td>
<td>19:30:0.0</td>
<td>42.342549</td>
<td>-71.085011</td>
<td>643.0</td>
<td>3568.0</td>
<td>heavy traffic</td>
</tr>
<tr>
<td>3005</td>
<td>21:25:0.238</td>
<td>21:30:0.0</td>
<td>42.349125</td>
<td>-71.10778</td>
<td>168.0</td>
<td>1962.0</td>
<td>weather hazard</td>
</tr>
<tr>
<td>3094</td>
<td>21:35:0.267</td>
<td>21:40:0.0</td>
<td>42.373336</td>
<td>-71.097731</td>
<td>509.0</td>
<td>897.0</td>
<td>road closed</td>
</tr>
<tr>
<td>3126</td>
<td>21:35:0.326</td>
<td>21:40:0.0</td>
<td>42.355048</td>
<td>-71.110335</td>
<td>528.0</td>
<td>1293.0</td>
<td>heavy traffic</td>
</tr>
</tbody>
</table>

- Comparing the locations and time stamps, map the jams in the final anomaly list to the alert data $J_3$, and find that one of them was reported as “road closed,” another of them as “weather hazard,” and all the others as “heavy traffic.”

- All happened during non-peak hours and would be detected as atypical within 5 minutes from their start time.
Now, let’s turn to the slower time-scale results.

Outline:

- Model and methods
- Numerical results
Single-Class Transportation Network Model

- **Directed graph:** \( G = (V, A) \); \( V \) denotes the set of nodes and \( A \) the set of links
- **Node-link incidence matrix:** \( N \in \{0, 1, -1\}^{|V| \times |A|} \)
- **Demand:** Origin-Destination (OD) pairs \( w = (w_s, w_t) \in W \) with demand traffic \( d^w \) flowing \( w_s \rightarrow w_t \); let \( d^w = (0, \ldots, -d^w, \ldots, d^w, \ldots, 0) \)
- **Feasible flows:**
  \[ \mathcal{F} = \{ x \mid \exists x^w \text{s.t. } x = \sum_{w \in W} x^w, \text{ } N x^w = d^w, \forall w \in W \} \]
- **Indicator vector:** \( e_a \) denotes the vector with an entry equal to 1 corresponding to link \( a \) and all the other entries equal to 0.
Single-Class Transportation Network Model

- **Directed graph**: \( G = (\mathcal{V}, \mathcal{A}) \); \( \mathcal{V} \) denotes the set of nodes and \( \mathcal{A} \) the set of links

- **Node-link incidence matrix**: \( \mathbf{N} \in \{0, 1, -1\}^{|\mathcal{V}| \times |\mathcal{A}|} \)

- **Demand**: Origin-Destination (OD) pairs \( \mathbf{w} = (w_s, w_t) \in \mathcal{W} \) with demand traffic \( d^\mathbf{w} \) flowing \( w_s \rightarrow w_t \); let \( d^\mathbf{w} = (0, \ldots, -d^\mathbf{w}, \ldots, d^\mathbf{w}, \ldots, 0) \)

- **Feasible flows**: \( \mathcal{F} = \{ \mathbf{x} \mid \exists \mathbf{x}^\mathbf{w} \text{s.t. } \mathbf{x} = \sum_{\mathbf{w} \in \mathcal{W}} \mathbf{x}^\mathbf{w}, \mathbf{N}\mathbf{x}^\mathbf{w} = d^\mathbf{w}, \forall \mathbf{w} \in \mathcal{W} \} \)

- **Indicator vector**: \( \mathbf{e}_a \) denotes the vector with an entry equal to 1 corresponding to link \( a \) and all the other entries equal to 0
Single-Class Transportation Network Model

- **Directed graph**: $G = (V, A)$; $V$ denotes the set of nodes and $A$ the set of links
- **Node-link incidence matrix**: $N \in \{0, 1, -1\}^{|V| \times |A|}$
- **Demand**: Origin-Destination (OD) pairs $w = (w_s, w_t) \in W$ with demand traffic $d^w$ flowing $w_s \to w_t$; let $d^w = (0, \ldots, -d^w, \ldots, d^w, \ldots, 0)$
- **Feasible flows**: $\mathcal{F} = \{x | \exists x^w \text{s.t. } x = \sum_{w \in W} x^w, \ N x^w = d^w, \forall w \in W\}$
- **Indicator vector**: $e_a$ denotes the vector with an entry equal to 1 corresponding to link $a$ and all the other entries equal to 0
Single-Class Transportation Network Model

- **Directed graph**: $G = (V, A)$; $V$ denotes the set of nodes and $A$ the set of links
- **Node-link incidence matrix**: $N \in \{0, 1, -1\}^{|V| \times |A|}$
- **Demand**: Origin-Destination (OD) pairs $w = (w_s, w_t) \in W$ with demand traffic $d^w$ flowing $w_s \to w_t$; let $d^w = (0, \ldots, -d^w, \ldots, d^w, \ldots, 0)$
- **Feasible flows**: 
  $\mathcal{F} = \{x | \exists x^w \text{s.t. } x = \sum_{w \in W} x^w, \quad Nx^w = d^w, \quad \forall w \in W\}$
- **Indicator vector**: $e_a$ denotes the vector with an entry equal to 1 corresponding to link $a$ and all the other entries equal to 0

22/46 Jing Zhang, Boston University Christmas Talk at Zhejiang University: Slower
Single-Class Transportation Network Model

- **Directed graph**: $\mathcal{G} = (\mathcal{V}, \mathcal{A})$; $\mathcal{V}$ denotes the set of nodes and $\mathcal{A}$ the set of links.

- **Node-link incidence matrix**: $\mathbf{N} \in \{0, 1, -1\}^{\vert \mathcal{V} \vert \times \vert \mathcal{A} \vert}$

- **Demand**: Origin-Destination (OD) pairs $\mathbf{w} = (w_s, w_t) \in \mathcal{W}$ with demand traffic $d^w$ flowing $w_s \rightarrow w_t$; let $d^w = (0, \ldots, -d^w, \ldots, d^w, \ldots, 0)$

- **Feasible flows**:
  $$\mathcal{F} = \{ \mathbf{x} \mid \exists \mathbf{x}^w \text{s.t. } \mathbf{x} = \sum_{\mathbf{w} \in \mathcal{W}} \mathbf{x}^w, \mathbf{N} \mathbf{x}^w = \mathbf{d}^w, \forall \mathbf{w} \in \mathcal{W} \}$$

- **Indicator vector**: $\mathbf{e}_a$ denotes the vector with an entry equal to 1 corresponding to link $a$ and all the other entries equal to 0.
Single-Class Transportation Network Model (cont.)

- **Dynamics:** Vehicles pick links (and routes) using a travel cost (time) function $t_a(x)$ for each link $a \in \mathcal{A}$
  - Assume $t_a(\cdot)$ takes the form:
    $$t_a(x_a) = t_0 a g \left( \frac{x_a}{m_a} \right),$$
    where $t_0 a$ is the free-flow travel time, $x_a$ is the flow on link $a$, $m_a$ is the flow capacity of link $a$, and $g(\cdot)$ is strictly increasing
  - Link flows converge to a user-equilibrium (Wardrop (1952))
Definition of Price of Anarchy (PoA)

Let $x_a^{\text{user}}$ (resp., $x_a^{\text{social}}$) denote the user-equilibrium (resp., socially-optimal) flow on link $a$

- Define the PoA as the ratio of the total travel cost under a user-centric policy versus a system-centric policy:

$$
\text{PoA} = \frac{\sum_{a \in \mathcal{A}} x_a^{\text{user}} t_a(x_a^{\text{user}})}{\sum_{a \in \mathcal{A}} x_a^{\text{social}} t_a(x_a^{\text{social}})} \geq 1
$$

- PoA quantifies the efficiency loss of selfish routing versus socially-optimal routing
Definition of Price of Anarchy (PoA)

- Let $x_{a}^{\text{user}}$ (resp., $x_{a}^{\text{social}}$) denote the user-equilibrium (resp., socially-optimal) flow on link $a$

- Define the PoA as the ratio of the total travel cost under a user-centric policy versus a system-centric policy:

$$\text{PoA} = \frac{\sum_{a \in \mathcal{A}} x_{a}^{\text{user}} t_{a}(x_{a}^{\text{user}})}{\sum_{a \in \mathcal{A}} x_{a}^{\text{social}} t_{a}(x_{a}^{\text{social}})} \geq 1$$

- PoA quantifies the efficiency loss of selfish routing versus socially-optimal routing
Definition of Price of Anarchy (PoA)

- Let $x_a^{\text{user}}$ (resp., $x_a^{\text{social}}$) denote the user-equilibrium (resp., socially-optimal) flow on link $a$.
- Define the PoA as the ratio of the total travel cost under a user-centric policy versus a system-centric policy:

$$\text{PoA} = \frac{\sum_{a \in A} x_a^{\text{user}} t_a(x_a^{\text{user}})}{\sum_{a \in A} x_a^{\text{social}} t_a(x_a^{\text{social}})} \geq 1$$

- PoA quantifies the efficiency loss of selfish routing versus socially-optimal routing.
Road Map of Eastern Massachusetts

Road map of Eastern Massachusetts (EMA), over which we have access to a vast amount of actual traffic data

Figure: All available road segments in the road map of EMA
Datasets

What we already have:

- speed data & capacity data
  - spatial average speeds for more than 13,000 road segments in EMA; covers every minute of the year 2012
  - flow capacity (# of vehicles per hour) for more than 100,000 road segments in EMA
Quantities Needed to Evaluate PoA

What we don’t have but can estimate from data:

- **user flows**: $x_{a}^{\text{user}}$
- travel cost functions $t_{a}(\cdot)$
  - Requires user flows and an OD demand matrix
- **social flows**: $x_{a}^{\text{social}}$
  - Requires travel cost functions and an OD demand matrix
What we don’t have but can estimate from data:

- **user flows**: $x^\text{user}_a$

- **travel cost functions** $t_a(\cdot)$
  - Requires user flows and an OD demand matrix

- **social flows**: $x^{\text{social}}_a$

- Requires travel cost functions and an OD demand matrix
Quantities Needed to Evaluate PoA

What we don’t have but can estimate from data:

- **user flows**: $x^{\text{user}}_a$
- **travel cost functions** $t_a(\cdot)$
  - Requires user flows and an OD demand matrix
- **social flows**: $x^{\text{social}}_a$
  - Requires travel cost functions and an OD demand matrix
Quantities Needed to Evaluate PoA

What we don’t have but can estimate from data:

- **user flows**: $x^\text{user}_a$
  - Requires user flows and an OD demand matrix
- **travel cost functions** $t_a(\cdot)$
  - Requires user flows and an OD demand matrix
- **social flows**: $x^\text{social}_a$
  - Requires travel cost functions and an OD demand matrix
Quantities Needed to Evaluate PoA

What we don’t have but can estimate from data:

- **user flows**: $x^\text{user}_a$
- **travel cost functions** $t_a(\cdot)$
  - Requires user flows and an OD demand matrix
- **social flows**: $x^\text{social}_a$
  - Requires travel cost functions and an OD demand matrix
Inferring User Flows — Converting Speeds to Flows

- Greenshield’s model (Mathew (2014)):

\[
x_a = 4 m_a \left[ \frac{v_a}{v_0^a} - \left( \frac{v_a}{v_0^a} \right)^2 \right],
\]

where \( m_a \) is the flow capacity, \( v_a \) the current average speed, and \( v_0^a \) the free-flow speed.

- Assume these inferred flow observations form an equilibrium (Wardrop (1952)) under a “user-centric” routing policy; \( x_a^{\text{user}} \)
Estimating OD Demand Matrix

- Define $A$ as link-route incidence matrix, $P$ route choice probability matrix, $S$ sample covariance matrix, and $\lambda$ vectorized OD demand matrix.
- Let $R_i$ be the set of all feasible routes connecting OD pair $i$.
- Estimate $\lambda$ by using a Generalized Least Squares (GLS) method (Hazelton (2000)):

$$\min_{P \geq 0, \lambda \geq 0} \sum_{k=1}^{K} \left( x^{(k)} - AP' \lambda \right)' S^{-1} \left( x^{(k)} - AP' \lambda \right)$$

s.t. $p_{ir} = 0 \quad \forall (i, r) \in \{ (i, r) : r \notin R_i \}$,

$P1 = 1$. 

- Could be dependent on time-intervals (AM/MD/PM/NT) of day and days of week (weekday/weekend).
- Estimate $\lambda$ using data with different time stamps accordingly.
Estimating OD Demand Matrix

- Define $A$ as link-route incidence matrix, $P$ route choice probability matrix, $S$ sample covariance matrix, and $\lambda$ vectorized OD demand matrix.
- Let $\mathcal{R}_i$ be the set of all feasible routes connecting OD pair $i$.
- Estimate $\lambda$ by using a Generalized Least Squares (GLS) method (Hazelton (2000)):

$$\min_{P \geq 0, \lambda \geq 0} \sum_{k=1}^{K} \left( x^{(k)} - AP'\lambda \right)' S^{-1} \left( x^{(k)} - AP'\lambda \right)$$

s.t. $p_{ir} = 0 \quad \forall (i, r) \in \{(i, r) : r \notin \mathcal{R}_i\}$,

$P1 = 1$.

- Could be dependent on time-intervals (AM/MD/PM/NT) of day and days of week (weekday/weekend).
- Estimate $\lambda$ using data with different time stamps accordingly.
Estimating OD Demand Matrix

- Define $A$ as link-route incidence matrix, $P$ route choice probability matrix, $S$ sample covariance matrix, and $\lambda$ vectorized OD demand matrix.
- Let $R_i$ be the set of all feasible routes connecting OD pair $i$.
- Estimate $\lambda$ by using a Generalized Least Squares (GLS) method (Hazelton (2000)):

$$
\min_{P \geq 0, \lambda \geq 0} \sum_{k=1}^{K} \left( x^{(k)} - AP'\lambda \right)' S^{-1} \left( x^{(k)} - AP'\lambda \right)
$$

subject to:

$$
p_{ir} = 0 \quad \forall (i, r) \in \{(i, r) : r \notin R_i\},
$$

$$
P1 = 1.
$$

- Could be dependent on time-intervals (AM/MD/PM/NT) of day and days of week (weekday/weekend).
- Estimate $\lambda$ using data with different time stamps accordingly.
Estimating OD Demand Matrix

- Define $A$ as link-route incidence matrix, $P$ route choice probability matrix, $S$ sample covariance matrix, and $\lambda$ vectorized OD demand matrix.
- Let $R_i$ be the set of all feasible routes connecting OD pair $i$.
- Estimate $\lambda$ by using a Generalized Least Squares (GLS) method (Hazelton (2000)):

$$
\begin{align*}
\min_{P \geq 0, \lambda \geq 0} & \quad \sum_{k=1}^{K} \left( x^{(k)} - AP'\lambda \right)' S^{-1} \left( x^{(k)} - AP'\lambda \right) \\
\text{s.t.} & \quad p_{ir} = 0 \; \forall (i, r) \in \{(i, r) : r \notin R_i\}, \\
& \quad P1 = 1.
\end{align*}
$$

- Could be dependent on time-intervals (AM/MD/PM/NT) of day and days of week (weekday/weekend).
- Estimate $\lambda$ using data with different time stamps accordingly.
Estimating OD Demand Matrix

- Define $A$ as link-route incidence matrix, $P$ route choice probability matrix, $S$ sample covariance matrix, and $\lambda$ vectorized OD demand matrix.
- Let $R_i$ be the set of all feasible routes connecting OD pair $i$.
- Estimate $\lambda$ by using a Generalized Least Squares (GLS) method (Hazelton (2000)):

$$
\min_{P \geq 0, \lambda \geq 0} \sum_{k=1}^{K} \left( x^{(k)} - AP'\lambda \right)' S^{-1} \left( x^{(k)} - AP'\lambda \right)
$$

s.t. $p_{ir} = 0 \quad \forall (i, r) \in \{(i, r) : r \notin R_i\}$,

$P1 = 1$.

- Could be dependent on time-intervals (AM/MD/PM/NT) of day and days of week (weekday/weekend).
- Estimate $\lambda$ using data with different time stamps accordingly.
Estimating Cost Functions

- **Recall** \( t_a(x_a) = t^0_a g \left( \frac{x_a}{m_a} \right) \)

- Seek to find a \( g(\cdot) \) under which the observed user flows \( x^\text{user}_a \) and the estimated OD demand \( \lambda \) are as consistent as possible (good data reconciling)
  - Could be dependent on time-intervals (AM/MD/PM/NT) of day and days of week (weekday/weekend)
  - Estimate \( g(\cdot) \) using data with different time stamps accordingly

- Seek to find \( g(\cdot) \) having strong predictive power (good generalization properties)

- Achieve these by solving an inverse optimization problem, which is reduced to a Quadratic Programming (QP) problem
Estimating Cost Functions

- Recall $t_a(x_a) = t^0_ag\left(\frac{x_a}{m_a}\right)$

- Seek to find a $g(\cdot)$ under which the observed user flows $x^{user}_a$ and the estimated OD demand $\lambda$ are as consistent as possible (good data reconciling)
  - Could be dependent on time-intervals (AM/MD/PM/NT) of day and days of week (weekday/weekend)
  - Estimate $g(\cdot)$ using data with different time stamps accordingly

- Seek to find $g(\cdot)$ having strong predictive power (good generalization properties)

- Achieve these by solving an inverse optimization problem, which is reduced to a Quadratic Programming (QP) problem
Estimating Cost Functions

- Recall $t_a(x_a) = t_0^a g \left( \frac{x_a}{m_a} \right)$
- Seek to find a $g(\cdot)$ under which the observed user flows $x_a^{\text{user}}$ and the estimated OD demand $\lambda$ are as consistent as possible (good data reconciling)
  - Could be dependent on time-intervals (AM/MD/PM/NT) of day and days of week (weekday/weekend)
  - Estimate $g(\cdot)$ using data with different time stamps accordingly
- Seek to find $g(\cdot)$ having strong predictive power (good generalization properties)
- Achieve these by solving an inverse optimization problem, which is reduced to a Quadratic Programming (QP) problem
Estimating Cost Functions

- Recall $t_a(x_a) = t_a^0 g\left(\frac{x_a}{m_a}\right)$
- Seek to find a $g(\cdot)$ under which the observed user flows $x_{\text{user}}^a$ and the estimated OD demand $\lambda$ are as consistent as possible (good data reconciling)
  - Could be dependent on time-intervals (AM/MD/PM/NT) of day and days of week (weekday/weekend)
  - Estimate $g(\cdot)$ using data with different time stamps accordingly
- Seek to find $g(\cdot)$ having strong predictive power (good generalization properties)
- Achieve these by solving an inverse optimization problem, which is reduced to a Quadratic Programming (QP) problem
Estimating Cost Functions

- Recall \( t_a (x_a) = t^0_a g \left( \frac{x_a}{m_a} \right) \)
- Seek to find a \( g(\cdot) \) under which the observed user flows \( x^{\text{user}}_a \) and the estimated OD demand \( \lambda \) are as consistent as possible (good data reconciling)
  - Could be dependent on time-intervals (AM/MD/PM/NT) of day and days of week (weekday/weekend)
  - Estimate \( g(\cdot) \) using data with different time stamps accordingly
- Seek to find \( g(\cdot) \) having strong predictive power (good generalization properties)
  - Achieve these by solving an inverse optimization problem, which is reduced to a Quadratic Programming (QP) problem
Motivation

Fast

Slower

Discussions

Ack

Model & Methods

Numerical

Estimating Cost Functions

- Recall $t_a(x_a) = t_0^a g\left(\frac{x_a}{m_a}\right)$

- Seek to find a $g(\cdot)$ under which the observed user flows $x_{a_{user}}$ and the estimated OD demand $\lambda$ are as consistent as possible (good data reconciling)
  - Could be dependent on time-intervals (AM/MD/PM/NT) of day and days of week (weekday/weekend)
  - Estimate $g(\cdot)$ using data with different time stamps accordingly

- Seek to find $g(\cdot)$ having strong predictive power (good generalization properties)

- Achieve these by solving an inverse optimization problem, which is reduced to a Quadratic Programming (QP) problem
Forward Problem

Forward optimization problem (given cost functions and OD demands, find user flows):

- Determine the user flows $x_{\text{user}}^a$ by solving the following Non-Linear Programming (NLP) problem (Dafermos (1969)):

$$\min_{x \in \mathcal{F}} \sum_{a \in \mathcal{A}} \int_0^{x_a} t_a(s) ds$$
Sensitivity Analysis

Write \( t_0^0 \equiv (t_a^0; \ a \in \mathcal{A}) \), \( m \equiv (m_a; \ a \in \mathcal{A}) \), and

\[
V (t^0, m) \equiv \min_{x \in \mathcal{F}} \sum_{a \in \mathcal{A}} \int_{0}^{x_a} t^0_a g \left( \frac{s}{m_a} \right) ds.
\]

Then

\[
\frac{\partial V (t^0, m)}{\partial t_a^0} = \int_{0}^{x_a^*} g \left( \frac{s}{m_a} \right) ds,
\]

\[
\frac{\partial V (t^0, m)}{\partial m_a} = \int_{0}^{x_a^*} t^0_a \dot{g} \left( \frac{s}{m_a} \right) \left( -\frac{s}{m_a^2} \right) ds,
\]

where \( \dot{g}(\cdot) \) denotes the derivative of \( g(\cdot) \).
Estimating Cost Functions (cont.)

Given user flows \( \{(x_a^k; a \in \mathcal{A}_k); k = 1, \ldots, K\} \) and OD demands \( d^w \). Let \( \mathcal{H} \) be a Reproducing Kernel Hilbert Space (RKHS). Solve the following inverse optimization problem (Bertsimas et al. (2014)):

\[
\begin{align*}
\min_{g, y, \epsilon} \quad & \|\epsilon\| + \gamma \|g\|_H^2 \\
\text{s.t.} \quad & e'_a N'_k y^w \leq t_0^a g \left( \frac{x_a}{m_a} \right), \quad \forall w \in \mathcal{W}_k, a \in \mathcal{A}_k, k = 1, \ldots, K, \\
& \sum_{a \in \mathcal{A}_k} t_0^a x_a g \left( \frac{x_a}{m_a} \right) - \sum_{w \in \mathcal{W}_k} (d^w)' y^w \leq \epsilon_k, \quad \forall k = 1, \ldots, K, \\
& \epsilon \geq 0, \quad g \in \mathcal{H}, \\
& g \left( \frac{x_a}{m_a} \right) \leq g \left( \frac{x_{\tilde{a}}}{m_{\tilde{a}}} \right), \quad \forall a, \tilde{a} \in \bigcup_{k=1}^{K} \mathcal{A}_k \quad \text{s.t.} \quad \frac{x_a}{m_a} \leq \frac{x_{\tilde{a}}}{m_{\tilde{a}}}, \\
& g(0) = 1.
\end{align*}
\]
Estimating Cost Functions (cont.)

Take polynomial kernel $\phi(x, y) = (c + xy)^n$. Reformulate the inverse optimization problem as the following QP:

$$\min_{\beta, y, \epsilon} \|\epsilon\| + \gamma \sum_{i=0}^{n} \frac{\beta_i^2}{\binom{n}{i} c^{n-i}}$$

s.t. $e'_a N'_k y^w \leq t_a \sum_{i=0}^{n} \beta_i \left( \frac{x_a}{m_a} \right)^i$, $\forall w \in \mathcal{W}_k$, $a \in \mathcal{A}_k$, $k = 1, \ldots, K$,

$$\sum_{a \in \mathcal{A}_k} t_a x_a \sum_{i=0}^{n} \beta_i \left( \frac{x_a}{m_a} \right)^i - \sum_{w \in \mathcal{W}_k} (d^w)' y^w \leq \epsilon_k, \forall k = 1, \ldots, K,$$

$\epsilon_k \geq 0, \forall k = 1, \ldots, K,$

$$\sum_{i=0}^{n} \beta_i \left( \frac{x_a}{m_a} \right)^i \leq \sum_{i=0}^{n} \beta_i \left( \frac{x_{\tilde{a}}}{m_{\tilde{a}}} \right)^i, \forall a, \tilde{a} \in \bigcup_{k=1}^{K} \mathcal{A}_k \text{ s.t. } \frac{x_a}{m_a} \leq \frac{x_{\tilde{a}}}{m_{\tilde{a}}},$$

$\beta_0 = 1.$
Estimating Cost Functions (cont.)

Solving the QP gives an estimator $\hat{g}(\cdot)$ of $g(\cdot)$:

$$\hat{g}(x) = \sum_{i=0}^{n} \beta_i^* x_i = 1 + \sum_{i=1}^{n} \beta_i^* x^i.$$
Finding Social Flows

PoA = \frac{\sum_{a \in A} x^\text{user}_a t_a(x^\text{user}_a)}{\sum_{a \in A} x^\text{social}_a t_a(x^\text{social}_a)} \geq 1

Now ready to calculate the social flows

Find the social flows $x^\text{social}_a$ by solving the following NLP problem (Patriksson (1994)):

(socialOpt) \min_{x \in F} \sum_{a \in A} x_a t_a(x_a)$
A Sub-Map of EMA

Figure: (a) An interstate highway sub-network; (b) The topology of the sub-network.

- 8 nodes
- 24 links
- $8 \times (8-1) = 56$ OD pairs
Results for Cost Function Estimation

**Figure:** Estimated $g(\cdot)$ for different time periods (Apr. 2012).
Results for PoA

Average PoA \approx 1.5, meaning we can improve the road network by about 50\%; some PoA > 2, meaning we can gain more than 100\% improvement!
Social Flow vs. User Flow
Results for Sensitivity Analysis

\( \frac{\partial V(t_0^0, m)}{\partial t_0^a} \): sensitivity of the optimal objective function value of \((\text{userOpt})\) with respect to the free-flow travel time, for link \(a\)

\( \frac{\partial V(t_0^0, m)}{\partial m_a} \): sensitivity of the optimal objective function value of \((\text{userOpt})\) with respect to the flow capacity, for link \(a\)

(PM peak period on Jan. 10, 2012)

<table>
<thead>
<tr>
<th>( \frac{\partial V(t^0, m)}{\partial t_0^a} )</th>
<th>( \frac{\partial V(t^0, m)}{\partial m_a} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0, 0.226)</td>
<td>(0, -0.009)</td>
</tr>
<tr>
<td>(4, 0.54)</td>
<td>(4, -0.036)</td>
</tr>
<tr>
<td>(8, 0.924)</td>
<td>(8, -0.496)</td>
</tr>
<tr>
<td>(12, 0.389)</td>
<td>(12, -0.048)</td>
</tr>
<tr>
<td>(16, 0.178)</td>
<td>(16, -0.004)</td>
</tr>
<tr>
<td>(20, 0.892)</td>
<td>(20, -0.888)</td>
</tr>
<tr>
<td>(1, 0.177)</td>
<td>(1, -0.005)</td>
</tr>
<tr>
<td>(5, 0.198)</td>
<td>(5, -0.002)</td>
</tr>
<tr>
<td>(9, 0.091)</td>
<td>(9, -0.009)</td>
</tr>
<tr>
<td>(10, 0.951)</td>
<td>(10, -0.605)</td>
</tr>
<tr>
<td>(14, 0.539)</td>
<td>(14, -0.007)</td>
</tr>
<tr>
<td>(18, 0.341)</td>
<td>(18, -0.002)</td>
</tr>
<tr>
<td>(22, 0.234)</td>
<td>(22, -0.006)</td>
</tr>
<tr>
<td>(2, 0.26)</td>
<td>(2, -0.011)</td>
</tr>
<tr>
<td>(6, 0.199)</td>
<td>(6, -0.009)</td>
</tr>
<tr>
<td>(10, -0.017)</td>
<td></td>
</tr>
<tr>
<td>(3, 0.208)</td>
<td>(3, -0.005)</td>
</tr>
<tr>
<td>(7, 0.242)</td>
<td>(7, 0.017)</td>
</tr>
<tr>
<td>(11, -0.009)</td>
<td>(11, 0.094)</td>
</tr>
<tr>
<td>(15, 0.42)</td>
<td></td>
</tr>
<tr>
<td>(19, 1.0)</td>
<td>(15, -0.103)</td>
</tr>
<tr>
<td>(23, -1.0)</td>
<td></td>
</tr>
</tbody>
</table>

Red (resp., blue) numbers: the indices of the 5 links which have biggest values (resp., absolute values) of \( \frac{\partial V(t_0^0, m)}{\partial t_0^a} \) (resp., \( \frac{\partial V(t_0^0, m)}{\partial m_a} \)
Ongoing and Future Work

- In the fast time-scale:
  - Consider a robust version Hoeffding test
  - Conduct more extensive numerical experiments
Ongoing and Future Work (cont.)

- In the slower time-scale:
  - Seek to reduce PoA especially when PoA \( \gg 1 \), e.g.,
    - by taking advantage of the emergence of the Connected Automated Vehicles (CAVs)
    - by devising "smarter" (giving preference to the system-centric routing policy) algorithms for GPS navigation
  - Consider a larger network
  - Extend the single-class travel latency cost function estimation formulation to the multi-class case, where the coupling of different types of vehicle flows will be tackled
  - Analyze parameter sensitivities for the multi-class case
  - Consider a joint problem, i.e., jointly estimating the per-road travel latency cost functions and adjusting the OD flow demand matrices
Ongoing and Future Work (cont.)

- **In the slower time-scale:**
  - Seek to reduce PoA especially when $\text{PoA} \gg 1$, e.g.,
    - by taking advantage of the emergence of the Connected Automated Vehicles (CAVs)
    - by devising “smarter” (giving preference to the system-centric routing policy) algorithms for GPS navigation
  - Consider a larger network
  - Extend the single-class travel latency cost function estimation formulation to the multi-class case, where the coupling of different types of vehicle flows will be tackled
  - Analyze parameter sensitivities for the multi-class case
  - Consider a joint problem, i.e., jointly estimating the per-road travel latency cost functions and adjusting the OD flow demand matrices
Ongoing and Future Work (cont.)

- In the slower time-scale:
  - Seek to reduce PoA especially when \( \text{PoA} \gg 1 \), e.g.,
    - by taking advantage of the emergence of the Connected Automated Vehicles (CAVs)
    - by devising “smarter” (giving preference to the system-centric routing policy) algorithms for GPS navigation
  - Consider a larger network
  - Extend the single-class travel latency cost function estimation formulation to the multi-class case, where the coupling of different types of vehicle flows will be tackled
  - Analyze parameter sensitivities for the multi-class case
  - Consider a joint problem, i.e., jointly estimating the per-road travel latency cost functions and adjusting the OD flow demand matrices
Ongoing and Future Work (cont.)

- **In the slower time-scale:**
  - Seek to reduce PoA especially when PoA \( \gg 1 \), e.g.,
    - by taking advantage of the emergence of the Connected Automated Vehicles (CAVs)
    - by devising “smarter” (giving preference to the system-centric routing policy) algorithms for GPS navigation
  - Consider a larger network
  - Extend the single-class travel latency cost function estimation formulation to the multi-class case, where the coupling of different types of vehicle flows will be tackled
  - Analyze parameter sensitivities for the multi-class case
  - Consider a joint problem, i.e., jointly estimating the per-road travel latency cost functions and adjusting the OD flow demand matrices
Ongoing and Future Work (cont.)

- **In the slower time-scale:**
  - Seek to reduce PoA especially when $\text{PoA} \gg 1$, e.g.,
    - by taking advantage of the emergence of the Connected Automated Vehicles (CAVs)
    - by devising “smarter” (giving preference to the system-centric routing policy) algorithms for GPS navigation
  - Consider a larger network
    - Extend the single-class travel latency cost function estimation formulation to the multi-class case, where the coupling of different types of vehicle flows will be tackled
    - Analyze parameter sensitivities for the multi-class case
    - Consider a joint problem, i.e., jointly estimating the per-road travel latency cost functions and adjusting the OD flow demand matrices
Ongoing and Future Work (cont.)

- In the slower time-scale:
  - Seek to reduce PoA especially when $\text{PoA} \gg 1$, e.g.,
    - by taking advantage of the emergence of the Connected Automated Vehicles (CAVs)
    - by devising “smarter” (giving preference to the system-centric routing policy) algorithms for GPS navigation
  - Consider a larger network
  - Extend the single-class travel latency cost function estimation formulation to the multi-class case, where the coupling of different types of vehicle flows will be tackled
    - Analyze parameter sensitivities for the multi-class case
    - Consider a joint problem, i.e., jointly estimating the per-road travel latency cost functions and adjusting the OD flow demand matrices
In the slower time-scale:

- Seek to reduce PoA especially when $\text{PoA} \gg 1$, e.g.,
  - by taking advantage of the emergence of the Connected Automated Vehicles (CAVs)
  - by devising “smarter” (giving preference to the system-centric routing policy) algorithms for GPS navigation
- Consider a larger network
- Extend the single-class travel latency cost function estimation formulation to the multi-class case, where the coupling of different types of vehicle flows will be tackled
- Analyze parameter sensitivities for the multi-class case
- Consider a joint problem, i.e., jointly estimating the per-road travel latency cost functions and adjusting the OD flow demand matrices
Ongoing and Future Work (cont.)

- In the slower time-scale:
  - Seek to reduce PoA especially when $\text{PoA} \gg 1$, e.g.,
    - by taking advantage of the emergence of the Connected Automated Vehicles (CAVs)
    - by devising “smarter” (giving preference to the system-centric routing policy) algorithms for GPS navigation
  - Consider a larger network
  - Extend the single-class travel latency cost function estimation formulation to the multi-class case, where the coupling of different types of vehicle flows will be tackled
  - Analyze parameter sensitivities for the multi-class case
  - Consider a joint problem, i.e., jointly estimating the per-road travel latency cost functions and adjusting the OD flow demand matrices
Publications


Acknowledgement

- Advisor: Ioannis Ch. Paschalidis
- Collaborators: Christos G. Cassandras, Sepideh Pourazarm
- Labmates: Jing Wang, Wuyang Dai, Qi Zhao, Theodora Brisimi, and more
Thank You!

Questions: jzh@bu.edu

Further References:
http://people.bu.edu/jzh/
https://github.com/jingzbu