Block Simplex Signal Recovery: Methods, Trade-Offs, and an Application to Routing

Cathy Wu, *Student Member, IEEE*, Alexei Pozdnukhov, and Alexandre M. Bayen, *Member, IEEE*

***Abstract*— This paper presents the problem of block simplex constrained signal recovery, which has been demonstrated to be a suitable formulation for estimation problems in networks such as route flow estimation in traffic. There are several natural approaches to this problem: compressed sensing, Bayesian inference, and convex optimization. This paper presents new methods within each framework and assesses their respective abilities to reconstruct signals, with the particular emphasis on sparse recovery, ability to incorporate prior information, and scalability. We then apply these methods to route flow estimation in traffic networks of various sizes and network topologies. We find that both compressed sensing and Bayesian inference approaches are appropriate for structured recovery but have scalability limitations. The convex optimization approach does not directly incorporate prior information, but scales well and has been shown to achieve 90% route flow accuracy on a full-scale network of over 10 000 links and 280 000 routes on a synthetic benchmark based on the I-210 corridor near Los Angeles, CA, USA.**

***Index Terms*— Compressed sensing, Bayesian inference, block simplex, flow estimation, routing, signal reconstruction.**

I. Introduction

In this article, we examine a particular setting of signal recovery where the signal is constrained by a block simplex, and we refer to this problem as the *block simplex signal recovery problem*. There are many applications for this problem, e.g. computing equilibria for noncooperative games [1], [2] in game theory, structural support vector machines (SVMs) [3] in machine learning, and route flow estimation in transportation networks [4] (see Section III). The standard simplex is a natural way of encoding a discrete probability distribution, thus lending itself to recovery problems whose signal takes the form of multiple probability distributions, i.e. block probability distributions, such as multiple agents interacting and taking actions in a system. Prior work focuses on closely related but fundamentally different signal recovery structures, including ℓ^1-constrained signal recovery and signal recovery from a single probability simplex. In this article, we discuss the fundamental limitations of these approaches for block simplex signal recovery and propose a series of methods to address these limitations.

Rather than general objective functions, our work studies the special case of linear measurements, which is already a very challenging setting. In the setting of underdetermined linear measurements, it is already difficult to recover signals both accurately and efficiently. To overcome these challenges, we study the effects of placing assumptions on the signal (e.g. sparsity or belief priors) or making use of additional problem structure (e.g. constraints). Our aim is to study appropriate mathematical formulations and algorithms that 1) scale well to large problems, 2) recover sparse signals, and 3) encode our beliefs about the solution.

We investigate several approaches to this signal recovery problem–convex optimization, compressed sensing, and Bayesian inference–and we examine empirical recovery, to assess the effectiveness of these methods. We provide the first extensive view at the block simplex setting of signal recovery. We present and assess the following methods:

- Convex optimization (CO): A projected gradient method based on isotonic regression [4].
- Compressed sensing (CS): A new oracle and two new sampling-based methods for ℓ^∞-based regularization.
- Bayesian inference (BI): A new sampling approach based on Markov chain Monte Carlo (MCMC), in particular, Metropolis-Hastings.

Convex optimization is an important sub-field of optimization which takes advantage of convex analysis tools to provide efficient computational methods. However, this mathematical framework is limited and naturally may not encode crucial non-convex information, such as sparsity or probabilistic information [5], [6]. *Compressed sensing* is a widely-used signal processing technique for efficiently acquiring and reconstructing a sparse signal in underdetermined linear systems. However as discussed in Section V-A.2, the classical ℓ^1-based methods do not apply to our setting [7]–[9]. *Bayesian inference* is a statistical inference method which uses Bayes' theorem to update a probabilistic model given some evidence and provides a principled way to incorporate beliefs. However its sampling-based techniques for general probabilistic distributions typically do not scale well computationally [10], [11].

We provide a numerical evaluation of the methods on a practical application, where we assess both accuracy and

Manuscript received August 1, 2018; revised February 13, 2019; accepted March 31, 2019. The work of C. Wu was supported in part by the NSF Graduate Research Fellowship and in part by the Berkeley Chancellor's Fellowship. The Associate Editor for this paper was J. Miller.
(Corresponding author: Cathy Wu.)

C. Wu is with the Department of Electrical Engineering and Computer Science, UC Berkeley, Berkeley, CA 94720-1234 USA (e-mail: cathywu@eecs.berkeley.edu).

A. Pozdnukhov is with the Department of Civil, UC Berkeley, Berkeley, CA 94720-1234 USA.

A. M. Bayen is with the Department of Electrical Engineering and Computer Science, UC Berkeley, Berkeley, CA 94720-1234 USA, and also with the Institute for Transportation Studies (ITS), UC Berkeley, Berkeley, CA 94720-1234 USA.

Digital Object Identifier 10.1109/TITS.2019.2914174

scalability. We take our application to be route flow estimation in transportation networks, presented in Section III.

Our contributions are as follows:

- We introduce and formalize the problem of block simplex signal recovery.
- We derive and analyze a regularization scheme suitable for sparse recovery in the block simplex setting. We accordingly propose the first methods for *sparse* block simplex signal recovery. We present a novel oracle and two sampling-based compressed sensing methods.
- We derive a simple probabilistic graphical model from the block simplex structure, which encodes a known prior on the spread of the signal. This model is suitable for Bayesian inference methods such as Markov chain Monte Carlo (MCMC), and in particular, Metropolis-Hastings.
- We perform numerical experiments on large-scale matrices derived from a real-world application, rather than on matrices with known recovery properties (such as Gaussian random matrices).
- We demonstrate the scalability of the CO method. We empirically determine that CS and BI are suitable for recovery of small problems.

The remainder of the article is organized as follows: Section III describes our motivating application to state estimation in terms of route flow in road networks, and Section III-B presents its corresponding signal recovery problem. Section II describes related work. In Section IV, we present the problem. In Section V, we present three problem formulations and analysis, from the perspective of convex optimization, compressed sensing, and Bayesian inference. In Section VI, we present the respective methods for each formulation. Then, Section VII presents the experimental setup and numerical results. Section VIII concludes the article.

II. Related Work

A natural approach to incorporate domain-specific problem structure for signal recovery is to extend classical compressive sensing [7]–[9] by incorporating structured sparsity models, which may occur as domain-specific information, as done by [12]. Their work introduces and analyzes theoretical properties for model-based compressive sensing and explores several such structured sparsity models, namely a tree structure and block sparsity. Although the latter sounds similar to our concept of block sparsity, they focus on the case where it is equivalent to *joint sparsity*, where the supports of the blocks of the signals are shared between all of the blocks. Importantly, this specific structure avoids the curse of dimensionality in recovering the supports. Unfortunately, this model does not fit our application. For example, routes are different between different origin-destination pairs in a road network, and thus we naturally want to allow the support to be different for each block. Our problem setting thus does unfortunately suffer from the curse of dimensionality, thereby motivating the sampling-based methods studied in this article.

Our desired model differs in yet another way: our signal is block simplex *constrained*, implying that each block sums up to a known value. This means that classical ℓ_1-based compressive sensing theory does not apply directly to our problem (see Section V-A.2). Similarly, the work in [13] studies the problem of linearly constrained nonnegative least squares (NNLS) as a signal recovery problem, which is a more general form of our problem. Due to the specific structure of the block simplex constraint, however, their ℓ^1-based method is not suitable for our setting. Additionally, these methods do not address the issue of scalability. Recent work on sparse signal recovery in the *simplex*-constrained setting introduced a novel regularization scheme based on the ℓ^∞ norm for the single simplex [14]. This article builds upon [14] to introduce a regularization scheme for the block simplex setting.

In summary, we address the following limitations of previous work: First, traditional ℓ^1-based methods of sparse recovery through compressed sensing techniques do not apply to our problem due to the inherent ℓ^1 constraint in a (block) simplex constraint. Second, Due to the curse of dimensionality, signal recovery methods are typically not demonstrated to scale to large-scale problems.

III. Route Flow Estimation as Block Simplex Signal Recovery

The motivating application in our study is the problem of state estimation in traffic networks by examining the recovery of static route flow from a variety of sensor measurements in road networks. *Route flow* is the vehicles per time (e.g. hour) on a particular route, and a *route* is a sequence of links. The implications of accurate route flow estimation is vast, since it provides rich state information about the traffic network that cannot be recovered by commonly studied *link* and *origin-destination* (OD) flows. Route flow can subsequently be used for control on the network, for example re-routing affected vehicles in the event of a road closure. In fact, route flow may be used to compute the aforementioned types of flow, thereby providing backwards compatibility to the rich research built off of the disparate types of flow. These flow estimates are subsequently used for applications such as ramp metering, optimizing plans for signalized intersections, as well as long-term land use planning. Accurate route flow estimates are increasingly critical for a more effective use of existing traffic infrastructure as population density and the need for enhancing mobility in cities grow.

Route flow estimation is a challenging and inherently underdetermined problem because there are an exponential number of possible routes [15, §1.2], yet typically only a linear number of measurements. Thus, transportation science has historically aimed at modeling, computing, and estimating the movement of traffic in terms of the more computationally tractable link flows [16], turning flows, and origin-destination (OD) flows [17]–[20], which capture limited information about road networks. In this article, we thus focus on opportunities to improve route flow estimation, and we refer the reader to [21] for the rich history of state estimation for traffic networks.

A. Related Work in Route Flow Estimation

To overcome sensing limitations, numerous approaches to route flow have utilized modeling assumptions such as user

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

equilibrium (UE) [22], which permits the modeling of unique link flows and feasible route (or path) flows without requiring full route enumeration [23, §3.3], [24, §5.2]. The stochastic user equilibrium (SUE) (probit-based [25], [26] and logit-based [27], [28]) addresses some of the shortcomings of UE by modeling imperfect knowledge of the network and variation in drivers' preferences, which makes the estimation of route flows possible [24]. However, frequent perturbations in traffic networks indicate that real-world transportation networks may not be in equilibrium (or only approximately so) [29]. In this article we study methods which impose less restrictive assumptions on the route flow distribution, such as sparsity, probabilistic priors, or simply feasibility.

Due to advances and increasing adoption of sensing infrastructure, including cellular networks [4], [30], Bluetooth [31], [32], and license plate readers [33], [34], there is renewed interest in *route* flow estimation. Relatedly, the growing number of mobile sensors in urban areas has enabled the use of probe vehicles for route inference from GPS traces [35], [36]. For route flow estimation in particular, [34] incorporates timing and correlated location measurements from license plate readers into a SUE framework, achieving improved link and route flow. A study by [30] on route choice estimation proposes a Euclidean projection of cellular network base station traces to its nearest route in a traffic network. Another study uses aggregated cellular base station measurements in a convex optimization framework, without any UE assumptions, further demonstrating improved link and route flow accuracy [4]. Instead of the simple k-shortest paths considered in [4], a recent work [37] studies richer set of routes to consider and their impact on the resulting link flow estimates. This article builds upon [4] by proposing several new estimation methods based on powerful mathematical frameworks, for route flow, which also permit structured priors on the route distribution.

In traffic networks, applications of compressed sensing have been found in optimal sensor placement [38], and applications of Bayesian inference have been found in OD estimation [39], [40] and day-level dynamic link flow estimation [41]. This article studies compressed sensing and Bayesian inference in the context of block simplex signal recovery and its application to route flow estimation in traffic networks.

B. Problem Definition

As our working example, we now present the problem of route flow estimation as a block simplex signal recovery problem. In Sections IV and V, we will present the general problem and three formulations for the problem, from the perspectives of compressed sensing, Bayesian inference, and convex optimization. In Section VI, we will present the corresponding solution methods.

In addition to traditional traffic sensors which measure flow on individual links in the network, we consider aggregated measurements from cellular traces, sequences of cell towers that cell phones connect to over time, which provide a rich data source. We call these measurements *cellpath flow*, that is, the vehicles per time which connect to a particular *cellpath*; a

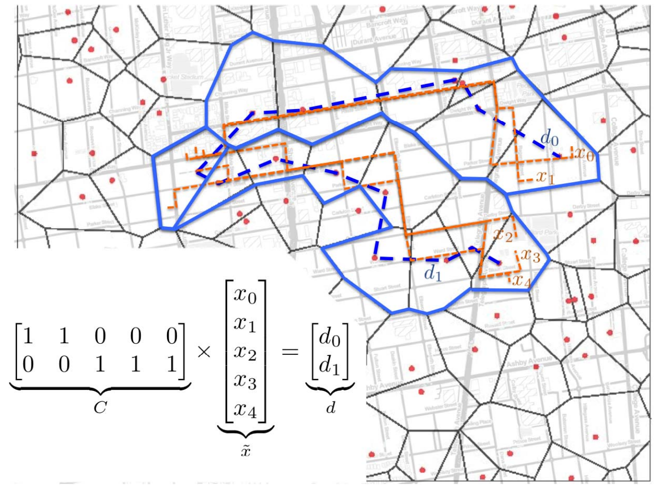


Fig. 1. In the routing application, scattered sensor nodes (e.g. cell towers, OD "sensors") measure the signal strength of the transitory agents in the system and relay that information to a central datastore. The aim is to recover the number of agents traversing any path in the region. Each unique combination of sensor regions (e.g. cells, origins) begets a single simplex structure, as each path passes through exactly one sensor sequence (e.g. cellpath, OD) and yet multiple paths can pass through the same one. Thus, the overall signal recovery problem yields a block simplex structure (denoted by the ones in the C matrix, along with a nonnegativity constraint, i.e. $x_i \geq 0, \forall i$), one simplex for each combination of sensor regions. Depending on the setting, the practitioner may wish to extract a sparse solution, a solution that can encode complex prior information, or just a feasible solution. This article studies methods which permit each type of desired solution.

cellpath is a sequence of cell towers. After we form a partition of the network based on cellular tower coverage, we observe that we may formulate the relationship between route flow and cellpath flow as a block simplex, one block for each cellpath. This captures the constraint on the multiple routes that may pass through the same sequence of cell regions. For each cellpath measurement, we have a (scaled) simplex constraint on the route flow measurements for those routes, hence motivating the study of the block simplex signal recovery problem. Similarly, we may consider a simplex constraint for each set of routes corresponding to each OD, sequence of license plate readers, sequence of Bluetooth readers, aggregated GPS traces, or measurements from other similar technologies. We refer to this class of traffic flow measurements with block simplex structure as *block simplex flow measurements*.

We wish to recover the route flow x, which has a block simplex structure due to the availability of the aforementioned block simplex flow measurements, which are encoded by (C, d). The matrix C is an incidence matrix with exactly one 1 per column to form simplex blocks, and d is a vector denoting the scaling of each simplex measurement. An equality constraint $Cx = d$, jointly with a nonnegativity constraint $x \geq 0$, forms a block simplex constraint. We use (A, b) to encode additional linear sensor measurements, such as link flows, where similarly A encodes an incidence matrix (without any particular simplex structure) and b encodes the measurement value. An illustrative example is provided in Figure 1. Thus, this gives us the following optimization problem

$$\min_{x} \ \frac{1}{2}\|Ax - b_2^2 + \lambda\|x_2^2 \qquad \text{s.t. } Cx = d, \quad x \succeq 0 \tag{1}$$

In Section VII, we will evaluate and compare the methods in Section VI on this route flow estimation problem.

IV. Block Simplex Signal Recovery

In this section, we pose the general *block simplex signal recovery problem*, which is a constrained signal recovery problem. We first introduce the general signal recovery problem.

A. General Signal Recovery

Let $x \in \mathbb{R}^n$ be a signal with at most k non-zero components. This type of signal is called *k-sparse*. Let $\{a_1, \ldots, a_m\}$ be a sequence of *measurement vectors* in $\mathbb{R}^n$ that are independent of the signal. We denote the *measurement matrix* to be $A = \begin{bmatrix} a_1 \cdots a_m \end{bmatrix}^T$. The inner products $a_1^T x, \ldots, a_m^T x$ produce m linear measurements of the signal x, denoted by a vector $b = Ax \in \mathbb{R}^m$, also called the *data vector*. In the problem of *signal recovery*, two distinct questions are asked [42]:

1) How many measurements m are needed for reconstruction?
2) Given these measurements (A, b), what algorithms can perform the reconstruction task?

B. Block Simplex Signal Recovery

Following the previous section, with $x \in \mathbb{R}^n$ a k-sparse signal, $A \in \mathbb{R}^{m\times n}$ the measurement matrix independent of the signal x, and $b = Ax \in \mathbb{R}^m$ the data vector, we now suppose that the signal x is also constrained to be in a block simplex, a commonly occurring linear equality constraint [13], often corresponding to block probability distributions that arise in game theory, machine learning, and flow networks.

A (single-block) p-simplex constraint on a $\mathbb{R}^p$-valued signal s is commonly written as

$$s \in \Delta_+^p \stackrel{\Delta}{=} \{s \in \mathbb{R}^p : s_i \geq 0 \quad \forall i,\ \mathbf{1}^T s = 1\}$$

More generally, a q-block $\mathbf{p}$-simplex constraint on x of dimensions $\mathbf{p} = \{p_1, \cdots, p_q\}$ (where $n = \sum_{i=1}^q p_i$) is written

$$x \in \Delta_+^{\mathbf{p}} \stackrel{\Delta}{=} \Delta_+^{p_1} \times \cdots \times \Delta_+^{p_q}$$

In this problem, $A, b, \mathbf{p}$ are known and given, and x is unknown. Similarly to before, in the block simplex signal recovery problem, we wish to reconstruct the signal x in this constrained setting.

Note that if the sparsity is not known (or if the problem is not sparse), we indicate that $k = n$.

In this article, we focus on the following questions

- Given these measurements $(A, , \mathbf{p})$, what algorithms can perform the reconstruction task?
- What algorithms can perform the reconstruction task on large problems?

The question of the number of measurements m necessary for reconstruction is also of fundamental importance and is deferred to future work.

V. Formulations of the Problems of Interest

In this section, we present three problem formulations along with a computational method for each. We examine approaches from convex optimization, Bayesian inference, and compressed sensing, each of which is a principled approach to the block simplex signal recovery problem. Each of the respective methods have trade-offs in terms of computational efficiency, and assumptions on the signal.

A. Compressed Sensing

In this section, we present the compressed sensing formulation for the block simplex signal recovery problem. To simplify the notation, we present the noiseless case, i.e. $Ax = b$ in this section. A least squares loss may be added back into the objective without changing any of our subsequent results.

The classic k-sparse formulation of the block simplex signal recovery problem is

$$\min_{x\in\mathbb{S}} \quad \|x\|_0 \tag{2}$$

where $\mathbb{S} = \Delta_+^{\mathbf{p}} \cap \{x : Ax = b\}$. This finds the minimum cardinality signal satisfying the measurements and the block simplex constraint.

We first motivate sparsity in the context of the route flow estimation application. We then describe two relaxation approaches in the context of a single simplex, first to provide an explanation of why classical compressed sensing approaches do not apply in the block simplex setting, and second to describe the approach we will extend to the block simplex setting.

1) Sparsity in Route Flow Estimation: Although a road network permits a combinatorial number of possible routes, it is likely that only a tiny fraction of them are used by people. For instance, routes through highways and major streets are much more likely to be frequented than a random sequence of small streets. Hence, we may wish to recover a route flow signal is sparse.

2) Single Simplex Convex Relaxation via ℓ^1: Equation (2) is well-known to be NP-hard due to its combinatorial nature. Therefore, in the past decade there has been a line of work in compressed sensing which discusses the ℓ^1 norm convex relaxation (see Equation (3)) [5].

In our problem the relaxation would read

$$\min_{x\in\mathbb{S}} \quad \|x\|_1 \tag{3}$$

We note that in our case of a simplex constraint, the ℓ^1 norm value is already known. Unless the k-sparse signal x^* is the unique solution of Equation (3), we thus need a different regularization scheme to achieve strong recovery results. In the literature, this problem is referred to as recovering sparse probability distributions [14].

Without loss of generality, consider for a moment the single-block standard simplex case, that is $C = 1^T, d = 1$ with $C \in \mathbb{R}^{1\times n}$ and let x be k-sparse. For a single standard simplex, we observe that $1 = \|x\|_1 \leq \|x\|_0 \cdot \|x\|_\infty$, thus $\|x\|_\infty^{-1}$ is a lower bound on the size of the support of x. This suggests trying to use the inverse ℓ^∞ norm as a proxy for sparsity.

Let o^* denote the objective value at the solution to Equation (2), the minimum value of $\|x\|_0$ subject to the constraints, we then consider the relaxation

$$o^* \geq o^*_\infty = \min_{x \in \mathbb{S}} \|x\|^{-1}_\infty \quad \text{and}$$
$$1/o^*_\infty = \max_{i=1,\ldots,n} \max_{x \in \mathbb{S}} x_i. \tag{4}$$

which is solvable with n linear programs (LP), giving an overall complexity of at most $\mathrm{O}n^4$ with a primal-dual LP solver.

Instead of taking the maximum over $1 \leq i \leq n$ in the linear program formulation of Equation (4), we can equivalently take the solution with the lowest cardinality, thus solving

$$\hat{x}_i = \arg\max_{x \in \mathbb{S}} x_i \quad \text{for} \quad i \in \{1, \cdots, n\} \tag{5}$$
$$\hat{x}_{\min} = \arg\min_x \|x\|_0 : x \in \{\hat{x}_1, \cdots, \hat{x}_n\}$$

which has also been proposed in [14] to recover the k-sparse signal. This alternative minimization scheme has the same complexity of $\mathrm{O}n^4$. We call this inverse-ℓ^∞ regularization.

3) Block Inverse-ℓ^∞ Regularization: Instead of a single standard simplex $x \in \mathbb{R}^n$, we consider p standard simplices, following the standard simplex form given in (13). Recall that $\mathbf{p} \in \mathbb{R}^p$ is the vector of individual simplex dimensions and let $\mathbf{k} \in \mathbb{R}^p$ be the vector of nonzero entries for each respective block of x.

As before, we need a convex relaxation for this problem to make reconstruction tractable. We could ignore the block structure and directly apply Equation (3) by replacing the objective with $p\|x\|^{-1}_\infty$. However the observation

$$\|x\|_0 = \sum_i \|x^i\|_0 \geq \sum_{i=1}^q \|x^i\|^{-1}_\infty \geq q\|x\|^{-1}_\infty$$

suggests that a tighter approximation can be achieved by solving

$$o^* \geq o^*_\infty = \min_{x \in \mathbb{S}} \textstyle\sum_{i=1}^q \|x^i\|^{-1}_\infty$$
$$o^*_\infty = \min_{\substack{j_i=1,\ldots,p_i \\ \forall i=1,\ldots,q}} \min_{x \in \mathbb{S}} \textstyle\sum_{i=1}^q (x^i_{j_i})^{-1} \tag{6}$$

Two new computational challenges arise from this approximation. First, a direct extension of the method presented in [14] requires computing combinatorially many ($\prod_{i=1}^q p_i$) subproblems in order to compute the minimum, which is computationally intractable (see Section V-A.4 for more details). Second, the subproblems are no longer linear programs, but rather constrained non-linear convex programs. We address the first and more troublesome problem by supposing that with some additional knowledge about x, we can make the following definition and observation, from which we may derive efficient solution methods.

Definition 1 (Maximal Support): The maximal support i of a vector $x \in \mathbb{R}^n$ is given as

$$i = \arg\max_{i \in \{1,\cdots,n\}} x$$

We denote the block-wise maximal support

$$\mathrm{maxsupp}(x) = (\arg\max_i x^1, \cdots, \arg\max_i x^q) \in Q \tag{7}$$

Definition 2 (Maximal Support Optimality): Let $x^* \in X^*$ be an optimal solution to Equation (6), where X^* denotes the set of optimal solutions, and let $\mathbf{j}^* = (j_1, \cdots, j_q)$ denote some subproblem. Then, $\mathbf{j}^* \in \{\mathrm{maxsupp}(x) : x \in X^*\}$ (i.e. in the set of optimal block-wise maximal support) if and only if $\mathbf{j}^*$ achieves

$$o^*_\infty = \min_{x \in \mathbb{S}} \sum_{i=1}^q (x^i_{j^*_i})^{-1} \tag{8}$$

Proof: ($\rightarrow$): Let x^* denote an optimal solution to Equation (6) and let $\mathbf{j}^* = \mathrm{maxsupp}(x^*)$. Then

$$o^*_\infty = \min_{\mathbf{j} \in Q} \sum_{i=1}^q \frac{1}{(x^*)^i_{j_i}} \leq \sum_{i=1}^q \frac{1}{(x^*)^i_{j^*_i}}$$

Now consider some $\mathbf{j} \neq \mathbf{j}^*$. Without loss of generality consider the case where a single index is moved from the maximal support, then there is some j_i where $x^*_{j_i} \leq x^*_{j^*_i}$. This leads to an increase in the objective. Then for arbitrary $\mathbf{j}$, we have an upper bound (which will turn out to be tight)

$$o^*_\infty \leq \sum_{i=1}^q \frac{1}{(x^*)^i_{j^*_i}} \leq \sum_{i=1}^q \frac{1}{(x^*)^i_{j_i}} \quad \forall\, \mathbf{j} \in Q$$
$$\leq \min_{\mathbf{j} \in Q} \sum_{i=1}^q \frac{1}{(x^*)^i_{j_i}} = o^*_\infty$$

from which we conclude that $\mathbf{j}^*$ attains the minimum o^*_∞.

($\leftarrow$): Now consider $\mathbf{j} \notin \{\mathrm{maxsupp}(x) : x \in X^*\}$. Then for any $x^* \in X^*$, clearly

$$o^*_\infty < \sum_{i=1}^q \frac{1}{(x^*)^i_{j_i}}$$

so the minimum o^*_∞ is not attained. By definition of optimal solution, $x \notin X^*$ trivially does not attain o^*_∞. ■

Corollary 1 (Optimality From Maximal Support): If the maximal support of x^* is known (or can be guessed), we may compute the optimal solution to Equation (6) by solving a single convex program instead of combinatorially many.

This can thereby drastically reduce the complexity of the solution method for the problem in Equation (6). For more details on these methods, see Section VI-A.

We refer the reader to [14, §3.1] for sufficient conditions on A for sparse recovery in the single block inverse-ℓ^∞ case. As in compressed sensing with ℓ^1 regularization, these conditions are difficult to check.

4) Block Inverse-ℓ^∞ via Direct Extension: An alternative approach is a direct extension of the method presented in [14] and is analogous to Equation (5). The approach is to solve the following program, which selects an index from each simplex block $\mathbf{q} = (p_0, \cdots, p_q)$. We denote the possible block-wise indices $\mathcal{P} = \{\mathbf{p}' \in \mathbb{Z}^q : 0 \preceq \mathbf{p}' \preceq \mathbf{p}\}$.

$$\hat{x}_{\mathbf{q}} = \arg\min_{x \in \mathbb{S}} \sum_{i=1}^{p} (x^i_{q_i})^{-1} \quad \text{for} \quad \mathbf{q} \in Q \tag{9}$$

$$\hat{x}_{\min} = \arg\min_{x} \|x\|_0 : x \in \{\hat{x}_{\mathbf{q}} : \mathbf{q} \in Q\}$$

This approach requires, in the worst case, computing combinatorially many ($\prod_{i=1}^{q} p_i$) subproblems in order to achieve the minimum, which is computationally intractable.

B. Bayesian Inference

We derive a probabilistic graphical model based on linear measurements and the block simplex constraint. We first note that the Dirichlet distribution encodes a standard simplex with a prior on the spread.

1) Spread in Route Flow Estimation: It may be desirable to tune the spread of the recovered signal, between sparse and uniform. In terms of route flow, that means that we may have some prior on how much people spread themselves over all possible routes; people may all choose a few specific routes between an origin-destination pair, or people may spread themselves evenly over more routes. We may encode such information into a probabilistic model as a prior.

2) Probabilistic Model: We propose to compute the posterior distribution for the following probabilistic model (Figure 2), which has a single (scalar) fixed concentration hyperparameter α:

$$x^i|\alpha \;=\; (x^i_1, \cdots, x^i_{p_i}) \overset{ind}{\sim} \mathrm{Dir}(\alpha \mathbf{1}^i) \quad \forall\, i \in [1, 2, \ldots, q] \tag{10}$$

$$y|x \overset{ind}{\sim} \mathcal{N}(b, \sigma^2 I) \tag{11}$$

$$\text{where } b = \tilde{A}x \qquad \text{and} \qquad x = (x^1, \cdots, x^q)$$

The $\mathbf{1}^i$ denotes an all-ones vector of size p_i, which is the ith entry of the simplex dimensionality vector $\mathbf{p} \in \mathbb{N}^q$. The signal x is comprised of q blocks with sizes $\mathbf{p}$, and subsets of the signal x are drawn from a Dirichlet prior in Equation (10), where x^i denotes the ith simplex block. $\mathrm{Dir}(\vec{\alpha})$ denotes the Dirichlet distribution with $\vec{\alpha} \in \mathbb{R}^{|\vec{\alpha}|}$ concentration parameter.

The data vector b in Equation (11) is encoded as the mean for m normally distributed random variables, which have a dependence on the x vector. The measurement matrix A encodes precisely the linear dependence of b on x and it represents the edges in the graphical model (see Figure 2). That is, in this problem, the dependence function is a weighted sum of random variables given by linear equations.

For simplicity we assume Gaussian noise in the data vector b given by variance σ^2, and we assume no noise in the OD flow measurements. That is, y is stochastic, but A is fixed.

3) Posterior Distribution: The posterior probability is proportional to the product of the likelihood and prior probability (Bayes theorem) and may be written as:

$$\begin{aligned} P(x|y,\alpha) \propto P(y,x|\alpha) &= P(y|x)P(x|\alpha) \\ &= \prod_{j=1}^{m} P(y_j|x) \prod_{i=1}^{p} P(x^i|\alpha) \end{aligned} \tag{12}$$

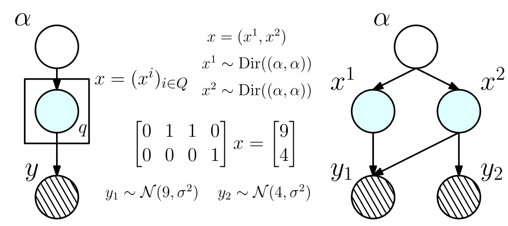


Fig. 2. Graphical model for the block simplex recovery problem given by Equations 10-11. The left shows the general case, and the right shows an example, given by the setting in the center. The model consists of fixed hyper-parameter α, observed measurements y (of the data vector b), and latent variables x (the signal). The precise relationship between x and y is given by a measurement matrix A, as demonstrated by the equation in the center, $Ax = b$.

by a certain abuse of notation in using α to denote a vector of varying sizes, repeating the single hyperparameter as in Equation (10).

C. Block Simplex Constrained Least Squares (BSLS)

Finally, we present the block simplex signal recovery problem (Section IV-B) directly as a convex optimization problem. We call this the block simplex constrained least squares (BSLS) problem. In this framework, we do not consider external information such as our beliefs about the sparsity or the spread of the signal. The problem may be written as the minimization of a quadratic program:

$$\begin{aligned} &\min_{x} \; \frac{1}{2}\|Ax - b\|_2^2 + \lambda \|x\|_2^2 \\ &\text{s.t. } Cx = 1, \quad x \succeq 0 \end{aligned} \tag{13}$$

which minimizes the sum-squared measurement residual subject to the constraints and some regularization parameter λ. Recall that $x \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, and $b \in \mathbb{R}^m$. We re-write the block simplex constraint in matrix form

$$x \in \Delta_+^{\mathbf{p}} \overset{\Delta}{=} \Delta_+^{p_1} \times \cdots \times \Delta_+^{p_q} \iff Cx = 1, \quad x \succeq 0 \tag{14}$$

where $1 \in \mathbb{R}^q$ and the simplex constraint matrix $C \in \mathbb{R}^{q \times n}$ takes the following block diagonal form with all-ones vectors as the component blocks

$$C = \begin{bmatrix} \mathbf{1}_{p_1}^T & & (0) \\ & \mathbf{1}_{p_2}^T & \\ & & \ddots & \\ (0) & & & \mathbf{1}_{p_q} \end{bmatrix} \quad \text{and} \quad \mathbf{1}_{p_i} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \in \mathbb{R}^{p_i}.$$

Note that the ordering of the signal x must respect the ordering of the blocks.

In general, $m \ll n$ (underdetermined) and $p \le n$ (non-degenerate simplex blocks), thus typically the Hessian $A^T A$ of our convex quadratic objective is singular ($A^T A \in \mathbb{R}^{n \times n}$ but $\mathrm{rank}\,(A^T A) \le m \ll n$). Thus without regularization, i.e. $\lambda = 0$, the problem might have multiple optimal solutions (underdetermined) or might have more observations than unknowns (overdetermined). Moreover, when there are uncorrelated measurement errors on the data vector b, the ordinary least squares is the best unbiased estimator of x.

1) Route Flow Estimation as BSLS: We have already posed the route flow estimation problem as a BSLS problem in Section III-B. This formulation is appropriate in settings where we do not have strong prior information on the structure of the signal, i.e. if some routes are preferred over others.

2) Scaled Simplex Form: Though we pose the problem in terms of the standard simplex, which is important for the analysis of Equation (13) and interpretation (see Equation (10)), the problem generalizes directly to the case of scaled simplicies. This can be shown through a straight-forward change of variables to yield the desired scaled block simplex constraint $C\tilde{x} = d$, where $\tilde{x}$ is a block-wise re-scaled version of x and $d \in \mathbb{R}^q_+$.

Proposition 1 (Equivalence to Scaled Simplex): Equation (13) can be reduced from a least-squares problem with (separable) scaled simplex constraints:

$$\begin{aligned} &\min_x \ \frac{1}{2}\|\tilde{A}\tilde{x} - b\|_2^2 + \lambda\|E^{-1}\tilde{x}\|_2^2 \\ &\text{s.t. } C\tilde{x} = d, \quad \tilde{x} \succeq 0 \end{aligned} \tag{15}$$

where $d \in \mathbb{R}^q_+$, $E = \operatorname{diag}(C^T d)$, $\tilde{A} = A^{-1}E$, and $x = E^{-1}\tilde{x}$. The vector d denotes the scaling for each simplex.

Proof: Take the formulation in Equation (15). Let $D = \operatorname{diag}(d)$ and $E = \operatorname{diag}(C^T d)$. First observe

$$\mathbf{1} = D^{-1}d = D^{-1}C\tilde{x} = D^{-1}CEE^{-1}\tilde{x}$$

where $\mathbf{1} \in \mathbb{R}^p$ denotes the ones vector. We define $x = E^{-1}\tilde{x}$ and $A = \tilde{A}E$, and notice that $Ax = \tilde{A}\tilde{x}$.

We may assume without loss of generality that $d \succ 0$, since $d_i = 0$ for some block $i \in \{1, \cdots, p\}$ implies the trivial case where $\tilde{x}^i = 0$ (non-negativity). The block $\tilde{x}^i$ does not contribute to the objective, and so we may write an equivalent problem to Equation (15) where $d \succ 0$ by removing all such trivial blocks $\tilde{x}^i$. Then since $d \succ 0$, the non-negativity constraints $\tilde{x} \succeq 0$ and $x \succeq 0$ are equivalent. We may also then assume without loss of generality that D, E are non-singular.

Finally, we observe that $D^{-1}CE = C$ due to the block structure of C, yielding $Cx = \mathbf{1}$. This can be seen by noticing that E is a *right* operator that scales the rows of C by the entries in d, and D is a *left* operator that scales the rows of C by the entries in d, i.e. $CE = DC$, thus performing both operations results in the original matrix C.

Then we have the original standard simplex formulation of the BSLS problem given by Equation (13). Provided that the problem is well-posed (E is non-singular), all steps are reversible, and thus Problems (13) and (15) are equivalent. ■

Thus, without loss of generality, we refer to A, rather than $\tilde{A}$, as the measurement matrix. We refer to $\tilde{A}$ as the scaled measurement matrix.

VI. METHODS

We present briefly the state of the art or present our own methods for each problem formulation of the block simplex signal recovery problem. The first two presented methods are relaxations, the third is algorithmic, and the last two methods are sampling-based.

A. Compressed Sensing (CS)

The objective in Equation (6) can be written as optimizing over $\mathbb{S} = \Delta_+^{\mathbf{p}} \cap \{x : Ax = b\}$ for

$$o_\infty^* = \min_{\mathbf{j} \in Q} \min_{x \in \mathcal{S}} \sum_{i=1}^q (x^i_{j_i})^{-1}$$

where $Q = \{\mathbf{q} \in \mathbb{Z}^p : 0 \preceq \mathbf{q} \preceq \mathbf{p}\}$. The minimum is now taken over combinatorially many (convex) subproblems, which is computationally intractable. By Property 1, we have that if we know $\mathbf{j} = \operatorname{maxsupp}(x^*)$ for some $x^* \in X^*$, then we can compute x^* by solving the single subprogram in Equation (8)

$$x^* = \arg\min_x \sum_{i=1}^q \frac{1}{x^i_{j_i^*}} \quad \text{such that } Ax = b, \quad Cx = 1, \quad x \succeq 0 \tag{16}$$

Thus we propose the following oracle method, which incorporates $\operatorname{maxsupp}(x^*)$ directly.

Algorithm 1 `Oracle(·)` Oracle Method for Block Sparse Recovery

Require: $\mathbf{j} = \operatorname{maxsupp}(x^*) \in Q$ for some $x^* \in X^*$

1: Compute $\hat{x} = \arg\min_{x \in \mathcal{S}} \sum_{i=1}^q (x^i_{j_i})^{-1}$

2: **return** $\hat{x}$

Since $\operatorname{maxsupp}(x^*)$ is not known in general, x^*, we propose the following random sampling method to find $\operatorname{maxsupp}(x^*)$: First, compute a rough pilot estimate $\hat{x}$ for x^* (or the maximal support of x^*), e.g. by minimizing the ℓ^1 norm or the elastic net over all x satisfying $Ax = b$ (that is, ignoring the block simplex constraints). Basic results in compressed sensing guarantee that the true support can be recovered with high probability [43] in the single block case. Then, we treat the estimate as the prior for selecting the maximal support for each block, and we update our prior as we find better estimates of x^*. The random sampling method is detailed in Algorithm 2.

Algorithm 2 `Random-Sampling(·)` Random Sampling Method for Block Sparse Recovery

---

Require: initial point (prior) $\hat{x} \in \mathbb{R}^n$, need not be feasible

1: **while** max iterations not reached **do**

2: Pick some index for each simplex block, $j_i \sim \text{Categorical}(p_i, \hat{x}^i)$ for $i \in \{1, \cdots, q\}$

3: For the selected $\mathbf{j} = (j_1, j_2, \cdots, j_q)$, compute $p_\infty = \min_{x \in \mathcal{S}} \sum_{i=1}^q (x^i_{j_i})^{-1}$

4: If p_∞ is the best so far, then update $\hat{x} = \arg\min_{x \in \mathcal{S}} \sum_{i=1}^q (x^i_{j_i})^{-1}$

5: **end while**

We additionally propose the following random sampling based heuristic (Algorithm 3), directly extending the method proposed in [14], which similarly has the benefit of solving linear subprograms instead of nonlinear subprograms.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Algorithm 3 `Random-Sampling-LP(·)` Random Sampling LP Method for Block Sparse Recovery

Require: initial point $\hat{x} \in \mathbb{R}^n$, need not be feasible
while max iterations not reached **do**
  Pick some index for each simplex block, $q_i \sim$ Categorical$(p_i, \hat{x}^i)$ for $i \in \{1, \cdots, p\}$
  For the selected $\mathbf{q} = (q_1, q_2, \cdots, q_p)$, compute $p_\infty = \max_{x \in \mathcal{S}} \sum_{i=1}^{p} x^i_{q_i}$
  If p_∞ is the best so far, then update $\hat{x} = \arg\max_{x \in \mathcal{S}} \sum_{i=1}^{p} x^i_{q_i}$
end while
return $\hat{x}$

B. Bayesian Inference (BI)

1) Markov Chain Monte Carlo (MCMC) Method: The normal-Dirichlet posterior probability in Equation (12) does not have a known analytical form, unfortunately, as in the case of conjugate priors. As it requires the integration of complex high-dimensional functions, it is difficult to derive an efficient Expectation-Maximization (EM) method or Gibbs sampler. Instead, we take an MCMC approach [44], which attempts to simulate direct draws from complex distributions. We apply a modified Metropolis-Hastings sampling method with normal proposal distributions.

2) Metropolis-Hastings: Metropolis-Hastings is an MCMC sampling method that is well suited for drawing samples from some distribution $p(\theta) = f(\theta)/Z$, where normalizing constant Z may not be known and may be very difficult to compute.

Our problem is well suited for a Metropolis-Hastings approach [45], as we wish to sample from the posterior distribution

$$P(\tilde{x}|y, \alpha) = \frac{P(y|\tilde{x})P(\tilde{x}|\alpha)}{Z}$$

where the numerator is known (analytically) but the denominator is not.

We use an implementation of the Metropolis-Hastings sampling method, with two common augmentations:

1) An additional periodic tuning of a scaling parameter for the proposal distribution, and
2) A sequential block-updates of the latent variable. The second augmentation to the standard Metropolis-Hastings method is updating each $\tilde{x}^i$ in sequence. That is, each simplex block $\tilde{x}^i$ is drawn in sequence, conditioned on the rest of the blocks, denoted $\tilde{x}^{-i}$, and accepted (or rejected) in sequence.

The full procedure is detailed in Algorithm 4.

Thus, we have the following p proposal distributions to the Metropolis-Hastings method for each latent variable $\tilde{x}^i, i \in [1, 2, \cdots, p]$. For simplicity, we take our proposal distribution to be the normal distribution.

$$q_k(\tilde{x}^i|\tilde{x}^{-i}) = \tilde{x}^{-i} + \epsilon_i$$

Algorithm 4 `MH(f,q,N)` Metropolis-Hastings Sampling With Block Updates and Periodic Tuning

Require: Any initial point $\theta^{(0)}$ satisfying $f(\theta^{(0)}) > 0$
Require: Scaling parameter $\beta = 1$, tuning interval T
Require: Proposal distributions $q_i(\theta|\theta^{(t-1)}),\ i \in \{1, \ldots, K\}$
for $t = 1, \ldots, N$ **do**
  for $k = 1, \ldots, K$ **do**
    Sample $\theta_k \sim \beta q(\theta_k|\theta_k^{(t-1)})$
    Acceptance probability $\alpha = \min(\frac{f(\theta_k)q(\theta_k|\theta_k^{(t-1)})}{f(\theta_k^{(t-1)})q(\theta_k^{(t-1)}|\theta_k)})$
    Accept with probability α: $\theta_k^{(t)} = \theta_k$
    Else: $\theta_k^{(t)} = \theta_k^{(t-1)}$
  end for
  Every T iterations, tune scaling parameter β according to the acceptance rate over last interval
end for
return $\{\theta^{(1)}, \ldots, \theta^{(N)}\}$

where $\epsilon^i = (\epsilon^i_1, \epsilon^i_2, \cdots, \epsilon^i_{p_i}) \overset{iid}{\sim} \mathcal{N}(0, \sigma)$, and the complete likelihood is given by $f : \mathbb{R}^n \to \mathbb{R}$,

$$f(\tilde{x}|y, \alpha) = \prod_{j=1}^{m} P(y_j|\tilde{x}) \prod_{i=1}^{p} P(\tilde{x}^i|\alpha)$$

C. Convex Optimization (CO)

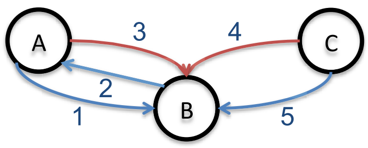
1) BSLS Problem: To solve the BSLS problem, we apply an efficient projected first-order descent method (see Algorithm 5), with a block isotonic regression projection step (which adapts the well-studied Pool Adjacent Violators Algorithm (PAVA) [46], [47] to the box-constrained setting, see Algorithm 6). For a full treatment of PAVA, see [4, §3] and [48].

Algorithm 5 `Proj-Descent(·)` General Projected Descent Method

Require: initial point $z = (z^p)_{p \in \mathcal{P}}$ in the feasible set $\mathcal{X}$.
while stopping criteria not met **do**
  Determine a descent direction $\Delta z = (\Delta z^p)_{p \in \mathcal{P}}$
  Projection: $(z^p)^+ := \underset{u^p}{\operatorname{argmin}} \{\|z^p + \Delta z^p - u^p\|^2 \ :\ 0 \le u^p_1 \le u^p_2 \le, \cdots \le u^p_{n_p-1} \le 1\},\ \forall\, p \in \mathcal{P}$
  Line search on the projected arc: $\gamma :\approx \operatorname{argmin}\{f(z + t(z^+ - z)) \ :\ t \in [0, 1]\}$
  $z := z + \gamma(z^+ - z)$
end while
return z

A key component of our approach is the analysis of the constraints of the convex optimization program. We apply a standard equality constraint elimination technique [5, §4.2.4] with a particular change of variable which converts the non-negativity constraints on the variables to ordering constraints. In the new space induced by the change of variables, we show that the projection on the feasible set (characterized by the ordering constraints) can be performed in linear time

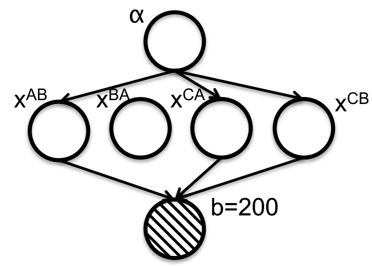
This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.



(a)

OD flow	OD	Route
400	AB	1
	AB	3
500	BA	2
600	CA	4,2
	CA	5,2
1000	CB	4
	CB	5

(b)



(c)

Fig. 3. Benchmark network example. Figure 3a depicts a directed network representation, with 3 origins (nodes, denoted by letters), 5 links (edges, denoted by numbers), and 7 routes (denoted by their sequence of links). Indicated in are 2 links that share a sensor measurement $b = 200$. Figure 3b shows the OD flow measurements between all OD pairs, along with the set of routes (sequence of links) we consider for the toy problem. We denote again in the routes that share sensor measurement b. Figure 3c shows the graphical model corresponding to the network, available measurements, and latent variables. Note that b does not depend on x^{BA} or the flow between BA. (a) Benchmark network. (b) OD flows and routes. (c) OD flows and routes.

Algorithm 6 `PAVA-proj`(y^p) Projection Onto Ordering Constraints in Line 3 of Algorithm 5

Require: vector $y^p \in \mathbb{R}^{n_p-1}$

1: compute with the PAV algorithm [46]:
$$y^{p,\text{iso}} := \underset{u^p}{\text{argmin}}\{\|u^p - y^p\|_2^2 : u_1^p \le u_2^p \le \cdots \le u_{n_p-1}^p\}$$
2: project $y^{p,\text{iso}}$ onto $[0,1]^{n_p-1}$:
$$\tilde{y}_k^p = \begin{cases} y_k^{p,\text{iso}} & \text{if } y_k^{p,\text{iso}} \in [0,1] \\ 0 & \text{if } y_k^{p,\text{iso}} \le 0 \\ 1 & \text{if } y_k^{p,\text{iso}} \ge 1 \end{cases}.$$
3: **return** return $\tilde{y}^p$

via bounded isotonic regression (see [49] for a short survey on isotonic regression). Then we solve our convex optimization program with an accelerated first order or second order projected descent algorithm. The change of variables presents two main advantages: the dimensionality is reduced (for each block), which is critical for large-scale problems, and we can perform the projection in On, an improvement over O$n \log n$ required by the projection onto the simplex [50], [51], where n is the size of a block. In addition, it is worth noting that a wide variety of problems can benefit from this methodology. First, the use of algorithms that feature a projection step, e.g. projected descent methods and alternating direction methods, is very popular since they often provide a simple and efficient way to solve constrained convex optimization problem as opposed to more specialized active set methods. There is also a great deal of applications that feature simplex constraints, such as the aforementioned traffic assignment problem and games in general for the computation of strategy distributions, and ℓ_1-based approach in machine learning [50].

In the block simplex setting, this method achieves Omp time, and improvement from O$mp \log p$, where m is the number of blocks and p is the size of the blocks. An analogous complexity can be shown for blocks $\{p_i\}$ of different sizes.

VII. Experimental Results: Method Evaluation

Finally, we discuss the results of the numerical experiments in the traffic network setting. We denote $\gamma = \frac{\|x\|_0}{n}$ the fraction of nonzero entries in the true signal, and our experiments range with $\gamma \in [0.067, 1]$. Ultimately, our measure of accuracy is relative route flow error, denoted $\epsilon(x) = \|x^{\text{true}} - x\|_1 / \|x^{\text{true}}\|_1$.

We first present the test networks used in our experiments. Then, we present our findings with respect to accurate signal recovery and efficient signal recovery. In many situations, CS and BI recover signals more accurately than other methods, and BI has the advantage of having tune-able parameters. Among these methods, CO scales best; however, the block size significantly affects the performance of the CO and BI methods. Overall, we find that the CO method is recovers signals accurately and efficiently in large-scale routing applications.

A. Networks

We perform experiments on a variety of network scales, topologies, and solution distributions (i.e. route choice models). We first introduce the types of network models:

1) Benchmark Network (Figure 3): We use a benchmark network, which consists of 3 nodes, 5 links, and 7 routes.

2) Grid Networks: We construct grid networks of size 1x3 up to 11x11 (6-440 links, 12-217,800 routes). For each OD pair, up to 15 acyclic shortest paths are selected.

3) Full-Scale Los Angeles Network: Finally, we use a full-scale network located in Los Angeles, CA, consisting of 20,513 links, 10,538 nodes, 280,691 routes, 700 origins, 1,033 link flow sensors, 1,000 cellpath sensors (cell towers), and 1 million simulated agents in the 25-mile wide I-210 highway.

B. Accurate Recovery

We first note that $\epsilon(\cdot)$ is an extremely difficult metric, especially for under-determined settings, since having a non-empty nullspace implies an infinite number of feasible solutions.

1) Sparse Recovery (See Figure 4): As typical in highly under-determined settings, signal recovery is challenging across the board (note the blue to green circles in 4a and 4b). However, remarkably, in sparse settings ($\gamma < 0.1$), CS outperforms CO and BI in terms of accuracy of recovery. Note the error of the yellow and red circles in Figures 4a vs 4b; BI is not shown, due to scalability limitations. While the vast majority of high block-density instances (more red)

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

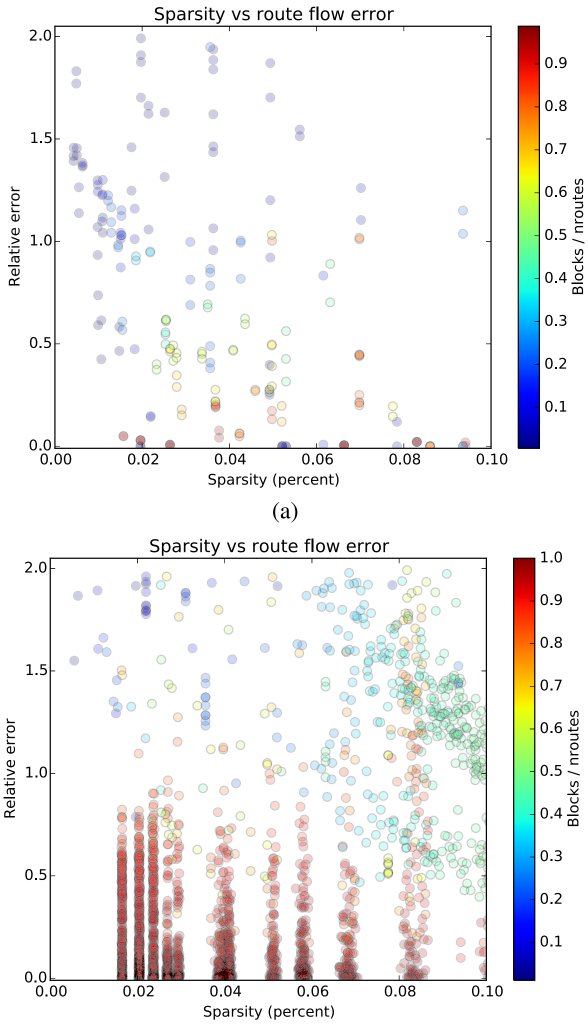


Fig. 4. Evaluation of methods (accuracy). Each circle is the outcome of an experiment (in a sparse setting) in terms of signal recovery accuracy. There are 447 CS experiments and 6143 CO experiments. The color scale indicates the ratio of blocks per number of routes, i.e. q/n. The relative error is $\epsilon(\cdot)$, and the percent sparsity γ is the percentage of nonzero entries in x. Although the recovery is challenging across the board, in high block-density settings, CS far outperforms CO in terms of accuracy. Best viewed in color. (a) Recovery accuracy via compressed sensing CS. (b) Recovery accuracy via convex optimization CO.

are recovered accurately through the CS approach, their CO counterpart typically recovers the signals with up to 85% error. The sufficient and necessary conditions on the (traffic) matrices for guaranteed sparse recovery is beyond the scope of this work.

2) Spread Recovery (See Figure 5): By tuning the hyper-parameters of the graphical model, the BI method is able to capture a range of spread for a benchmark network (See Figure 3), as measured by the cardinality of the solution $\|x\|_0$. The benchmark network has the following probabilistic model:

$$x^{AB} \sim \mathrm{Dir}((\alpha,\alpha)), x^{CA} \sim \mathrm{Dir}((\alpha,\alpha)), x^{CB} \sim \mathrm{Dir}((\alpha,\alpha))$$
$$y|\tilde{x} \sim \mathcal{N}(400x_2^{AB} + 600\ x_1^{CA} + 1000x_1^{CB}, \sigma^2)$$

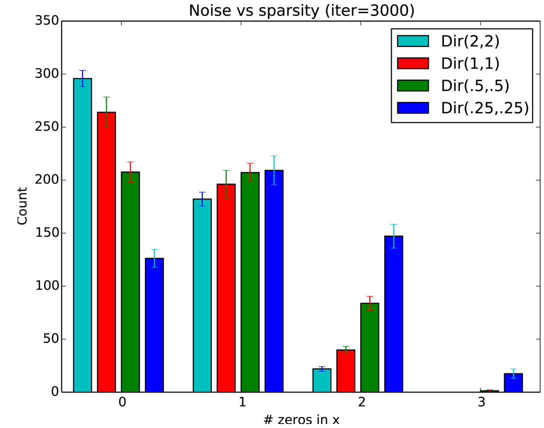


Fig. 5. Bayesian inference on benchmark network. For 500 trials using each prior $\alpha \in \{(2,2), (1,1), (0.5,0.5), (0.25,0.25)\}$ and each variance value ($\sigma^2 \in \{10^{-1}, 10^{-2}, 5\times10^{-3}, 10^{-3}, 5\times10^{-4}, 10^{-4}\}$), we record the number of zeros (i.e. $n - \|x\|_0$) in the recovered signal and display the aggregate counts as a histogram. Each trial is run with burn-in of 3000. The trials are averaged over the variances, and the (small) standard deviation bars are displayed accordingly. Best viewed in color.

where y is scalar (we have only one measurement) and

$$x = \left[x^{AB}\ x^{BA}\ x^{CA}\ x^{CB}\right]^T,$$

noting that x^{BA} is given. Following an experiment on this network consisting of 500 trials with experimentally determined burn-in of 3,000 samples, the spread recovery is seen by the number of zeros in the recovered solution; by tuning the hyperparameter α down, solutions with more zeros (x-axis of Figure 5) are encouraged for Dirichlet distributions. For instance, among solutions with two zeros (in a vector of size seven), $\alpha = (2,2)$ produces 20 instances (out of 500), whereas $\alpha = (0.25, 0.25)$ produces 150. Note that in the benchmark network, at least four of the entries must be nonzero, due to the four ODs, and thus at most three entries may be zeros. In contrast, though the CS method encourages sparsity, the BI method provides a tune-able parameter to encourage or discourage sparsity, which may be informed by complex priors on specific contexts, for instance travel time, safety, comfort, and may also vary ablock simplex to the next. The CO method neither encourages nor discourages spread.

C. Efficient Recovery

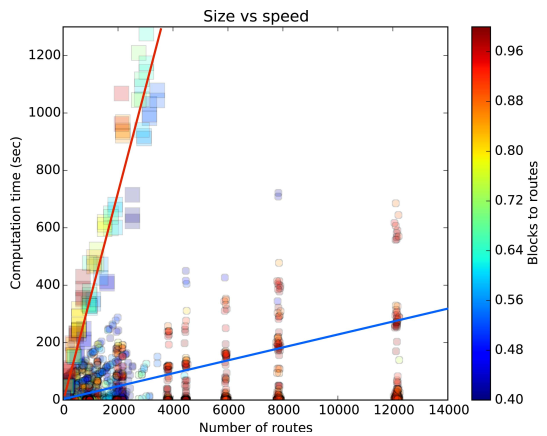
1) Scalability (See Figure 6): With respect to the signal size n, the CS method runs well for medium-sized problems, BI works primarily for small problems, and CO method scales well to large problems. In particular, the proposed methods solve problems of size $n = 1800, 200, 12000$ for the CS, BI, CO methods, respectively, in 10 minutes on commodity hardware (including 3.3GHz with 4GB memory instances and 2.5GHz with 15.25GB memory-optimized instances). Although our methods all scale linearly in each iteration, the constant factors vary widely, and the number of iterations can grow super-linearly with the number of routes considered.

2) Block Size vs Computational Efficiency (See Figure 6): For a fixed signal n, we find additionally that our methods behave significantly differently for varying block sizes

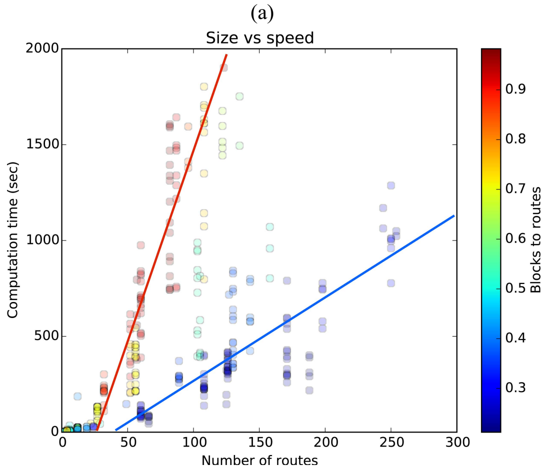
TABLE I

SUMMARY OF METHOD EVALUATION. WE HAVE CONSIDERED THREE MAIN COMPUTATIONAL APPROACHES (CS, BI, CO). FOR EACH ONE, WE SUMMARIZE ITS UNDERLYING MODEL, THE METHOD WE CHOSE OR DEVELOPED, ALONG WITH A COMPARISON OF ITS EFFICIENCY AND RECOVERY ACCURACY ATTRIBUTES

Attributes	Compressed sensing	Bayesian inference	Convex optimization
Model	Sparsity model	Dirichlet model	Model-free
Method	Random sampling	MCMC	Projected gradient
Problem scale	Medium	Small	Large
Block size	-	Faster if large	Faster if small
Sparse recovery	✓		
Spread recovery		✓	



(a)



(b)

Fig. 6. Evaluation of methods (scalability). BI is displayed separately from CS and CO due to the difference in scale of problems that the respective methods are able to compute. Noting the trend lines, each method scales linearly (empirically); however, the specific scaling ratio depends on the specific block sparsity pattern and varies by method. The color scale indicates the ratio of blocks per number of routes, i.e. q/n. Best viewed in color. (a) CS (squares) and CO (circles), with and trend lines, respectively.. (b) Scalability for BI, with and trend lines representing the high block-density and low block-density scenarios. BI scales better for low block-density scenarios.

(i.e. simplex dimensionality). We find that CO converges faster when the size of the blocks is small (note the red circles in Figure 6a), i.e. when the ratio of blocks per signal length q/n is high, that is, when we have few routes per block simplex flow measurement, e.g. cellpath, OD. This is a result of the dimensionality reduction scheme, which transforms a problem of n variables into one with $n-q$ variables, yielding a smaller optimization problem. In contrast, BI converges slower for small block sizes than for large block sizes (note the slope of the red circles vs the slope of the blue circles in Figure 6b). These experiments used an experimentally determined burn-in of 6000 samples. This slow convergence is explained by the difficulty for sampling-based methods such as MCMC to sample probable (or feasible) solutions in the presence of relatively many constraints. On the other hand, the sampling-based CS method is unaffected by block size.

3) Large-Scale Experiment: Only the CO method scaled to the full-scale Los Angeles network. In a quasi-static traffic network setting, we achieve an error of $\epsilon(x) = 0.1$, that is 90% route flow accuracy. This is the first accurate result on large-scale estimation of route flow in the traffic setting, using infrastructure available today. For a full treatment of this experiment and more detailed results, please refer to [4]. This result provides evidence that available cellular sensors are suitable for traffic estimation and management, and that CO is a suitable method for solving this problem accuracy and efficiently in large urban networks.

D. Discussion

The complexity of the route flow signal motivates approaches for the problem of block simplex signal recovery, from the perspectives of convex optimization, compressed sensing, and Bayesian inference. We present a summary of methods, results, and intuitive takeaways for the block simplex signal recovery problem in Table I. Our numerical results highlight salient trade-offs between the approaches. In particular, CO scales well to full-sized traffic networks, but may only recover one of infinitely many feasible solutions. On the other extreme, CS attempts to recover the sparsest among the feasible solutions. The proposed methods are appropriate for medium-sized networks, however the suitability of the sparsest solution is context-dependent. Striving for a middle ground, the BI approach allows the designer to encode context in the form of a prior, which may even vary per block simplex and thus is an appealing approach to recovering the most likely route flow signal. However, the BI method exhibits poor scalability beyond very small networks. Additionally, we find that the size and thus number of blocks q affects the empirical convergence speed of the methods in vastly different ways, independent of the overall size of the problem n.

VIII. CONCLUSION

This article has presented the problem of *block simplex signal recovery*, three principled approaches, an empirical method evaluation, and its application to an important real-world problem. It is the first extensive methods exposition and comparative analysis of this problem and aims to provide insights to its disparate yet natural methodologies. Each method has its strengths and weaknesses, and our experiments establish that there exist methods that exhibit some of our desired properties—efficient recovery and recovery with varying degrees of sparsity—but there does not yet exist a method that has all our desired properties. Nonetheless, a compressed sensing approach is shown to recover sparse solutions, and a Bayesian inference approach is shown to encourage different amounts of spread with a tune-able hyperparameter. Finally, our experiments show that the convex optimization approach is not only computationally efficient, but accurately recovers the signal in our routing application.

For future work, we plan to see how we can combine ideas from these three approaches and more, while keeping the strengths. For instance, we are interested in exploring an optimization approach to the Bayesian inference method through variational inference as an avenue to improve its scalability. An interesting open question is whether there is a formal connection between regularizing the BSLS objective and placing a prior on the spread of the solution. We are also interested in alternative sparsity relaxation techniques and simplex projection methods which may lead to better results. Furthermore, we are interested in the theoretical analysis of structured sparsity for non-ℓ_1-based compressive sensing. We also plan to study the robustness of these methods in the presence of measurement noise. Finally, real-world data rarely conforms to clean assumptions, as made in this work, and the future study of the conjunction of these proposed approaches with data filtering techniques will be important for real-world applicability.

Acknowledgement

The authors would like to thank Philipp Moritz, Fanny Yang, Richard Shin, Jerome Thai, and Steve Yadlowsky for many enlightening discussions and early collaborations on the proposed techniques. They would like to thank Jason Du and Chenyang Yuan for help with the large-scale experiments and visualizations. They are also grateful for insightful conversations with Professor Laurent El Ghaoui, Dr. Alexander Kurzhanskiy, and Professor Suvrit Sra.

References

[1] G. van der Laan and A. J. J. Talman, "On the computation of fixed points in the product space of unit simplices and an application to noncooperative *N*person games," *Math. Oper. Res.*, vol. 7, no. 1, pp. 1–13, Feb. 1982.
[2] T. M. Doup and A. J. J. Talman, "A new simplicial variable dimension algorithm to find equilibria on the product space of unit simplices," *Math. Program.*, vol. 37, no. 3, pp. 319–355, Oct. 1987.
[3] M. Jaggi, S. Lacoste-Julien, M. Schmidt, and P. Pletscher, "Block-coordinate frank–wolfe for structural SVMS," in *Proc. NIPS Workshop Optim. Mach. Learn.*, Lake Tahoe, NV, USA, 2012, pp. 1–5.
[4] C. Wu, J. Thai, S. Yadlowsky, A. Pozdnoukhov, and A. Bayen, "Cellpath: Fusion of cellular and traffic sensor data for route flow estimation via convex optimization," *Transp. Res. Procedia*, vol. 7, pp. 212–232, Aug. 2015.
[5] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, Mar. 2004.
[6] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*. vol. 2. Philadelphia, PA, USA: SIAM, 2001.
[7] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
[8] Y. C. Eldar and G. Kutyniok, *Compressed Sensing: Theory and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2012.
[9] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magn. Reson. Med.*, vol. 58, no. 6, pp. 1182–1195, Oct. 2007.
[10] D. Koller and N. Friedman, *Probabilistic Graphical Models Principles and Techniques*. Cambridge, MA, USA, MIT Press, 2009.
[11] M. J. Wainwright and M. I. Jordan, "Graphical models, exponential families, and variational inference," *Found. Trends Mach. Learn.*, vol. 1, nos. 1–2, pp. 1–305, Jan. 2008.
[12] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde, "Model-based compressive sensing," *IEEE Trans. Inf. Theory*, vol. 56, no. 4, pp. 1982–2001, Apr. 2010.
[13] J. Vila and P. Schniter, "An empirical-Bayes approach to recovering linearly constrained non-negative sparse signals," in *Proc. 5th IEEE Int. Workshop Comput. Adv. Multi-Sensor Adapt. Process. (CAMSAP)*, Dec. 2013, pp. 5–8.
[14] M. Pilanci, L. E. Ghaoui, and V. Chandrasekaran, "Recovery of sparse probability measures via convex programming," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 2420–2428.
[15] L. R. Ford and D. R. Fulkerson, *Flow Network*. Princeton, NJ, USA: Princeton Univ. Press, 1962.
[16] H. J. van Zuylen and D. M. Branston, "Consistent link flow estimation from counts," *Transp. Res. B, Methodol.*, vol. 16, no. 6, pp. 473–476, Dec. 1982. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0191261582900066
[17] C. Fisk, "Trip matrix estimation from link traffic counts: The congested network case," *Transp. Res. B, Methodol.*, vol. 23, no. 5, pp. 331–336, Oct. 1989.
[18] H. Yang, T. Sasaki, Y. Iida, and Y. Asakura, "Estimation of origin-destination matrices from link traffic counts on congested networks," *Transp. Res. B, Methodol.*, vol. 26, no. 6, pp. 417–434, Dec. 1992.
[19] H. Yang, Y. Iida, and T. Sasaki, "The equilibrium-based origin-destination matrix estimation problem," *Transp. Res. B, Methodol.*, vol. 28, no. 1, pp. 23–33, Feb. 1994.
[20] H. Yang, "Heuristic algorithms for the bilevel origin-destination matrix estimation problem," *Transp. Res. B, Methodol.*, vol. 29, no. 4, pp. 231–242, Aug. 1995.
[21] T. Seo, A. M. Bayen, T. Kusakabe, and Y. Asakura, "Traffic state estimation on highway: A comprehensive survey," *Annu. Rev. Control*, vol. 43, pp. 128–151, Jan. 2017.
[22] J. G. Wardrop and J. I. Whitehead, "Correspondence. Some theoretical aspects of road traffic research," *ICE Proc: Eng. Divisions*, vol. 1, no. 5, pp. 767–768, Oct. 1952.
[23] Y. Sheffi, *Urban Transportation Networks: Equilibrium Analysis with Mathematical*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1985.
[24] M. G. H. Bell and Y. Iida, *Transportation Network Analysis*. West Sussex, U.K.: Wiley, 1997.
[25] C. F. Daganzo and Y. Sheffi, "On stochastic models of traffic assignment," *Transp. Sci.*, vol. 11, no. 3, pp. 253–274, Aug. 1977.
[26] M. J. Maher and P. C. Hughes, "A probit-based stochastic user equilibrium assignment model," *Transp. Res. B, Methodol.*, vol. 31, no. 4, pp. 341–355, Aug. 1997.
[27] C. Fisk, "Some developments in equilibrium traffic assignment," *Transp. Res. B, Methodol.*, vol. 14, no. 3, pp. 243–255, Sep. 1980.
[28] M. G. H. Bell, C. M. Shield, F. Busch, and G. Kruse, "A stochastic user equilibrium path flow estimator," *Transp. Res. C, Emerg. Technol.*, vol. 5, nos. 3–4, pp. 197–210, Aug./Oct. 1997.
[29] E. Hato, M. Taniguchi, Y. Sugie, M. Kuwahara, and H. Morita, "Incorporating an information acquisition process into a route choice model with multiple information sources," *Transp. Res. C, Emerg. Technol.*, vol. 7, nos. 2–3, pp. 109–129, Apr./Jun. 1999.
[30] T. Tettamanti, H. Demeter, and I. Varga, "Route choice estimation based on cellular signaling data," *Acta Polytechnica Hungarica*, vol. 9, no. 4, pp. 207–220, Jan. 2012.
[31] J. Barcelö, L. Montero, L. Marqués, and C. Carmona, "Travel time forecasting and dynamic origin-destination estimation for freeways based on bluetooth traffic monitoring," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2175, no. 1, pp. 19–27, 2010.

[32] G. Michau, A. Nantes, A. Bhaskar, E. Chung, P. Abry, and P. Borgnat, "Bluetooth data in an urban context: Retrieving vehicle trajectories," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 9, pp. 2377–2386, Sep. 2017.

[33] E. Castillo, I. Gallego, J. M. Menéndez, and A. Rivas, "Optimal use of plate-scanning resources for route flow estimation in traffic networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 380–391, Jun. 2010.

[34] J. Yang and J. Sun, "Vehicle path reconstruction using automatic vehicle identification data: An integrated particle filter and path flow estimator," *Transp. Res. C, Emerg. Technol.*, vol. 58, pp. 107–126, Sep. 2015.

[35] T. Hunter, R. Herring, P. Abbeel, and A. Bayen, "Path and travel time inference from GPS probe vehicle data," in *Proc. Neural Inf. Process. Syst. Found. (NIPS)*, Dec. 2009, pp. 1–8.

[36] M. Rahmani and H. N. Koutsopoulos, "Path inference from sparse floating car data for urban networks," *Transp. Res. C, Emerg. Technol.*, vol. 30, pp. 41–54, May 2013.

[37] N. Breyer, D. Gundlegård, and C. Rydergren, "Cellpath routing and route traffic flow estimation based on cellular network data," *J. Urban Technol.*, vol. 25, no. 2, pp. 85–104, Apr. 2018.

[38] P. Ye and D. Wen, "Optimal traffic sensor location for origin–destination estimation using a compressed sensing framework," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 7, pp. 1857–1866, Jul. 2017.

[39] M. J. Maher, "Inferences on trip matrices from observations on link volumes: A Bayesian statistical approach," *Transp. Res. B, Methodol.*, vol. 17, no. 6, pp. 435–447, Dec. 1983.

[40] C. Tebaldi and M. West, "Bayesian inference on network traffic using link count data," *J. Amer. Stat. Assoc.*, vol. 93, no. 442, pp. 557–573, Jun. 1998.

[41] K. Parry and M. L. Hazelton, "Bayesian inference for day-to-day dynamic traffic models," *Transp. Res. B, Methodol.*, vol. 50, pp. 104–115, Apr. 2013.

[42] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, Dec. 2007.

[43] M. J. Wainwright, "Sharp thresholds for high-dimensional and noisy sparsity recovery using ℓ_1-constrained quadratic programming (Lasso)," *IEEE Trans. Inf. Theory*, vol. 55, no. 5, pp. 2183–2202, May 2009.

[44] W. R. Gilks, *Markov Chain Monte Carlo*. Hoboken, NJ, USA: Wiley, 2005.

[45] S. Chib and E. Greenberg, "Understanding the metropolis-hastings algorithm," *Amer. Stat.*, vol. 49, no. 4, pp. 327–335, Nov. 1995.

[46] R. E. Barlow and H. D. Brunk, "The isotonic regression problem and its dual," *J. Amer. Stat. Assoc.*, vol. 67, no. 337, pp. 140–147, Mar. 1972.

[47] M. J. Best and N. Chakravarti, "Active set algorithms for isotonic regression; A unifying framework," *Math. Program.*, vol. 47, nos. 1–3, pp. 425–439, May 1990.

[48] J. Thai, C. Wu, A. Pozdnukhov, and A. Bayen, "Projected sub-gradient with $\ell 1$ or simplex constraints via isotonic regression," in *Proc. 54th IEEE Conf. Decis. Control (CDC)*, Dec. 2015, pp. 2031–2036.

[49] R. J. Tibshirani, H. Hoefling, and R. Tibshirani, "Nearly-isotonic regression," *Technometrics*, vol. 53, no. 1, pp. 54–61, Feb. 2011.

[50] J. Duchi, S. Gould, and D. Koller, "Projected subgradient methods for learning sparse gaussians," in *Proc. 24th Conf. Uncertainty Artif. Intell.*, Jul. 2008, pp. 153–160.

[51] W. Wang and M. Á. Carreira-Perpiñán. (2013). "Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application." [Online]. Available: https://arxiv.org/abs/1309.1541

Cathy Wu received the B.S. and M.Eng. degrees from the Massachusetts Institute of Technology and the Ph.D. degree from UC Berkeley, all in electrical engineering and computer science. She holds a post-doctoral position at Microsoft Research AI. She will join MIT as an Assistant Professor in 2019. She works at the intersection of machine learning, optimization, and urban systems. Her work was acknowledged by several awards, including the 2019 Microsoft Location Summit Hall of Fame, the 2018 Milton Pikarsky Memorial Dissertation Award, the 2016 IEEE ITSC Best Paper Award, and fellowships from NSF, Berkeley Chancellor, NDSEG, and Dwight David Eisenhower.

Alexei Pozdnukhov received the Computer Science degree in mathematical physics from the Physics Department, Moscow State University, in 2003, and the Ph.D. degree in computer science from the École Polytechnique Fédérale de Lausanne, Switzerland, in 2006, following his research in machine learning methods and computer vision that he carried out at the IDIAP Research Institute, Martigny, Switzerland. His current research at UC Berkeley is in the area of complex data analysis in the domain of Smart Cities and is supported by NSF, NASA, California DOT, and industry.

Alexandre M. Bayen received the B.S. degree from Ecole Polytechnique, France, and the M.S. and Ph.D. degrees from Stanford University. He is a Professor of electrical engineering and computer science, and civil and environmental engineering. He is currently the Liao-Cho Professor of engineering with UC Berkeley, and is also the Director of the Institute of Transportation Studies (ITS). Prior to joining UC Berkeley, he was a Visiting Researcher at NASA Ames Research Center. He worked at the Department of Defense, France, where he holds the rank of Major. He has published two books and over 200 publications. He has received numerous awards, including the Ballhaus Award from Stanford, the NSF CAREER Award, the Okawa Research Grant Award, the Ruberti Prize from the IEEE, and the Huber Prize from the ASCE. He is the NASA Top 10 Innovators on Water Sustainability. His projects Mobile Century and Mobile Millennium received the Best of ITS Award for the Best Innovative Practice at the ITS World Congress and a TRANNY Award from the California Transportation Foundation. He is a recipient of the Presidential Early Career Award for Scientists and Engineers (PECASE) Award from the White House.