ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/309573161

Data-driven Estimation of Origin-Destination Demand and User Cost Functions for the Optimization of...

Conference Paper · July 2017

CITATION
1

READS
97

4 authors:

Jing Zhang
Boston University
9 PUBLICATIONS **6** CITATIONS

SEE PROFILE

S. Pourazarm
Boston University
16 PUBLICATIONS **42** CITATIONS

SEE PROFILE

C.G. Cassandras
Boston University
408 PUBLICATIONS **8,644** CITATIONS

SEE PROFILE

Ioannis Ch. Paschalidis
Boston University
180 PUBLICATIONS **2,149** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project
The Price of Anarchy in Transportation Networks: Data-Driven Evaluation and Reduction Strategies
View project

Project
Detecting Data Exfiltration from a Networked Infrastructure View project

All content following this page was uploaded by Jing Zhang on 02 November 2016.

The user has requested enhancement of the downloaded file.

arXiv:1610.09580v1 [cs.SY] 29 Oct 2016

Data-driven Estimation of Origin-Destination Demand and User Cost Functions for the Optimization of Transportation Networks ⋆

Jing Zhang * **Sepideh Pourazarm** * **Christos G. Cassandras** ** **Ioannis Ch. Paschalidis** **

* *Division of Systems Engineering, Boston University (e-mail: {jzh, sepid}@bu.edu)*
** *Department of Electrical and Computer Engineering and Division of Systems Engineering, Boston University (e-mail: {cgc, yannisp}@bu.edu)*

Abstract:
In earlier work (Zhang et al., 2016) we used actual traffic data from the Eastern Massachusetts transportation network in the form of spatial average speeds and road segment flow capacities in order to estimate Origin-Destination (OD) flow demand matrices for the network. Based on a Traffic Assignment Problem (TAP) formulation (termed "forward problem"), in this paper we use a scheme similar to our earlier work to estimate initial OD demand matrices and then propose a new inverse problem formulation in order to estimate user cost functions. This new formulation allows us to efficiently overcome numerical difficulties that limited our prior work to relatively small subnetworks and, assuming the travel latency cost functions are available, to adjust the values of the OD demands accordingly so that the flow observations are as close as possible to the solutions of the forward problem. We also derive sensitivity analysis results for the total user latency cost with respect to important parameters such as road capacities and minimum travel times. Finally, using the same actual traffic data from the Eastern Massachusetts transportation network, we quantify the Price of Anarchy (POA) for a much larger network than that in Zhang et al. (2016).

Keywords: optimization, transportation networks, origin-destination demand estimation, travel latency cost function, price of anarchy, sensitivity analysis

1. INTRODUCTION

Using actual traffic data from a transportation network, we measure its performance by using the total travel time all drivers experience across different Origin-Destination (OD) pairs. Our objective is to compare two different strategies: a user-centric strategy vs. a system-centric one and quantify the difference between the two through the Price of Anarchy (POA) metric.

More specifically, for a user-optimal strategy we face a system with non-cooperative agents (drivers) in which each individual driver follows her personally optimal policy without caring about the system performance. Such "selfish behavior" results in a Nash (Wardrop) equilibrium, a point where no agent can benefit by altering its actions assuming that the actions of all the other agents remain fixed (see Youn et al. (2008)). The equilibrium flows in traffic networks, known as the "Wardrop equilibrium," (see Patriksson (2015)) is the solution to the Traffic Assignment Problem (TAP) that will be described in the next section. It is known that the Nash equilibrium is usually suboptimal from the system point of view.

On the other hand, under a system-optimal policy we enjoy a system in which all agents (drivers) cooperate in order to achieve a social optimum, which is the best policy for the society as a whole (see Youn et al. (2008)). To assess the scope of improvement in a transportation system, we define the Price of Anarchy (POA) as a measure to quantify the system inefficiency under a user-optimal strategy vs. a system-optimal strategy. The latter can be implemented, for example, by taking advantage of the emergence of Connected Automated Vehicles (CAVs) through which socially optimal routing decisions can be executed.

In Zhang et al. (2016) we have studied a similar problem using actual traffic data from the Eastern Massachusetts (EMA) road network. In particular we have investigated the system performance of a relatively small interstate highway subnetwork of EMA.

⋆ Research partially supported by the NSF under grants CNS-1239021, ECCS-1509084, CCF-1527292, IIS-1237022, and IIP-1430145, by the AFOSR under grant FA9550-15-1-0471, by the ARO under grants W911NF-11-1-0227 and W911NF-12-1-0390, and by The MathWorks.

The first goal of this paper is to quantify the POA for a general single-class transportation network model. We take similar steps as in Zhang et al. (2016) which include inferring equilibrium flows from the speed dataset, estimating OD demand matrices, estimating per-road travel latency cost functions, and calculating social optimum flows in order to quantify the POA. In contrast to Zhang et al. (2016), here we are able to analyze a much larger subnetwork of the EMA road network in the sense that it includes a much larger number of nodes and links.

To achieve this goal, we first propose an alternative formulation for the travel latency cost function estimation problem, which enables us to overcome numerical difficulties when the network gets larger. We then modify existing schemes to perform OD demand estimation for large-scale road networks, regardless of whether they exhibit congestion or not. In particular, we leverage a bi-level optimization problem formulation allowing us to estimate OD demand matrices for a given larger network based on the OD demands of its representative (landmark) subnetworks. Utilizing the estimated travel latency cost functions and OD demand matrices, we then quantify the POA.

Another goal of this paper is to analyze the sensitivities of the total users' travel latency costs with respect to key parameters such as road capacities and minimum travel times. The results would help prioritize road segments for interventions that can mitigate congestion. We derive sensitivity analysis formulae for the TAP and apply them to the same interstate highway subnetwork of EMA as that in Zhang et al. (2016).

The rest of the paper is organized as follows: In Sec. 2 we introduce models and methods that we use. In Sec. 3 we describe the datasets and explain the data processing procedures. Numerical results are shown in Sec. 4.

Notation: All vectors are column vectors. For economy of space, we write $\mathbf{x} = \left(x_1, \ldots, x_{\dim(\mathbf{x})}\right)$ to denote the column vector $\mathbf{x}$, where $\dim(\mathbf{x})$ is the dimension of $\mathbf{x}$. We use prime ($\prime$) to denote the transpose of a matrix or vector. Unless otherwise specified, $\|\cdot\|$ denotes the ℓ_2 norm. $|\mathcal{D}|$ denotes the cardinality of a set $\mathcal{D}$. $A \stackrel{\text{def}}{=} B$ indicates A is defined using B.

2. MODELS AND METHODS

2.1 Model for a single-class transportation network

We begin by reviewing the model of Zhang et al. (2016). Denote a road network by $(\mathcal{V}, \mathcal{A}, \mathcal{W})$, where $(\mathcal{V}, \mathcal{A})$ forms a directed graph with $\mathcal{V}$ being the set of nodes and $\mathcal{A}$ the set of links, and $\mathcal{W} = \{\mathbf{w}_i : \mathbf{w}_i = (w_{si}, w_{ti}), i = 1, \ldots, |\mathcal{W}|\}$ indicates the set of all OD pairs. Assume the graph $(\mathcal{V}, \mathcal{A})$ is strongly connected and let $\mathbf{N} \in \{0, 1, -1\}^{|\mathcal{V}| \times |\mathcal{A}|}$ be its node-link incidence matrix. Denote by $\mathbf{e}_a$ the vector with an entry being 1 corresponding to link a and all the other entries being 0. For any OD pair $\mathbf{w} = (w_s, w_t)$, denote by $d^{\mathbf{w}} \geq 0$ the amount of the flow demand from w_s to w_t. Let $\mathbf{d}^{\mathbf{w}} \in \mathbb{R}^{|\mathcal{V}|}$ be the vector which is all zeros, except for a $-d^{\mathbf{w}}$ (resp., $d^{\mathbf{w}}$) in the coordinate mapping to node w_s (resp., w_t).

Denote by $\mathcal{R}_i$ the set of simple routes (a route without cycles is called "*simple route*") between OD pair i. For each $a \in \mathcal{A}$, $i \in \{1, \ldots, |\mathcal{W}|\}$, $r \in \mathcal{R}_i$, define the link-route incidence by

$$\delta^i_{ra} = \begin{cases} 1, & \text{if route } r \in \mathcal{R}_i \text{ uses link } a, \\ 0, & \text{otherwise.} \end{cases}$$

Let x_a denote the flow on link $a \in \mathcal{A}$ and $\mathbf{x} \stackrel{\text{def}}{=} (x_a;\ a \in \mathcal{A})$ the flow vector. Denote by $t_a(\mathbf{x}) : \mathbb{R}^{|\mathcal{A}|}_+ \to \mathbb{R}_+$ the *travel latency cost* (i.e., *travel time*) function for link $a \in \mathcal{A}$. If for all $a \in \mathcal{A}$, $t_a(\mathbf{x})$ only depends on x_a, we say the cost function $\mathbf{t}(\mathbf{x}) = (t_a(x_a);\ a \in \mathcal{A})$ is *separable* (see Patriksson (2015)). Throughout the paper, we assume that the travel latency cost functions are separable and take the following form (see Bertsimas et al. (2015) & Branston (1976)):

$$t_a(x_a) = t^0_a f\left(\frac{x_a}{m_a}\right), \tag{1}$$

where t^0_a is the *free-flow travel time* of $a \in \mathcal{A}$, $f(0) = 1$, $f(\cdot)$ is strictly increasing and continuously differentiable on $\mathbb{R}_+$, and m_a is the *flow capacity* of $a \in \mathcal{A}$.

Define the set of feasible flow vectors $\mathcal{F}$ as (see Bertsimas et al. (2015))

$$\left\{\mathbf{x} : \exists \mathbf{x}^{\mathbf{w}} \in \mathbb{R}^{|\mathcal{A}|}_+ \text{ s.t. } \mathbf{x} = \sum_{\mathbf{w} \in \mathcal{W}} \mathbf{x}^{\mathbf{w}}, \mathbf{N}\mathbf{x}^{\mathbf{w}} = \mathbf{d}^{\mathbf{w}}, \forall \mathbf{w} \in \mathcal{W}\right\},$$

where $\mathbf{x}^{\mathbf{w}}$ indicates the flow vector attributed to OD pair $\mathbf{w}$. In order to formulate appropriate optimization and inverse optimization problems arising in transportation networks, we state the definition of *Wardrop equilibrium*:

Definition 1
(Patriksson (2015)). A feasible flow $\mathbf{x}^* \in \mathcal{F}$ is a *Wardrop equilibrium* if for every OD pair $\mathbf{w} \in \mathcal{W}$, and any route connecting (w_s, w_t) with positive flow in $\mathbf{x}^*$, the cost of traveling along that route is no greater than the cost of traveling along any other route that connects (w_s, w_t). Here, the cost of traveling along a route is the sum of the costs of each of its constituent links.

2.2 The forward problem

As in Zhang et al. (2016), here we refer to the Traffic Assignment Problem (TAP) as the *forward problem*, whose goal is to find the Wardrop equilibrium for a given single-class transportation network with a given OD demand matrix. It is a well-known fact that, for network $(\mathcal{V}, \mathcal{A}, \mathcal{W})$, the TAP can be formulated as the following optimization problem (see Dafermos and Sparrow (1969) & Patriksson (2015)):

$$\text{(TAP)} \quad \min_{\mathbf{x} \in \mathcal{F}} \sum_{a \in \mathcal{A}} \int_0^{x_a} t_a(s) ds. \tag{2}$$

As an alternative, we also formulate the TAP as a Variational Inequality (VI) problem:

Definition 2
(Bertsimas et al. (2015)). The VI problem, denoted as $\text{VI}(\mathbf{t}, \mathcal{F})$, is to find an $\mathbf{x}^* \in \mathcal{F}$ s.t.

$$\mathbf{t}(\mathbf{x}^*)'(\mathbf{x} - \mathbf{x}^*) \geq 0, \quad \forall \mathbf{x} \in \mathcal{F}. \tag{3}$$

We introduce an assumption:

Assumption A
$\mathbf{t}(\cdot)$ is strongly monotone (see Patriksson (2015) or Bertsimas et al. (2015) for the definition of strong monotonicity) and continuously differentiable on $\mathbb{R}_+^{|\mathcal{A}|}$. $\mathcal{F}$ is nonempty and contains an interior point (Slater's condition).

Theorem 1. (Patriksson (2015)). *Assump. A* implies that there exists a Wardrop equilibrium of the network $(\mathcal{V}, \mathcal{A}, \mathcal{W})$, which is the unique solution to $\mathrm{VI}(\mathbf{t}, \mathcal{F})$.

2.3 The inverse problem

To solve the forward problem, we need to know the travel latency cost function and the OD demand matrix. Assuming that we know the OD demand matrix and have observed the Wardrop equilibrium flows, we seek to formulate the inverse problem (the inverse VI problem, in particular), so as to estimate the travel latency cost function.

For a given $\epsilon > 0$, we define an *ϵ-approximate solution* to $\mathrm{VI}(\mathbf{t}, \mathcal{F})$ by changing the right-hand side of (3) to $-\epsilon$:

Definition 3
(Bertsimas et al. (2015)). Given $\epsilon > 0$, $\hat{\mathbf{x}} \in \mathcal{F}$ is called an *ϵ-approximate solution* to $\mathrm{VI}(\mathbf{t}, \mathcal{F})$ if

$$\mathbf{t}(\hat{\mathbf{x}})'(\mathbf{x} - \hat{\mathbf{x}}) \geq -\epsilon, \quad \forall \mathbf{x} \in \mathcal{F}. \tag{4}$$

Given K observations $(\mathbf{x}_k, \mathcal{F}_k)$, $k = 1, \ldots, K$, with $\mathbf{x}_k \in \mathcal{F}_k$ and each $\mathcal{F}_k$ being a set of feasible flow vectors meeting Slater's condition (see Boyd and Vandenberghe (2004)), the inverse VI problem amounts to finding a function $\mathbf{t}$ such that $\mathbf{x}_k$ is an ϵ_k-approximate solution to $\mathrm{VI}(\mathbf{t}, \mathcal{F}_k)$ for each k. Denoting $\boldsymbol{\epsilon} \stackrel{\text{def}}{=} (\epsilon_k;\ k = 1, \ldots, K)$, we can formulate the inverse VI problem as (see Bertsimas et al. (2015))

$$\begin{aligned} \min_{\mathbf{t}, \boldsymbol{\epsilon}} \quad & \|\boldsymbol{\epsilon}\| \\ \text{s.t.} \quad & \mathbf{t}(\mathbf{x}_k)'(\mathbf{x} - \mathbf{x}_k) \geq -\epsilon_k, \quad \forall \mathbf{x} \in \mathcal{F}_k,\ \forall k, \\ & \epsilon_k > 0, \quad \forall k. \end{aligned} \tag{5}$$

To make (5) solvable, we apply the nonparametric estimation approach which expresses the cost function in a Reproducing Kernel Hilbert Space (RKHS) (see Bertsimas et al. (2015) & Evgeniou et al. (2000)). To be specific, we pick the polynomial kernel, i.e., $\phi(x, y) \stackrel{\text{def}}{=} (c + xy)^n$ for some choice of $c \geq 0$ and $n \in \mathbb{N}$ (for the specifications of c and n, see Zhang (2016)). Assume we are given K networks $(\mathcal{V}_k, \mathcal{A}_k, \mathcal{W}_k)$, $k = 1, \ldots, K$, (as a special case, these could be K replicas of the same network $(\mathcal{V}, \mathcal{A}, \mathcal{W})$) and the normalized flow data $\{\mathbf{x}^k = (x_a^k / m_a^k;\ a \in \mathcal{A}_k);\ k = 1, \ldots, K\}$, where k is the network index and x_a^k (resp., m_a^k) is the flow (resp., capacity) for link $a \in \mathcal{A}_k$ correspondingly. Let $M = \sum_{k=1}^K |\mathcal{A}_k|$ and $\mathbf{z} = (z_1, \ldots, z_M) \stackrel{\text{def}}{=} ((\mathbf{x}^1)', \ldots, (\mathbf{x}^K)')$. Take the kernel matrix to be $\boldsymbol{\Phi} = [\phi(z_i, z_j)]_{i,j=1}^M$. By (Bertsimas et al., 2015, Thm. 2), we reformulate the inverse VI problem (5) as the following Quadratic Programming (QP) problem:

$$\begin{aligned} \min_{\boldsymbol{\alpha}, \mathbf{y}, \boldsymbol{\epsilon}} \quad & \boldsymbol{\alpha}'\boldsymbol{\Phi}\boldsymbol{\alpha} + \gamma \|\boldsymbol{\epsilon}\| \\ \text{s.t.} \quad & \mathbf{e}_a'\mathbf{N}_k'\mathbf{y}^{\mathbf{w}} \leq t_a^0 \boldsymbol{\alpha}'\boldsymbol{\Phi}\mathbf{e}_a, \\ & \qquad \forall \mathbf{w} \in \mathcal{W}_k,\ a \in \mathcal{A}_k,\ k = 1, \ldots, K, \\ & \boldsymbol{\alpha}'\boldsymbol{\Phi}\mathbf{e}_a \leq \boldsymbol{\alpha}'\boldsymbol{\Phi}\mathbf{e}_{\tilde{a}}, \\ & \qquad \forall a,\ \tilde{a} \in \bigcup_{k=1}^K \mathcal{A}_k \ \text{ s.t. } \frac{x_a}{m_a} \leq \frac{x_{\tilde{a}}}{m_{\tilde{a}}}, \\ & \sum_{a \in \mathcal{A}_k} t_a^0 x_a \boldsymbol{\alpha}'\boldsymbol{\Phi}\mathbf{e}_a - \sum_{\mathbf{w} \in \mathcal{W}_k} (\mathbf{d}^{\mathbf{w}})'\mathbf{y}^{\mathbf{w}} \leq \epsilon_k, \\ & \qquad \forall k = 1, \ldots, K, \\ & \epsilon_k > 0, \quad \forall k = 1, \ldots, K, \\ & \boldsymbol{\alpha}'\boldsymbol{\Phi}\mathbf{e}_{a_0} = 1, \end{aligned} \tag{6}$$

where $\boldsymbol{\alpha}$, $\mathbf{y} = (\mathbf{y}^{\mathbf{w}}; \mathbf{w} \in \mathcal{W}_k, k = 1, \ldots, K)$, and $\boldsymbol{\epsilon} = (\epsilon_k; k = 1, \ldots, K)$ are decision vectors ($\mathbf{y}^{\mathbf{w}}$ is the "price" of $\mathbf{d}^{\mathbf{w}}$, in particular), γ is a regularization parameter, and a_0 is some arbitrary link chosen for normalization purposes. The first constraint is for dual feasibility. The second constraint forces the function $f(\cdot)$ (the cornerstone of the travel latency cost function $\mathbf{t}(\cdot)$) to be non-decreasing on $\bigcup_{k=1}^K \mathcal{A}_k$. The third constraint is the primal-dual gap constraint.

Theoretically, we can derive an estimator $\bar{f}(\cdot)$ of $f(\cdot)$ by solving the QP (6), thereby obtaining an optimal $\boldsymbol{\alpha}^*$. Writing $\boldsymbol{\alpha}^* = (\alpha_1^*, \ldots, \alpha_M^*)$, then, by (Bertsimas et al., 2015, Thm. 4), we obtain

$$\bar{f}(\cdot) = \sum_{m=1}^M \alpha_m^* \boldsymbol{\phi}(z_m, \cdot). \tag{7}$$

However, directly solving (6) would typically encounter numerical difficulties, due to the low rank of the kernel matrix $\boldsymbol{\Phi}$ (typically in networks with many links). Thus, in our implementation, we will consider another formulation, by directly using the polynomial base functions of the same RKHS. In particular, by (Evgeniou et al., 2000, (3.2), (3.3), and (3.6)), writing

$$\phi(x, y) = (c + xy)^n = \sum_{i=0}^n \binom{n}{i} c^{n-i} x^i y^i,$$

we reformulate (6) as

$$\begin{aligned} \min_{\boldsymbol{\beta}, \mathbf{y}, \boldsymbol{\epsilon}} \quad & \sum_{i=0}^n \frac{\beta_i^2}{\binom{n}{i} c^{n-i}} + \gamma \|\boldsymbol{\epsilon}\| \\ \text{s.t.} \quad & \mathbf{e}_a'\mathbf{N}_k'\mathbf{y}^{\mathbf{w}} \leq t_a^0 \sum_{i=0}^n \beta_i \left(\frac{x_a}{m_a}\right)^i, \\ & \qquad \forall \mathbf{w} \in \mathcal{W}_k,\ a \in \mathcal{A}_k,\ k = 1, \ldots, K, \\ & \sum_{i=0}^n \beta_i \left(\frac{x_a}{m_a}\right)^i \leq \sum_{i=0}^n \beta_i \left(\frac{x_{\tilde{a}}}{m_{\tilde{a}}}\right)^i, \\ & \qquad \forall a,\ \tilde{a} \in \mathcal{A}_0 \text{ s.t. } \frac{x_a}{m_a} \leq \frac{x_{\tilde{a}}}{m_{\tilde{a}}}, \\ & \sum_{a \in \mathcal{A}_k} t_a^0 x_a \sum_{i=0}^n \beta_i \left(\frac{x_a}{m_a}\right)^i - \sum_{\mathbf{w} \in \mathcal{W}_k} (\mathbf{d}^{\mathbf{w}})'\mathbf{y}^{\mathbf{w}} \leq \epsilon_k, \\ & \qquad \forall k = 1, \ldots, K, \\ & \epsilon_k > 0, \quad \forall k = 1, \ldots, K, \\ & \beta_0 = 1, \end{aligned} \tag{8}$$

where the last constraint is for normalization purposes, which makes sense, because an updated estimator $\hat{f}(\cdot)$ (as opposed to $\bar{f}(\cdot)$ in (7)) should at least satisfy $\hat{f}(0) =$

1 (corresponding to the free-flow travel time; see (1)). Assuming an optimal $\boldsymbol{\beta}^* = (\beta_i^*;\ i = 0, 1, \ldots, n)$ is obtained by solving (8), then our final estimator for $f(\cdot)$ is

$$\hat{f}(x) = \sum_{i=0}^{n} \beta_i^* x^i = 1 + \sum_{i=1}^{n} \beta_i^* x^i. \tag{9}$$

2.4 OD demand estimation

Given network $(\mathcal{V}, \mathcal{A}, \mathcal{W})$, to estimate an initial OD demand matrix, we borrow the Generalized Least Squares (GLS) method proposed in Hazelton (2000), which assumes that the transportation network $(\mathcal{V}, \mathcal{A}, \mathcal{W})$ is uncongested (in other words, for each OD pair the route choice probabilities are independent of traffic flow), and that the OD trips (traffic counts) are Poisson distributed. Note that such assumptions are a bit strong and we will relax them when finalizing our OD demand estimator by performing an adjustment procedure.

Let $\{\mathbf{x}^{(k)};\ k = 1, \ldots, K\}$ denote K observations of the flow vector and $\bar{\mathbf{x}}$ the average. Denote by $\mathbf{S} = (1/(K-1)) \sum_{k=1}^{K} \left(\mathbf{x}^{(k)} - \bar{\mathbf{x}}\right)\left(\mathbf{x}^{(k)} - \bar{\mathbf{x}}\right)'$ the sample covariance matrix. Let $\mathbf{P} = [p_{ir}]$ be the route choice probability matrix, where p_{ir} is the probability that a traveler between OD pair i uses route r. Vectorize the OD demand matrix as $\mathbf{g}$ and let $\boldsymbol{\xi} = \mathbf{P}'\mathbf{g}$. Then, the GLS method proceeds as follows:

(i) Find feasible routes for each OD pair, thus obtaining the link-route incidence matrix $\mathbf{A}$.

(ii) Solve sequentially the following two problems:

$$\text{(P1)} \quad \min_{\boldsymbol{\xi} \succeq \mathbf{0}} \quad \frac{K}{2} \boldsymbol{\xi}' \mathbf{Q} \boldsymbol{\xi} - \mathbf{b}' \boldsymbol{\xi}, \tag{10}$$

where $\mathbf{Q} = \mathbf{A}'\mathbf{S}^{-1}\mathbf{A}$ and $\mathbf{b} = \sum_{k=1}^{K} \mathbf{A}'\mathbf{S}^{-1}\mathbf{x}^{(k)}$, and

$$\begin{aligned} \text{(P2)} \quad & \min_{\mathbf{P} \succeq \mathbf{0},\, \mathbf{g} \succeq \mathbf{0}} \quad h(\mathbf{P}, \mathbf{g}) \\ \text{s.t.} \quad & p_{ir} = 0 \quad \forall (i, r) \in \{(i, r) : r \notin \mathcal{R}_i\}, \\ & \mathbf{P}'\mathbf{g} = \boldsymbol{\xi}^*, \\ & \mathbf{P}\mathbf{1} = \mathbf{1}, \end{aligned} \tag{11}$$

where $h(\mathbf{P}, \mathbf{g})$ can be taken as any smooth scalar-valued function, $\boldsymbol{\xi}^*$ is the optimal solution to (P1), and $\mathbf{1}$ denotes the vector with all 1's as its entries. Note that (P1) (resp., (P2)) is a typical *Quadratic Programing (QP)* (resp., *Quadratically Constrained Programming (QCP)*) problem. Letting $(\mathbf{P}^*, \mathbf{g}^*)$ be an optimal solution to (P2), then $\mathbf{g}^*$ is our initial estimate of the demand vector.

Remark 1
We note here that the GLS method would encounter numerical difficulties when the network size is large, because there would be too many decision variables in (P2). Note also that the GLS method is valid under an "uncongestion" assumption and, to take the congestion, here the link flows, into account, we in turn consider a bi-level optimization problem in the following.

Assume now the function $f(\cdot)$ in (1) is available. For any given feasible $\mathbf{g}$ ($\succeq \mathbf{0}$), let $\mathbf{x}(\mathbf{g})$ be the optimal solution to the TAP (2). In the following, assume $\tilde{\mathbf{x}} = (\tilde{x}_a;\ a \in \mathcal{A})$ to be the observed flow vector. Assuming we are given an initial demand vector $\mathbf{g}^0$, we consider the following bi-level optimization problem (BiLev) (see Spiess (1990) & Lundgren and Peterson (2008)):

$$\text{(BiLev)} \quad \min_{\mathbf{g} \succeq \mathbf{0}} \ F(\mathbf{g}) \stackrel{\text{def}}{=} \sum_{a \in \mathcal{A}} \left(x_a(\mathbf{g}) - \tilde{x}_a\right)^2. \tag{12}$$

Note that $F(\mathbf{g})$ has a lower bound 0 which guarantees the convergence of the algorithm (see Alg. 1) that we are going to apply.

To solve the (BiLev) numerically, thus adjusting the demand vector iteratively, we leverage a gradient-based algorithm (Alg. 1). In particular, suppose that the route probabilities are locally constant. For OD pair i, considering only the shortest route $r_i(\mathbf{g})$, we have (see Spiess (1990))

$$\frac{\partial x_a(\mathbf{g})}{\partial g_i} = \delta_{r_i(\mathbf{g})a} = \begin{cases} 1, & \text{if } a \in r_i(\mathbf{g}), \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

Thus, by (13) we obtain the Jacobian matrix

$$\left[\frac{\partial x_a(\mathbf{g})}{\partial g_i};\ a \in \mathcal{A},\ i \in \{1, \ldots, |\mathcal{W}|\}\right]. \tag{14}$$

Remark 2
There are three reasons why we consider only the shortest routes for the purpose of calculating the Jacobian: (i) GPS navigation is widely-used by vehicle drivers so that they tend to always select the shortest routes between their OD pairs. (ii) There are many efficient algorithms for finding the shortest route for each OD pair. (iii) If considering more than one route for an OD pair, then the route flows cannot be uniquely determined by solving the TAP (2), thus leading to unstable route-choice probabilities, which would undermine the accuracy of the approximation to the Jacobian matrix in (14).

Let us now compute the gradient of $F(\mathbf{g})$. We have

$$\begin{aligned} \nabla F(\mathbf{g}) &= \left(\frac{\partial F(\mathbf{g})}{\partial g_i};\ i = 1, \ldots, |\mathcal{W}|\right) \\ &= \left(\sum_{a \in \mathcal{A}} 2\left(x_a(\mathbf{g}) - \tilde{x}_a\right) \frac{\partial x_a(\mathbf{g})}{\partial g_i};\ i = 1, \ldots, |\mathcal{W}|\right). \end{aligned} \tag{15}$$

We summarize the procedures for adjusting the OD demand matrices as Alg. 1, whose convergence will be proven in the following proposition.

Proposition 2. Alg. 1 converges.

Proof. If the initial demand vector $\mathbf{g}^0$ satisfies $F\left(\mathbf{g}^0\right) = 0$, then, by Step 0, the algorithm stops (trivial case). Otherwise, we have $F\left(\mathbf{g}^0\right) > 0$, and it is seen from (12) that the objective function $F(\mathbf{g})$ has a lower bound 0. In addition, by the line search and the update steps (Steps 3.2 and 4.1, in particular), we obtain

$$F\left(\mathbf{g}^{l+1}\right) = F\left(\mathbf{g}^l + \theta^l \bar{\mathbf{h}}^l\right) = \min_{\theta \in \mathcal{S}} F\left(\mathbf{g}^l + \theta \bar{\mathbf{h}}^l\right) \le F\left(\mathbf{g}^l\right), \forall l,$$

where the last inequality holds due to $0 \in \mathcal{S}$, indicating that the nonnegative objective function in (12) is nonincreasing as the number of iterations increases. Thus, by the well-known monotone convergence theorem, the convergence of the algorithm can be guaranteed. □

Remark 3
Alg. 1 is a variant of the algorithms proposed in Spiess (1990) & Lundgren and Peterson (2008). We use a different method to calculate the *step-sizes* (resp., *Jacobian matrix*) than that in Spiess (1990) (resp., Lundgren and Peterson (2008)). Moreover, as a subroutine, Alg. 2 is borrowed from

Algorithm 1 Adjusting OD demand matrices

Input: the road network $(\mathcal{V}, \mathcal{A}, \mathcal{W})$; the function $f(\cdot)$ in (1); the observed flow vector $\tilde{\mathbf{x}} = (\tilde{x}_a;\ a \in \mathcal{A})$; the initial demand vector $\mathbf{g}^0 = (g_i^0;\ i = 1, \ldots, |\mathcal{W}|)$; two positive integer parameters ρ, T; two real parameters $\varepsilon_1 \geq 0$, $\varepsilon_2 > 0$.

Step 0: Initialization. Take the demand vector $\mathbf{g}^0$ as the input, solve the TAP (2) using Alg. 2 to obtain $\mathbf{x}^0$. Set $l = 0$. If $F\left(\mathbf{g}^0\right) = 0$, stop; otherwise, go onto Step 1.

Step 1: Computation of a descent direction. Calculate $\mathbf{h}^l = -\nabla F\left(\mathbf{g}^l\right)$ by (15).

Step 2: Calculation of a search direction. For $i = 1, \ldots, |\mathcal{W}|$ set

$$\bar{h}_i^l = \begin{cases} h_i^l, \text{ if } \left(g_i^l > \varepsilon_1\right) \text{ or } \left(g_i^l \leq \varepsilon_1 \text{ and } h_i^l > 0\right), \\ 0, \text{ otherwise.} \end{cases}$$

Step 3: Armijo-type line search:

- **3.1:** Calculate the maximum possible step-size $\theta_{\max}^l = \min\left\{-\frac{g_i^l}{\bar{h}_i^l};\ \bar{h}_i^l < 0, i = 1, \ldots, |\mathcal{W}|\right\}$.
- **3.2:** Determine $\theta^l = \arg\min_{\theta \in \mathcal{S}} F\left(\mathbf{g}^l + \theta \bar{\mathbf{h}}^l\right)$, where $\mathcal{S} \stackrel{\text{def}}{=} \left\{\theta_{\max}^l, \theta_{\max}^l/\rho, \theta_{\max}^l/\rho^2, \ldots, \theta_{\max}^l/\rho^T, 0\right\}$.

Step 4: Update and termination.

- **4.1:** Set $\mathbf{g}^{l+1} = \mathbf{g}^l + \theta^l \bar{\mathbf{h}}^l$. Using $\mathbf{g}^{l+1}$ as the input, solve the TAP (2) to obtain $\mathbf{x}^{l+1}$.
- **4.2:** If $\frac{F\left(\mathbf{g}^l\right) - F\left(\mathbf{g}^{l+1}\right)}{F\left(\mathbf{g}^0\right)} < \varepsilon_2$, stop the iteration; otherwise, go onto Step 4.3.
- **4.3:** Set $l = l + 1$ and return to Step 1.

Algorithm 2 Method of Successive Averages (MSA) (see Noriega and Florian (2007))

Input: the road network $(\mathcal{V}, \mathcal{A}, \mathcal{W})$; the function $f(\cdot)$ in (1); the demand vector $\mathbf{g} = (g_i;\ i = 1, \ldots, |\mathcal{W}|)$; a real parameter $\varepsilon > 0$.

Step 0: Initialization. Initialize link flows: $x_a^\ell = 0$ for $a \in \mathcal{A}$; set iteration counter $\ell = 0$.

Step 1: Compute new extremal flow. Set $\ell = \ell + 1$.

- **1.1:** Update link travel times (costs) based on current link flows: $t_a^\ell = t_a\left(x_a^{\ell-1}\right)$.
- **1.2:** Carry out "all-or-nothing" assignment of the demands $\mathbf{g}$ on current shortest paths to obtain y_a^ℓ.

Step 2: Update link flows.

Update link flows by using the MSA step size:

$$x_a^\ell = x_a^{\ell-1} + \lambda^\ell\left(y_a^\ell - x_a^{\ell-1}\right),$$

where $\lambda^\ell = 1/\ell$.

Step 3: Stopping criterion (slightly different than that in Noriega and Florian (2007)).

Compute the *Relative Gap*, *RG*, as:

$$RG = \frac{\left\|\mathbf{x}^\ell - \mathbf{x}^{\ell-1}\right\|}{\left\|\mathbf{x}^\ell\right\|}.$$

If $RG < \varepsilon$ terminate; otherwise return to Step 1.

Noriega and Florian (2007), which has the advantage of being easy to implement.

2.5 Price of anarchy

As discussed in Sec. 1, one of our goals is to measure inefficiency in the traffic network performance due to the non-cooperative behavior of drivers. Thus, we compare the network performance under a user-centric policy vs. a system-centric policy. As a metric for this comparison we define the POA as the ratio between the total latency, i.e., the total travel time over all drivers in different OD pairs, obtained under Wardrop flows (user-centric policy) and that obtained under socially optimal flows (system-centric policy).

Given network $(\mathcal{V}, \mathcal{A}, \mathcal{W})$, as in Zhang et al. (2016), we define its *total latency* as

$$L(\mathbf{x}) = \sum_{a \in \mathcal{A}} x_a t_a(x_a). \tag{16}$$

Then, the socially optimum flow vector, denoted by $\mathbf{x}^*$, is the solution to the following Non-Linear Programming (NLP) problem (see Patriksson (2015) & Pourazarm et al. (2016)):

$$\min_{\mathbf{x} \in \mathcal{F}} \sum_{a \in \mathcal{A}} x_a t_a(x_a). \tag{17}$$

Note that here the objective function is different from the one we use in the TAP (see Sec. 2.2); for a detailed explanation, see Dafermos and Sparrow (1969). Recalling that we consider the total latency as the performance metric, we define the *Price of Anarchy* as

$$\text{POA} \stackrel{\text{def}}{=} \frac{L(\mathbf{x}^{ne})}{L(\mathbf{x}^*)} = \frac{\sum_{a \in \mathcal{A}} x_a^{ne} t_a(x_a^{ne})}{\sum_{a \in \mathcal{A}} x_a^* t_a(x_a^*)}, \tag{18}$$

where x_a^{ne} denotes the Wardrop's equilibrium flow on link $a \in \mathcal{A}$ estimated from the speed dataset (for details refer to Zhang et al. (2016)), and x_a^* is the socially optimal flow on link $a \in \mathcal{A}$ obtained by solving (17), which requires the travel latency cost functions $t_a(\cdot)$ that are estimated by solving the inverse problem (see Sec. 2.3). Thus, POA quantifies the inefficiency that a society has to deal with due to non-cooperative behavior of its members.

2.6 Sensitivity analysis

Write $\mathbf{t}^0 \stackrel{\text{def}}{=} \left(t_a^0;\ a \in \mathcal{A}\right)$, $\mathbf{m} \stackrel{\text{def}}{=} (m_a;\ a \in \mathcal{A})$, and

$$V\left(\mathbf{t}^0, \mathbf{m}\right) \stackrel{\text{def}}{=} \min_{\mathbf{x} \in \mathcal{F}} \sum_{a \in \mathcal{A}} \int_0^{x_a} t_a^0 f\left(\frac{s}{m_a}\right) ds. \tag{19}$$

Let $\mathbf{x}^* \stackrel{\text{def}}{=} (x_a^*;\ a \in \mathcal{A})$ denote the solution to (2). Using (Simon and Blume, 1994, Thm. 19.5), we obtain

$$\frac{\partial V\left(\mathbf{t}^0, \mathbf{m}\right)}{\partial t_a^0} = \int_0^{x_a^*} f\left(\frac{s}{m_a}\right) ds, \tag{20}$$

$$\frac{\partial V\left(\mathbf{t}^0, \mathbf{m}\right)}{\partial m_a} = \int_0^{x_a^*} t_a^0 \dot{f}\left(\frac{s}{m_a}\right)\left(-\frac{s}{m_a^2}\right) ds, \tag{21}$$

where $\dot{f}(\cdot)$ denotes the derivative of $f(\cdot)$.

3. DATASET DESCRIPTION AND PROCESSING

3.1 Description of the Eastern Massachusetts dataset

We deal with two datasets for the Eastern Massachusetts (EMA) road network: (i) The speed dataset, provided to us

by the Boston Region Metropolitan Planning Organization (MPO), includes the spatial average speeds for more than 13,000 road segments (with the average distance of 0.7 miles; see Fig. 1) of EMA, covering the average speed for every minute of the year 2012. For each road segment, identified with a unique *tmc (traffic message channel)* code, the dataset provides information such as speed data (instantaneous, average and free-flow speed) in mph, date and time, and traveling time (minute) through that segment. (ii) The flow capacity (# of vehicles per hour) dataset, also provided by the MPO, includes capacity data – vehicle counts for each road segment – for more than 100,000 road segments (average distance of 0.13 miles) in EMA. For more detailed information of these two datasets, see Zhang et al. (2016).

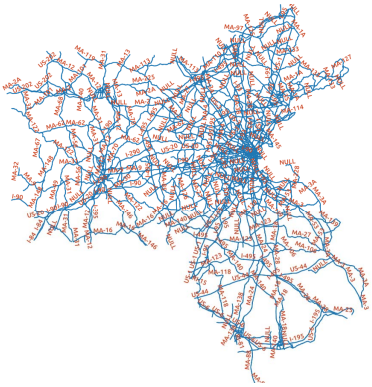


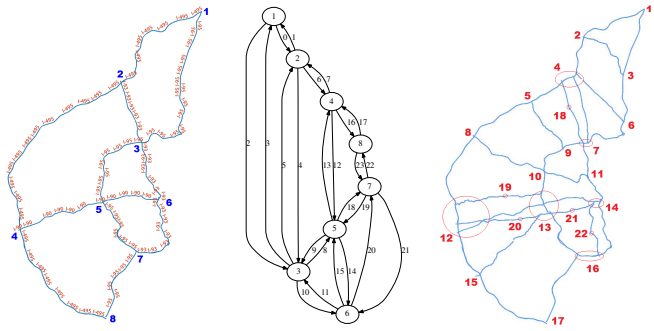
Fig. 1. All available road segments in Eastern Massachusetts (from Zhang et al. (2016)).

3.2 Preprocessing

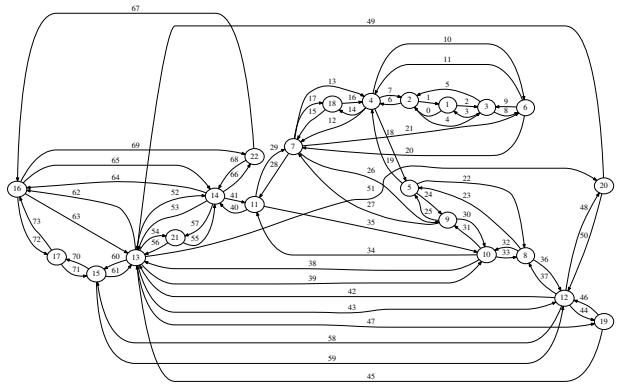
Other than selecting an extended subnetwork (shown in Fig. 2(c) as opposed to the smaller subnetwork Fig. 2(a)) of the EMA road network (shown in Fig. 1), we perform exactly the same preprocessing procedures as those in Zhang et al. (2016) on the datasets mentioned in Sec. 3.1. Thus, we end up with traffic flow data and road (link) parameters (*flow capacity* and *free-flow travel time*) for the two subnetworks (shown in Figs. 2(a) and 2(c) respectively; from now on, let us denote them by $\mathcal{I}_1$ and $\mathcal{I}_2$ respectively), where $\mathcal{I}_1$ contains only the interstate highways and $\mathcal{I}_2$ also contains the state highways in EMA. Note that $\mathcal{I}_1$ (resp., $\mathcal{I}_2$) consists of 8 (resp., 22) nodes and 24 (resp., 74) links; their topologies are shown in Figs. 2(b) and 2(d) respectively. Assuming that each node could be an origin and a destination, then there are $8 \times (8-1) = 56$ (resp., $22 \times (22-1) = 462$) OD pairs in $\mathcal{I}_1$ (resp., $\mathcal{I}_2$).

3.3 Estimating initial OD demand matrices

Operating on $\mathcal{I}_1$, we solve the QP (P1) and the QCP (P2) using data corresponding to five different time periods (AM, MD, PM, NT, and weekend) of four months (Jan., Apr., Jul., and Oct.) in 2012, thus obtaining 20 different OD demand matrices for these scenarios. As noted in Zhang et al. (2016), the GLS method assumes the traffic network to be uncongested, leading to the fact that the estimated OD demand matrices for non-peak periods (MD/NT/weekend) are relatively more accurate than those for peak periods (AM/PM). After obtaining



(a) (from Zhang et al. (2016)) (b) (from Zhang et al. (2016)) (c)



(d)

Fig. 2. (a) An interstate highway sub-network of EMA (the blue numbers indicate node indices); (b) The topology of the sub-network in Fig. 2(a) (the numbers beside arrows are link indices, and the numbers inside ellipses are node indices); (c) An extended highway sub-network of EMA (the red numbers indicate node indices); (d) The topology of the sub-network in Fig. 2(c).

estimates for travel latency cost functions in Sec. 3.4, based on the estimates for the OD demand matrices of $\mathcal{I}_1$, we will conduct a demand adjustment procedure for $\mathcal{I}_2$ in Sec. 3.6.

3.4 Estimating cost functions

Operating on $\mathcal{I}_1$ using the flow data and the OD demand matrices obtained in Secs. 3.2 and 3.3 respectively, we estimate the travel latency cost functions for 20 different scenarios, via the estimator (9), by solving the QP (8) accordingly. As in Zhang (2016), to make the estimates reliable, for each scenario, we perform a 3-fold cross-validation. Note that Zhang (2016) applied the alternative estimator (7), which is numerically not as stable.

3.5 Sensitivity analysis

To illustrate our method of analyzing sensitivities for the TAP, we again conduct the numerical experiments on $\mathcal{I}_1$. Take the scenario corresponding to the PM peak period on January 10, 2012. The results will be shown in Sec. 4.3.

3.6 OD demand adjustments

We first demonstrate the effectiveness of the demand-adjusting algorithm (Alg. 1) using the Sioux-Falls benchmark dataset (see Bar-Gera (2016)). Then, assuming the

per-road travel latency cost functions are available (we take the travel latency cost functions derived from $\mathcal{I}_1$ as in Sec. 3.4), we apply Alg. 1 to $\mathcal{I}_2$, which contains $\mathcal{I}_1$ as one of its representative subnetworks. Note that the main difference between $\mathcal{I}_1$ and $\mathcal{I}_2$ is the modeling emphasis ($\mathcal{I}_1$ only takes account of the interstate highways, while $\mathcal{I}_2$ also encompasses the state highways, thus containing more details of the real road network of EMA), and we can think of $\mathcal{I}_1$ as a "landmark" subnetwork of $\mathcal{I}_2$. Based on the initially estimated demand matrices for $\mathcal{I}_1$, we will implement the following generic demand-adjusting scheme so as to derive the OD demand matrices for $\mathcal{I}_2$:

Given a network ($\mathcal{I}_2$ in our case) with any size, we can select its "landmark" subnetworks ($\mathcal{I}_1$ in our case) (based on the information of road types, pre-identified centroids, etc.) with acceptably smaller sizes, say, we end up with N ($N = 1$ in our case) such subnetworks. Then, for each subnetwork, we estimate its demand matrix by solving sequentially the QP (P1) and the QCP (P2) (cf. Sec. 3.3). Setting the demand for any OD pair not belonging to this subnetwork to zero, we obtain a "rough" initial demand matrix for the entire network ($\mathcal{I}_2$ in our case). Next, take the average of these initial demand matrices. Finally, we adjust the average demand matrix based on the flow observations of the entire network.

As noted in Remark 1, the reason why we do not directly solve (P1) and (P2) for the larger network ($\mathcal{I}_2$ in our case) is that there are too many decision variables in (P2) and this would lead to numerical difficulties.

3.7 POA evaluations

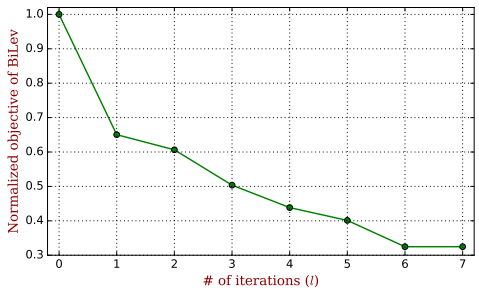
We calculate the POA values for $\mathcal{I}_2$ for the PM period of April 2012.

4. NUMERICAL RESULTS

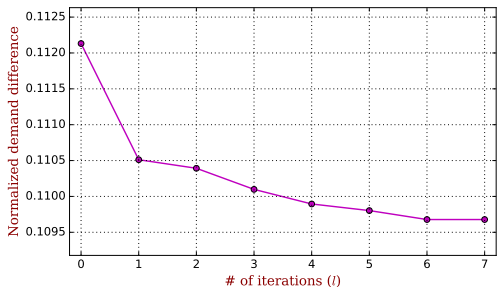
The results for the initial estimation of OD demand matrices and the estimation of travel latency cost functions are similar to those shown in Zhang et al. (2016), and we omit them here. We will focus on presenting the results for sensitivity analysis (derived for $\mathcal{I}_1$), demand adjustment (derived for the Sioux-Falls benchmark network; note that we will not show the detailed demand adjustment results for $\mathcal{I}_2$, because we do not have the ground truth for a comparison), and the POA (derived for $\mathcal{I}_2$).

4.1 Results from OD demand adjustment

We conduct numerical experiments on the Sioux-Falls network, which contains 24 zones, 24 nodes, and 76 links. For each OD pair, the initial demand is taken by scaling the ground truth demand with a uniform distribution $[0.8, 1.2]$. The ground truth $f(\cdot)$ is taken as $f(x) = 1 + 0.15x^4$, $\forall x \geq 0$, and is assumed directly available. When implementing Alg. 1 (use Alg. 2 as a subroutine), we set $\rho = 2$, $T = 10$, $\varepsilon_1 = 0$, $\varepsilon_2 = 10^{-20}$ and $\varepsilon = 10^{-6}$. Fig. 3(a) shows that, after 7 iterations, the objective function value of the (BiLev) (12) has been reduced by more than 65%. Fig. 3(b) shows that, the distance between the adjusted demand and the ground truth demand keeps decreasing with the number of iterations, and the distance changes very slightly, meaning the adjustment procedure does not alter the initial demand much. Note that in Fig. 3(a), the vertical axis corresponds to the normalized objective function value of the (BiLev), i.e., $F\left(\mathbf{g}^l\right)/F\left(\mathbf{g}^0\right)$ and, in Fig. 3(b), the vertical axis denotes the normalized distance between the adjusted demand vector and the ground truth, i.e., $\left\|\mathbf{g}^l - \mathbf{g}^*\right\|/\|\mathbf{g}^*\|$, where $\mathbf{g}^*$ is the ground-truth demand vector.



(a) Objective function value of the (BiLev) vs. # of iterations.



(b) Demand difference w.r.t. ground truth vs. # of iterations.

Fig. 3. Key quantities vs. # of iterations (Sioux-Falls).

4.2 Results for POA estimation

After implementing the demand adjusting scheme, we obtain the demand matrices for $\mathcal{I}_2$ on a daily-basis, as opposed to those for $\mathcal{I}_1$ on a monthly-basis. Note that, even for the same period of a day and within the same month, slight demand variations among different days are possible; thus, our POA results for $\mathcal{I}_2$ would be more accurate than those for $\mathcal{I}_1$ that have been shown in Zhang et al. (2016).

The POA values shown in Fig. 4 have larger variations than those for $\mathcal{I}_1$ in Zhang et al. (2016); some are now closer to 1 but some go beyond 2.2, meaning we have larger potential to improve the road network. However, when taking the average of the POA values for all the 30 days, both $\mathcal{I}_2$ and $\mathcal{I}_1$ result in an average POA approximately equal to 1.5, meaning we can gain an efficiency improvement of about 50%; thus, the results are consistent.

4.3 Results from sensitivity analysis

The sensitivity analysis results derived for $\mathcal{I}_1$ corresponding to the PM peak period on January 10, 2012 are shown in Tab. 1, where each entry is a pair with the first element being the link index and the second element being the

Table 1. Scaled sensitivity analysis results.

$\frac{\partial V(\mathbf{t}^0,\mathbf{m})}{\partial t_a^0}$				$\frac{\partial V(\mathbf{t}^0,\mathbf{m})}{\partial m_a}$			
(0, 0.226)	(1, 0.177)	(2, 0.26)	(3, 0.208)	(0, -0.009)	(1, -0.005)	(2, -0.011)	(3, -0.005)
(4, 0.54)	(5, 0.198)	(6, 0.199)	(7, 0.242)	(4, -0.036)	(5, -0.002)	(6, -0.009)	(7, -0.017)
(8, 0.924)	(9, 0.327)	(10, 0.951)	(11, 0.543)	(8, -0.496)	(9, -0.009)	(10, -0.605)	(11, -0.094)
(12, 0.389)	(13, 0.245)	(14, 0.539)	(15, 0.42)	(12, -0.048)	(13, -0.007)	(14, -0.119)	(15, -0.103)
(16, 0.178)	(17, 0.129)	(18, 0.341)	(19, 1.0)	(16, -0.004)	(17, -0.002)	(18, -0.011)	(19, -0.883)
(20, 0.892)	(21, 0.838)	(22, 0.234)	(23, 0.833)	(20, -0.888)	(21, -0.776)	(22, -0.006)	(23, -1.0)

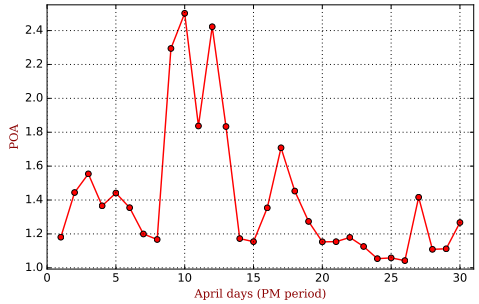


Fig. 4. POA of $\mathcal{I}_2$ for PM period in Apr. 2012 based on avg. flow on each link.

scaled partial derivative value. For instance, the entry $(0, 0.226)$ means

$$\frac{\partial V\left(\mathbf{t}^0,\mathbf{m}\right)}{\partial t_a^0} = 0.226 \times \max\left\{\frac{\partial V\left(\mathbf{t}^0,\mathbf{m}\right)}{\partial t_a^0};\, a = 0,\ldots,23\right\}$$

for link 0. It is seen that the largest five values of $\partial V\left(\mathbf{t}^0,\mathbf{m}\right)/\partial t_a^0$ correspond to links 19, 10, 8, 20, and 21 (red numbers on the left), and the largest five absolute values of $\partial V\left(\mathbf{t}^0,\mathbf{m}\right)/\partial m_a$ correspond to links 23, 20, 19, 21, and 10 (blue numbers on the right). This suggests that, around the PM peak period of January 10, 2012, the transportation management department could have most efficiently reduced the total users' travel time by taking actions with priorities on these links (e.g., improving road conditions to reduce the free-flow travel time for links 19, 10, 8, 20, and 21, and increasing the number of lanes to enlarge the flow capacity for links 23, 20, 19, 21, and 10).

ACKNOWLEDGEMENTS

The authors would like to thank the Boston Region MPO, and Scott Peterson in particular, for supplying the EMA data and providing us invaluable clarifications throughout our work.

REFERENCES

Bar-Gera, H. (2016). Transportation networks for research. *https://github.com/bstabler/TransportationNetworks*.

Bertsimas, D., Gupta, V., and Paschalidis, I.C. (2015). Data-driven estimation in equilibrium using inverse optimization. *Mathematical Programming*, 1–39.

Boyd, S. and Vandenberghe, L. (2004). *Convex optimization*. Cambridge university press.

Branston, D. (1976). Link capacity functions: A review. *Transportation Research*, 10(4), 223–236.

Dafermos, S.C. and Sparrow, F.T. (1969). The traffic assignment problem for a general network. *Journal of Research of the National Bureau of Standards B*, 73(2), 91–118.

Evgeniou, T., Pontil, M., and Poggio, T. (2000). Regularization networks and support vector machines. *Advances in computational mathematics*, 13(1), 1–50.

Hazelton, M.L. (2000). Estimation of origin–destination matrices from link flows on uncongested networks. *Transportation Research Part B: Methodological*, 34(7), 549–566.

Lundgren, J.T. and Peterson, A. (2008). A heuristic for the bilevel origin–destination-matrix estimation problem. *Transportation Research Part B: Methodological*, 42(4), 339–354.

Noriega, Y. and Florian, M.A. (2007). Algorithmic approaches for asymmetric multi-class network equilibrium problems with different class delay relationships. *https://www.cirrelt.ca/DocumentsTravail/CIRRELT-2007-30.pdf, CIRRELT*.

Patriksson, P. (2015). *The traffic assignment problem: models and methods*. Dover Books on Mathematics.

Pourazarm, S., Cassandras, C.G., and Wang, T. (2016). Optimal routing and charging of energy-limited vehicles in traffic networks. *International Journal of Robust and Nonlinear Control*, 26(6), 1325–1350.

Simon, C.P. and Blume, L. (1994). *Mathematics for economists*, volume 7. Norton New York.

Spiess, H. (1990). A gradient approach for the od matrix adjustment problem. *Centre de Recherche sur les Transports,* Universite de Montreal, (693).

Youn, H., Gastner, M.T., and Jeong, H. (2008). Price of anarchy in transportation networks: Efficiency and optimality control. *Physical review letters*, 101(12), 128701/1–128701/4.

Zhang, J. (2016). InverseVIsTraffic. *https://github.com/jingzbu/InverseVIsTraffic*.

Zhang, J., Pourazarm, S., Cassandras, C.G., and Paschalidis, I.C. (2016). The price of anarchy in transportation networks by estimating user cost functions from actual traffic data. In *IEEE 55th Annual Conference on Decision and Control (CDC),* to appear.

View publication stats