High-dimensional offline OD calibration for stochastic traffic simulators of large-scale road networks

Paper accepted for publication in Transportation Research Part B

Carolina Osorio*

January 14, 2019

Abstract

This paper considers high-dimensional offline calibration problems for large-scale simulation-based network models. We propose a metamodel simulation-based optimization (SO) approach. The proposed method is formulated and validated on a simple synthetic toy network. It is then applied to a high-dimensional case study of a large-scale Singapore network. Compared to two benchmark methods (a derivative-free pattern search method and the SPSA method), the proposed method improves the objective function estimates by two orders of magnitude. Moreover, this improvement is achieved after only 2 simulation runs. Hence, the proposed method is computationally efficient.

The main idea of the proposed approach is to embed, within the SO algorithm, information from an analytical (i.e., lower-resolution) yet differentiable and tractable network model. It is this analytical structural information that enables the SO algorithm to become both suitable for high-dimensional problems and computationally efficient. For a network with n links, the analytical network model is implemented as a system of n non-linear equations. Hence, it scales linearly with the number of links in the network and independently of link attributes (such as link length) and of the dimension of the route choice set.

1 Introduction

High-resolution urban traffic and mobility data is becoming increasingly available worldwide. This has sparked an increased interest in the development of traffic models to inform the design and the operations of urban mobility networks. Additionally, both the supply and the demand of our transportation systems are becoming more complex (e.g., with vehicle-to-vehicle and vehicle-to-infrastructure communications). This is leading to more sophisticated and more intricate traffic models. Nonetheless, in order to enable the use of

*Civil and Environmental Engineering Department, Massachusetts Institute of Technology, Office 1-232, Cambridge, Massachusetts 02139, USA, osorioc@mit.edu

this new generation of higher resolution traffic models to inform practice, there is a pressing need to provide practitioners with systematic tools that enable the adequate calibration and validation of these models. The problem of model calibration has been widely studied in the literature. Nonetheless, there is a lack of algorithms that can efficiently address the difficult (e.g., high-dimensional, simulation-based, non-convex) calibration problems faced in practice.

This paper focuses on the offline calibration of demand, as defined by origin-destination (OD) matrices, for simulation-based traffic models. In general, the problem of calibrating demand for simulation-based models aims to identify demand inputs for the simulator that minimize a distance function between network performance estimates (e.g., link flows, link speeds, link travel times) obtained from field measurement and those obtained via simulation.

We distinguish between the following two problems: (i) general OD estimation problems (also known as travel demand estimation problems), where the outputs are stand-alone OD matrices that can be used for a variety of planning and operational network analysis, and (ii) model calibration problems, where the goal is to calibrate (or estimate) the inputs (such as the demand inputs specified as OD matrices) of a specific affic model and the output is a calibrated traffic model that can itself be used for analysis. Traffic models have a variety of demand and supply parameters that require calibration. Hence, one implication of this distinction is that for the second class of problems (calibration problems) the values of the calibrated input model parameters should not be interpreted in isolation. In particular, the calibrated OD matrix should not be used as a stand-alone OD matrix for planning purposes. This is because it is calibrated conditional on the values of the other input model parameters (e.g., route choice model coefficients).

The two classes of problems often have common mathematical formulations. In particular, the first class can be formulated with the use of a traffic model. Since it is an underdetermined problem (i.e., there is an observability issue), the use of a traffic model serves to constrain the solution set in order to regularize the problem. In other words, the use of a model further constrains the problem by adding behavioral assumptions (e.g., route choice) or network knowledge.

This paper focuses on the second class of problems. Our goal is to calibrate the OD matrix input of a traffic simulator, which is then to be used to inform network planning and operations decisions. A recent review of general OD estimation problems (i.e., not limited to model calibration algorithms) is given in Cascetta *et al.* (2013).

A detailed problem formulation is given in Section 2.1. The goal of this paper is to design an OD calibration algorithm suitable for high-dimensional problems and large-scale networks. Moreover, the aim is to design a computationally efficient algorithm that can identify good quality solutions within few simulation runs. In practice, calibration algorithms are used within tight computational budgets (i.e., few simulation runs are carried out). Hence, the design of efficient algorithms contributes to current, and pressing, needs of practitioners.

A review of the recent OD calibration literature is provided in Zhang and Osorio (2017). Table 1, adapted from Zhang and Osorio (2017), summarizes recent OD calibration literature. It has been expanded to include this paper in the last row of the table. For each paper (i.e., each row), the table indicates whether the work focuses on: (i) OD calibration or on the joint calibration of OD's along with other demand and supply parameters, (ii) online or offline calibration, (iii) whether a time-dependent (i.e., dynamic) formulation is

addressed. For the largest case study of each paper, the table indicates the number of nodes, the number of links and the dimension of the decision vector. Table entries are left empty when the corresponding dimensions are not directly reported in the papers. All papers use counts as field data. The last column of the table indicates whether additional types of field data are also used.

The most common approach to the calibration of OD matrices for simulation-based traffic models has been the use of general-purpose optimization algorithms. The Stochastic Perturbation Simultaneous Approximation (SPSA) algorithm of Spall (1992, 2003) has been extensively used (Balakrishna *et al.* 2007, Vaze *et al.* 2009, Lee and Ozbay 2009, Cipriani *et al.* 2011, Ben-Akiva *et al.* 2012) and more recently extended to improve its efficiency (Cipriani *et al.* 2011, Cantelmo *et al.* 2014, Lu *et al.* 2015, Tympakianaki *et al.* 2015, Antoniou *et al.* 2015). The genetic algorithm (GA) has also been commonly used (Kim *et al.* 2001, Stathopoulos and Tsekeris 2004, Kattan and Abdulhai 2006, Vaze *et al.* 2009). These commonly used general-purpose algorithms are guaranteed to achieve asymptotic convergence properties for a broad class of problems (e.g., non-transportation problems). Nonetheless, this generality comes with a lack of computational efficiency. In other words, the algorithms are not designed to identify good quality solutions fast (i.e., within tight computational budgets or few simulation runs). Nonetheless, when used to address OD calibration problems, they are typically used within tight computational budgets. There is a current need to design efficient calibration algorithms.

In the broader field of OD estimation (not limited to the calibration of traffic models), there has been a long-standing and growing interest in the design of methods that are both more efficient and also more scalable. This has been mostly driven by challenges in the field of offline dynamic as well as real-time OD estimation (Bierlaire and Crittin 2004, Djukic *et al.* 2012, Prakash *et al.* 2018).

This paper proposes to achieve computational efficiency by designing algorithms specifically tailored for calibration problems. More specifically, we propose to embed within the algorithm analytical and differentiable problem-specific structural information that enables the algorithm to identify good quality solutions within few simulation runs. The main idea, which we have successfully used for other continuous transportation problems (e.g., signal control (Chong and Osorio 2018, Osorio and Nanduri 2015b), congestion pricing (Osorio and Atastoy 2017)), is to formulate, and embed within the algorithm, an analytical network model that provides an approximation of the (simulation-based) mapping between the decision vector and the objective function. For a calibration problem, the mapping approximates the relationship between the calibration vector (e.g., OD matrix) and the simulation-based components of the objective function (expected link flows).

The general challenges of designing simulation-based optimization (SO) algorithms apply to the design of SO calibration algorithms. Namely, the simulator is computationally costly to evaluate, the SO objective function is non-convex, and often non-differentiable, with many local minima. The calibration problem brings two additional challenges. First, the problem is high-dimensional. The dimension of the decision vector is often in the order of several thousand. In continuous SO, problems with dimension 200 are considered large-scale (Wang *et al.* 2016). Hence, there is a lack of, and a need for, algorithms that are both efficient and suitable for high-dimensional problems. Second, the formulation of an analytical mapping that approximates the OD matrix to the expected link flows is particularly challenging due to the need to provide an approximation of traffic assignment that is both analytical, differentiable, scalable and tractable (i.e., can be evaluated efficiently

Table 1: Recent demand calibration literature overview. Table adapted from Zhang and Osorio (2017)

	OD	Joint	Online	Offline	Simulation-based	Analytical	Time-dependency	Nodes	Links	Dimension	Other field data
Kim *et al.* (2001)	✓			✓		✓		9	14	8	
Tavana (2001)	✓			✓	✓		✓	178	441		
Zhou *et al.* (2003)	✓			✓	✓		✓	31	80		
Antoniou (2004)		✓	✓		✓		✓	15 on/off-ramps		80	Speed, density
Bierlaire and Crittin (2004)	✓		✓		✓		✓	296	618	627	
Jha *et al.* (2004)	✓[1]			✓	✓		✓	1,479	3,756	41,906	
Kattan and Abdulhai (2006)	✓			✓	✓		✓	30	50	400	
Nie (2006)	✓			✓		✓		17	23	4	Link-to-link counts, path travel time
Zhou and Mahmassani (2006)	✓			✓	✓		✓	31	80		Link-to-link counts
Balakrishna *et al.* (2007)		✓		✓	✓		✓	243	606	4,629	
Hazelton (2008)	✓			✓		✓	✓	21	50	1,190	
Zhang *et al.* (2008)	✓			✓		✓	✓	29 on/off-ramps		928	Subpath travel time
Lee and Ozbay (2009)	✓			✓	✓		✓	A one-way freeway			Speed
Vaze *et al.* (2009)		✓		✓	✓		✓	825	1,767	6,470	Subpath travel time
Huang (2010)		✓	✓		✓		✓	56	85	638	Speed, density
Cipriani *et al.* (2011)	✓			✓	✓		✓	221	734		Speed
Flötteröd *et al.* (2011)	✓			✓	✓		✓	24,180	60,492	187,484	
Frederix *et al.* (2011)	✓			✓		✓	✓	39	56		
Verbas *et al.* (2011)	✓			✓	✓		✓	28,406	68,490	10^6-10^8	
Ben-Akiva *et al.* (2012)		✓		✓	✓		✓	1,698	3,180	69,093	Link travel time
Lu *et al.* (2015)		✓		✓	✓		✓	831	1040	373,646	
Tympakianaki *et al.* (2015)	✓			✓	✓		✓	N.A.	1,101	1,848	
Zhang and Osorio (2017)	✓			✓	✓			11,345	24,335	2,585	
Osorio (this paper)	✓			✓	✓			924	1,150	4,050	

[1] Demand parameters of driver behavior and route choice models are also calibrated

for large-scale networks). These challenges are described in more detail in Sections 2.1 and 2.2.

Our recent work in this area (Zhang and Osorio 2017) has also followed these ideas, yet it has proposed the use of an (approximate) analytical mapping with exogenous assignment. It has shown, for a large-scale Berlin case study, that the analytical information from the analytical network model provides structural information needed to yield an efficient algorithm. Nonetheless, for general networks with elaborate traffic management strategies (e.g., congestion pricing), the use of a mapping with endogenous assignment is more suitable. For instance, the case study of this paper considers a Singapore network with congestion pricing. Thus, the use of an analytical network model with endogenous assignment is appropriate. The methodological differences between the methodology proposed in this paper and that of our recent demand calibration work (Zhang and Osorio 2017, Zhang *et al.* 2017) are detailed after the proposed methodology is presented in Section 2.2.

This paper uses the metamodel SO algorithm of Osorio and Bierlaire (2013), which builds upon the derivative-free trust-region (TR) algorithm of Conn *et al.* (2009). Computational efficiency is achieved through the formulation and use of an analytical and differentiable metamodel that approximates the mapping between the OD matrix and expected link flows, allowing for endogenous traffic assignment.

This paper proposes a methodology to calibrate the OD matrix input of a traffic simulator, which is then to be used to inform network planning and operations decisions. The traffic simulator we use is a mesoscopic dynamic traffic model. The simulator defines demand as a set of time-dependent OD matrices. This paper focuses on the estimation of an OD matrix for a single time period. Its extension to simultaneously estimate multiple OD matrices is given in Osorio (2019). We propose a methodology that enables high-dimensional offline OD matrix calibration problems for large-scale simulation-based network models to be addressed in a computationally efficient way. The proposed method is applied to a high-dimensional problem (4050 decision variables) for a congested large-scale network of Singapore. More specifically, the main contributions of this paper are the following.

- **Computational efficiency.** For all algorithmic runs of the Singapore case study, the proposed method identifies points with objective function estimates that are reduced (i.e., improved) by two orders of magnitude compared to those of the initial points. This is achieved within 2 simulation runs (i.e., after 2 demand points are simulated). Hence, the proposed method is computationally efficient. The method is compared to a general-purpose derivative-free algorithm and to the (Simultaneous Perturbation Stochastic Approximation) algorithm, which upon depletion of the computational budget (i.e., after 20 points are simulated) yield average improvements of 4.4% and 1.4%, respectively. Experiments with larger computational budgets are also carried out. They indicate that even when allowing for more than double the computation time of the proposed method, the benchmark methods are still outperformed by the proposed method by 2 orders of magnitude. This remarkable computational efficiency establishes the proposed approach as a potential building block for real-time (i.e., online) OD calibration.
- **Scalability.** The performance of the proposed method is evaluated on a high-dimensional problem with 4050 decision variables. In the field of SO, problems with dimension in the order of 200 are considered high-dimensional (Wang *et al.* 2016). Hence, the proposed approach illustrates how the use of analytical structural information enables high-dimensional SO problems to be addressed. Moreover, such problems can be addressed

efficiently. The structural information is derived by an analytical network model which was first formulated in Osorio and Atastoy (2017) for toll optimization. The model is specified as a system of nonlinear equations. For a network with n links, the model is implemented as a system of n nonlinear equations. This makes it a suitable model for high-dimensional OD calibration problems.

• **Metamodel SO.** The proposed approach is a metamodel SO method. More specifically, at every iteration of the SO algorithm an analytical approximate optimization problem is solved. The latter problem is known as the metamodel optimization problem. It replaces the simulation-based objective function with an analytical objective function, known as the metamodel. The key component of the proposed metamodel approach is the use of a metamodel that embeds a problem-specific, analytical and differentiable, approximation of the simulation-based objective function. The goal of the problem-specific component is to provide the SO algorithm with structural information. It is this information that enables the design of a computationally efficient algorithm. This general idea of formulating a metamodel with a problem-specific component has been the key to designing computationally efficient algorithms for various continuous transportation SO problems (Chong and Osorio 2018, Osorio and Selvam 2017, Osorio and Chong 2015, Osorio and Nanduri 2015a,b, Osorio *et al.* 2017). For a given problem, the main challenge is the formulation of an analytical and differentiable network model that approximates well the SO objective function while also being scalable (i.e., suitable for large-scale networks and for high-dimensional problems), computationally tractable (because the metamodel optimization problem is solved at *every* iteration of the SO algorithm), and differentiable (such that efficient traditional gradient-based algorithms can be used to address the metamodel optimization problem). The formulation of such a metamodel is particularly intricate for calibration problems because it involves formulating an analytical approximate mapping of the decision vector (e.g., OD matrix) to the simulation-based performance metrics (e.g., link flows). Given the elaborate traffic dynamics present in large-scale road networks, the formulation of such an analytical network model is a challenge. Compared to our past metamodel formulations for various demand calibration problems (Zhang and Osorio 2017, Zhang *et al.* 2017), the proposed approach both allows for an analytical description of the impacts on the network of endogenous traffic assignment (i.e., it describes how changes in demand can impact route choice and, ultimately, also impact the spatial distribution of congestion throughout the network) yet it still uses a tractable metamodel formulation. For instance, compared to our past calibration work, a single metamodel is used, rather than one metamodel for each link with a sensor. The proposed method is particularly suitable for the calibration of large-scale networks with elaborate traffic dynamics (e.g., with traffic-responsive traffic management strategies).

• **Robustness to the quality of initial points.** The experiments indicate that the structural information provided by the analytical network model enable the SO algorithm to be robust the quality of the initial points. This is of particular interest for cases where the prior OD matrix is not reliable (e.g., outdated).

• **Transportation practice.** Transportation practitioners address these calibration problems under tight computational budgets. There is a lack of computationally efficient calibration algorithms. The proposed approach contributes to fill this gap. Moreover, given the growing interest, among both public and private transportation stakeholders, in the use of traffic models to inform their decision making, the design of efficient calibration algorithms is essential for transportation practice.

This paper is structured as follows. Section 2 formulates the OD calibration problem and the proposed methodology. The method is validated with experiments on a small toy network (Section 3) and then applied and benchmarked on a high-dimensional problem for a large-scale network of Singapore (Section 4). Conclusions are given in Section 5.

2 Methodology

This section presents the OD calibration problem of interest (Section 2.1) and formulates the proposed methodology (Section 2.2).

2.1 Problem formulation

An urban area is spatially divided into traffic analysis zones, each of which can constitute a potential origin or destination for trips. Travel demand, for a given time period, is modeled by an origin-destination (OD) matrix, which defines the expected number of trips from each origin to each destination. We focus on an offline calibration problem. During the time interval of interest, we consider a single OD matrix. The OD calibration problem consists of identifying an OD matrix that leads to simulated traffic metrics that are reflective of traffic conditions observed from the field. For offline calibration, the OD matrix is determined offline based on historical field measurements. This differs from online (i.e., real-time) OD calibration, where the OD matrix is determined in real-time based on real-time traffic measurements. To formulate the problem, we introduce the following notation.

d_z	expected travel demand for OD pair z (scalar);
f	simulation-based objective function;
F_i	flow on link i as defined by the simulator;
y_i	average flow on link i estimated from field data (scalar);
$\tilde{d}_z$	prior value for the expected demand for OD pair z (scalar);
u_1	vector of endogenous simulation variables;
u_2	vector of exogenous simulation parameters;
δ	scalar weight parameter for prior information;
$d^{\max}$	upper bound vector;
z_0	number of OD pairs;
$\mathcal{I}$	set of indices of links with sensors;
$\mathcal{Z}$	set of indices of OD pairs $\mathcal{Z} = \{1, 2, \ldots, z_0\}$;

The problem is traditionally formulated as follows:

$$\min_d f(d) = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \left(y_i - E[F_i(d, u_1; u_2)]\right)^2 + \delta \frac{1}{|\mathcal{Z}|} \sum_{z \in \mathcal{Z}} \left(\tilde{d}_z - d_z\right)^2 \tag{1}$$

$$0 \leq d \leq d^{\max}. \tag{2}$$

The decision vector d is the vector of expected demand for each OD pair. The notation $|\mathcal{I}|$ (resp. $|\mathcal{Z}|$) denotes the cardinality of the set $\mathcal{I}$ (resp. $\mathcal{Z}$). The first summation term of (1) represents the distance between traffic conditions described by field measurements and traffic conditions estimated by the simulator. More specifically, a set $\mathcal{I}$ of links in the network are deployed with sensors. This set is usually a low-dimensional set, i.e., there

are few links with sensors. For a given traffic performance metric (e.g., link counts, link speeds), the corresponding field measurement on link i is denoted y_i, and the corresponding simulation-based estimate on link i is denoted $E[F_i(d, u_1; u_2)]$. In this paper, we focus on the measurements that are most commonly available worldwide: link counts. In other words, y_i represents an estimate, based on field measurements, of flow on link i, and $E[F_i(d, u_1; u_2)]$ represents the simulation-based expected flow on link i. The simulator has a vector of endogenous variables, u_1 (e.g., link queue-lengths, travel times) and a vector of exogenous variables, u_2 (e.g., specification of other demand models, such as route choice; specification of supply models, such as traffic management strategies or link attributes). The second term of the objective function (1) represents the distance between the decision vector, d, and a prior OD matrix, $\tilde{d}$. This distance is normalized by the number of terms in the summation $|\mathcal{Z}|$ and by a scalar weight factor δ. The OD calibration problem is, for a realistic network, an underdetermined problem. In particular, the dimension of the decision vector, d, is often orders of magnitude higher than the cardinality of the set of links with measurements, $\mathcal{I}$. The second summation of (1) is introduced such as to yield solutions that are physically plausible (e.g., OD matrices that are consistent with land-use patterns in the city). In the literature, the prior matrix is also referred to as the seed matrix or the initial matrix. It is often a matrix obtained from past traffic studies, such as studies with lower resolution traffic models (e.g., static assignment models). The parameter δ reflects the level of reliability or importance of the prior OD matrix. The OD calibration problem has upper and lower bound constraints (2).

Let us illustrate the intricacy of the function $E[F_i(d, u_1; u_2)]$ (hereafter denoted $E[F_i]$). It embeds a detailed description of elaborate spatial-temporal supply-demand interactions. More specifically, it maps travel demand, as defined by the OD matrix, to link-level traffic conditions. For a given realization of demand, the flows on a given link i are the outcomes of the, pre-trip and en-route, travel decisions made by each of the simulated travelers (e.g., route choice, lane-changing, car-following). These decisions are made based on past and prevailing traffic conditions. In particular, they depend themselves on the spatial and temporal distribution of link flows. Hence, an estimate of the expected link flow ($E[F_i]$) can be obtained by running sequential simulations, which can be interpreted as a learning process over consecutive days. Every day the travelers make travel decisions based on the most recent (current and previous days) traffic conditions, this leads to new traffic conditions (i.e., new link flows).

Problem (1)-(2) consists of a continuous, most often non-convex and non-differentiable, simulation-based objective function with analytical (i.e., non-simulation-based) bound constraints. The main challenges of this SO problem are the following. First, it is a high-dimensional problem. The dimension of d is often in the thousands. Problems with a dimension in the order of 200 are considered high-dimensional for SO algorithms (Wang *et al.* 2016). The case study of Section 4 addresses a problem of dimension 4050. Second, as discussed above, the problem is underdetermined with many local minima. The simulation-based term $E[F_i]$ is a nonlinear and non-convex function. It is computationally costly to estimate. The goal of this work is to design an algorithm that tackles the above challenges. Additionally, the goal is to design a computationally efficient algorithm. In other words, the algorithm identifies solutions with good performance at a low computational cost (i.e., within few simulation runs).

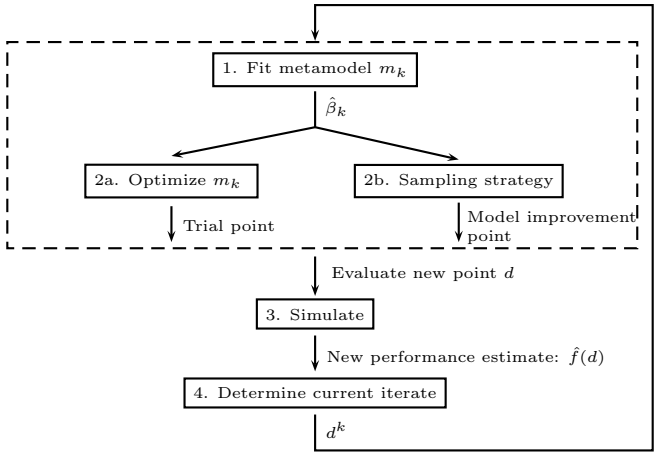


Figure 1: Metamodel simulation-based optimization framework

2.2 Metamodel formulation

We first summarize the main ideas of the SO framework used in this paper. We then present the metamodel formulation. We use the general metamodel SO approach of Osorio and Bierlaire (2013), which is based on the derivative-free trust-region (TR) algorithm of Conn *et al.* (2009). The main steps of each iteration of the SO algorithm are displayed in Figure 1. The main idea of a metamodel approach is to approximate the simulation-based objective function with an analytical function, known as the metamodel, and to use this metamodel to solve an analytical and differentiable optimization problem, known as the metamodel optimization problem. The use of the metamodel allows to perform optimization directly on an analytical and differentiable problem, for which a variety of standard solvers are available.

Let m_k denote the metamodel at iteration k of the algorithm. The metamodel is a parametric function with parameter vector β_k. The following steps are carried out at every iteration of the metamodel SO algorithm. The set of simulation observations, collected so far, are used to fit the parameters of the metamodel (see Step 1 of Figure 1). An estimate of the metamodel parameter vector is denoted $\hat{\beta}_k$ in Figure 1. This fitting or estimation problem is defined as a least-squares problem that minimizes the distance between the simulation-based objective function estimates and the metamodel approximations. Appendix A formulates the least squares problem. The metamodel is then used in Step 2a to solve the metamodel optimization problem, the solution of which is known as the trial point. The latter is simulated (Step 3), leading to new objective function estimates, $\hat{f}(d)$. In Step 4, the point with best performance, known as the current iterate, is determined. It is denoted d^k in Figure 1. As the iterations advance, these steps are iterated, such as to collect new simulation observations, update the metamodel and ultimately identify points with improved (simulation-based) performance. Step 2b of Figure 1 indicates that points other than those that are solutions to the metamodel optimization problem can also be simulated. The sampling of such points can aim to improve the metamodel fit or the geometric properties of the sampled space.

In the proposed framework, a metamodel is fitted and a metamodel optimization problem is solved, *at every iteration* of the SO algorithm. This differs from the pioneering

metamodel literature which first collects an extensive number of simulation observations through a design of experiments and then proceeds once (rather than iteratively) to fit the metamodel and to solve an analytical metamodel optimization problem. In this case, the metamodel replaces the simulator and the SO problem is replaced by a single analytical optimization problem. These traditional approaches lack computational efficiency because they typically require a large number of simulation evaluations prior to performing optimization.

Instead, in the iterative approach adopted here the role of the metamodel is not to replace the simulator nor is it to be an accurate approximation of the simulator. Instead, the role of the metamodel is to provide the SO algorithm with analytical (approximate) problem-specific structural information. This allows the SO algorithm to no longer treat the simulator as a black-box, which in turn enables it to efficiently search high-dimensional feasible regions. In other words, it enables the SO algorithm to achieve both computational efficiency and scalability.

Metamodels are classified as functional metamodels, which are general-purpose functions chosen based on mathematical properties, and physical metamodels, which are problem-specific functions (Søndergaard 2003, Chaper 2). The majority of the metamodel literature has focused on the use of functional metamodels, the most common choices are low-order polynomials, radial-basis functions, Kriging functions (Jones *et al.* 1998, Barton and Meckesheimer 2006, Wild *et al.* 2008, Kleijnen *et al.* 2010, Ankenman *et al.* 2010). A review of metamodel approaches appears in Osorio (Chap. 5, 2010).

To formulate the proposed metamodel, we introduce the following notation. The index

k refers to a given SO algorithm iteration.

m_k	metamodel function;
β_k	parameter vector of metamodel m_k;
$\beta_{k,j}$	scalar element j of the parameter vector β_k;
$f_A(d)$	analytical approximation of the first summation of the SO objective function provided by an analytical traffic model;
$\phi(d;\beta_k)$	polynomial component of the metamodel m_k;
	Endogenous variables of the analytical traffic model:
λ_i	expected hourly demand for link i (scalar);
k_i	expected density per lane of link i (scalar);
v_i	expected (space-mean) speed per lane of link i (scalar);
t_r	expected travel time for route r (scalar);
$P(r)$	route choice probability for route r (scalar).
	Exogenous parameters of the analytical traffic model:
k_i^{jam}	jam density per lane of link i (scalar);
$v_i^{\max}$	maximum speed of link i (scalar);
q^{cap}	lane flow capacity (scalar);
n_i	number of lanes of link i (scalar);
ℓ_i	average lane length of link i (scalar);
z_r	toll cost for route r;
θ_1, θ_2	coefficients of the route choice model (scalars);
$\alpha_{1,i}, \alpha_{2,i}$	parameters of the fundamental diagram of link i (scalars);
c	scaling parameter common to all links (scalar);
$O(r)$	OD pair of route r;
$\mathcal{R}_1(i)$	set of routes that include link i;
$\mathcal{R}_2(s)$	set of routes of OD pair s;
$\mathcal{L}(r)$	set of links of route r.

In this paper, the metamodel optimization problem, at iteration k of the SO algorithm,

is formulated as follows.

$$\min_{d} m_k(d;\beta_k) = \beta_{k,0} f_A(d) + \phi(d;\beta_k) + \delta \frac{1}{|\mathcal{Z}|} \sum_{z\in\mathcal{Z}} \left(\tilde{d}_z - d_z\right)^2 \tag{3}$$

$$\phi(d;\beta_k) = \beta_{k,1} + \sum_{z\in\mathcal{Z}} \beta_{k,z+1} d_z \tag{4}$$

$$f_A(d) = \frac{1}{|\mathcal{I}|} \sum_{i\in\mathcal{I}} (y_i - \lambda_i)^2 \tag{5}$$

$$\lambda_i = \sum_{r\in\mathcal{R}_1(i)} P(r) d_{O(r)} \tag{6}$$

$$P(r) = \frac{e^{\theta_1 t_r + \theta_2 z_r}}{\sum_{j\in\mathcal{R}_2(O(r))} e^{\theta_1 t_j + \theta_2 z_j}} \tag{7}$$

$$t_r = \sum_{i\in\mathcal{L}(r)} t_i \tag{8}$$

$$t_i = \frac{\ell_i}{v_i} \tag{9}$$

$$v_i = v_i^{\max}\left(1 - \left(\frac{k_i}{k_i^{\text{jam}}}\right)^{\alpha_{1,i}}\right)^{\alpha_{2,i}} \tag{10}$$

$$k_i = c\frac{k_i^{\text{jam}}}{q^{\text{cap}}}\frac{\lambda_i}{n_i} \tag{11}$$

$$0 \le d \le d^{\max}. \tag{12}$$

There are two main differences between the SO Problem (1)-(2) and the metamodel optimization Problem (3)-(12). First, the simulation-based objective function (f of (1)) is replaced by an analytical metamodel function (m_k of (3)). The bound Constraint (2) appears as Constraint (12), yet Problem (3)-(12) has an additional set of Constraints (4)-(11) defined as a system of nonlinear equations. These serve to define the analytical components of the metamodel. We now describe these in more detail.

Note that the second summation of the SO objective function (Eq. (1)) is an analytical function (i.e., we do not need to evaluate it via simulation). Hence, there is no need to propose an analytical approximation for this summation term, it appears as is in the metamodel function (Eq. (3)). On the other hand, the first summation of the SO objective function (Eq. (1)) does contain simulation-based terms. Hence, the metamodel proposes an analytical approximation of this summation term. This approximation is defined as the sum of a functional (i.e., general-purpose) metamodel component, denoted ϕ, and of a physical metamodel component, denoted f_A. The functional component is defined as a linear (polynomial) function (Eq. (4)). The physical component is defined in Eq. (5). The role of this physical component is to provide a problem-specific, analytical and differentiable, approximation of the first summation of Eq. (1). This approximation considers the first summation and replaces the simulation-based term ($E[F_i]$) by the expected hourly demand on link i, denoted λ_i. The latter is derived from an analytical traffic model which is defined by the system of nonlinear Equations (6)-(11). This analytical traffic model was first formulated in Osorio and Atastoy (2017) to address a toll optimization problem.

Let us now describe the analytical traffic model. The expected hourly demand on link i, λ_i, is defined in Eq. (6) as the sum of the expected route demand, for all the routes

that include link i. The expected route demand for route r is defined as the product of the expected (hourly) demand for the OD of that route, $d_{O(r)}$, times the route choice probability, $P(r)$. The latter probability is defined in Eq. (7) as a multinomial logit model with a utility function that depends on the route's travel time, t_r, and travel toll cost, z_r. Since the case study of Section 4 considers a Singapore network with tolled links, the toll cost is accounted for in this formulation. For networks without tolling, the tolling cost terms in the route choice model can be neglected. The (expected) travel time for route r is defined as the sum of the (expected) link travel times for each link of route r (Eq. (8)). The (expected) travel time of link i is approximated in Eq. (9) as the ratio of the average link length (averaged over the various lanes of the link) and the expected (space-mean) speed of the link. The latter is defined in Eq. (10) and consists of a differentiable approximation of the non-differentiable fundamental diagram used by the mesoscopic simulator used in the case studies of this paper (DynaMIT, Ben-Akiva *et al.* (2010)). Finally, the link's (expected) density (Eq. (12)) is approximated by assuming a linear relationship between expected link demand, λ_i, and expected link density, k_i. The coefficients of this linear relationship are the link's jam density, k_i^{jam}, a lane flow capacity term, q^{cap}, which is assumed the same for all lanes of the network, and a constant, c, that is fitted based on insights from toy network experiments. For the experiments of this paper, it is set to 1/6.

To summarize, the first summation of the SO objective function is replaced by an analytical function, which is defined as a problem-specific analytical approximation derived by an analytical traffic model (term f_A) that is corrected parametrically by both a scaling term (term $\beta_{k,0}$) and an additive linear error term (term ϕ). The proposed approach addresses the non-differentiable simulation-based optimization Problem (1)-(2) by iteratively solving a set of analytical and differentiable (metamodel) optimization problems of the form (3)-(12).

As the case studies of this paper indicate, the use of an analytical traffic model yields a good global approximation (i.e., a good approximation in the entire feasible region) of the SO objective function, while metamodels that are limited to functional components are formulated to provide a good local approximation (e.g., a good approximation near the current iterate). The use of a global approximation enables the algorithm to become robust to the quality of the initial point. In other words, the proposed SO algorithm has similar performance regardless of the quality of the initial points. This is in contrast to general-purpose SO algorithms that, when used with tight computational budgets as in this paper, are sensitive to the quality of the initial point. The metamodel parameters are fitted by solving a weighted least squares problem, which is described in Appendix A.

Of particular importance is that, for a network with n links, the analytical traffic model (Eq. (6)-(11)) is implemented as a system of n nonlinear equations. This makes it scalable. In other words, the complexity of the model (i.e., the dimension of the corresponding system of equations) scales linearly with the network size. Its complexity does not depend on link attributes (e.g., link lengths). Most importantly, the model has endogenous traffic assignment, yet its complexity is independent of the dimension of the route choice set. For instance, for the Singapore case study of this paper, the analytical model considers a network with 860 links, 4050 OD pairs and over 18200 routes; and is implemented as a system of 860 nonlinear equations. Additionally, the analytical derivatives of this system of equations have been derived. This enables us to address the metamodel optimization problem by efficiently using a variety of traditional gradient-based optimization algorithms.

The analytical traffic model (Eq. (6)-(11)) embeds a variety of simplifications com-

pared to the simulation-based traffic model used for the case studies of this paper. It does not capture traffic dynamics, rather it is a stationary formulation. This is discussed in more detail in the next paragraph. Unlike the simulator, the analytical model assumes all lanes of a link are homogenous (i.e., they have both common supply parameters and they exhibit common traffic conditions). The simulator describes between-link interactions (e.g., vehicular spillbacks) in detail, while the analytical model has decoupled link models. More specifically, for a given link i, its traffic conditions are fully defined based on the links expected demand, λ_i, and the links supply configuration (e.g., its supply parameters $k_i^{\text{jam}}, v_i^{\max}, q^{\text{cap}}, n_i, \ell_i, \alpha_{1,i}$ and $\alpha_{2,i}$). Metamodels that analytically capture vehicular spillback have been used for signal control studies (Osorio *et al.* 2017, Osorio and Chong 2015). Their use for calibration has yet to be explored. As mentioned above, the fundamental diagram is a differentiable approximation of the simulators (non-differentiable) fundamental diagram. The route choice model is also simplified. For example, in DynaMIT, the route choice process is described in detail through a set of three types of models: habitual route choice, pre-trip route choice and en-route choice. Also, the simulator accounts for the populations heterogeneity in value of time (VOT) by assuming a random VOT that is lognormal distributed. In contrast, the analytical model considers a unique value of time for the entire population. The simulator takes as input a pre-defined route choice set. This route choice set is the same as that used by the analytical traffic model. In our past OD calibration work with a Berlin metropolitan region case study, we showed that the use of an analytical traffic model with a fixed exogenous route choice can be suitable even when the simulator has an endogenous route choice set (Zhang and Osorio 2017).

The OD calibration Problem (1)-(2) assumes a dynamic simulation-based traffic model is used (i.e., the expectation $E[F_i]$ in (1) is an expectation from a dynamic traffic model). On the other hand, the analytical network model used to formulate the metamodel, which is defined by Equations (6)-(11), considers a stationary regime. The fact that a stationary network model, defined as a simple system of nonlinear equations, can capture sufficient problem structure of an underlying dynamic network model is remarkable. In the majority of our past work, stationary formulations have been sufficient to approximate various network performance metrics of dynamic simulators. This has held across simulation softwares (e.g., Aimsun, MATSim, DynaMIT) and across optimization problems (e.g., signal control, congestion pricing, calibration). Metamodel formulations based on analytical dynamic network models have been proposed, yet they are less tractable than their stationary counterparts (Chong and Osorio 2018). Their use for high-dimensional SO problems, like OD estimation problems, has yet to be explored. Moreover recent work has showed that stationary models, like the one proposed in this paper, can be suitable to address dynamic OD estimation problems (i.e., problems where a set of time periods are considered and one OD matrix is estimated for each time period) (Osorio 2019).

There are three main differences between the proposed approach and that of Zhang and Osorio (2017). First, the calibration Problem (1)-(2) is formulated, in Zhang and Osorio (2017), without the use of the normalization terms $1/|\mathcal{I}|$ and $1/|\mathcal{Z}|$. Second, the method of Zhang and Osorio (2017) formulates one metamodel for each of the links with sensors (i.e., $\forall i \in \mathcal{I}$). In other words, there is one metamodel for each simulation-based term, $E[F_i]$, in the SO objective function (Eq. (1)). The proposed approach formulates a single metamodel for the entire SO objective function. This leads to a lower dimensional set of metamodel parameters. The third, and most important, difference is that the proposed metamodel uses an analytical traffic model with endogenous traffic assignment. In other

words, the route choice probabilities are defined as a function of the OD matrix. In Zhang and Osorio (2017), the traffic assignment is exogenous. The use of endogenous assignment allows for a more realistic analytical traffic model. Nonetheless, this comes at the cost of decreased tractability in the analytical network model. More specifically, for a network with n links, the analytical model of Zhang and Osorio (2017) is implemented as a linear system of n equations, while the proposed model is implemented a nonlinear system of n equations.

The work of Zhang *et al.* (2017) proposes a metamodel approach to the calibration of parameters of disaggregate behavioral models. It considers the calibration of the travel time parameter of the route choice model. Unlike the present paper, which addresses a demand calibration problem of dimension 4050, the work of Zhang *et al.* (2017) considers a one-dimensional problem. From a methodological perspective the main differences are as follows. First, Zhang *et al.* (2017) formulate one metamodel for each of the links with sensors, while the proposed approach considers a single metamodel for the entire network. As mentioned above, this reduces the number of metamodel parameters. Second, the analytical network model of Zhang *et al.* (2017) is based on a link model derived from probabilistic finite capacity queueing theory, while the proposed approach uses a differentiable approximation of the simulator's deterministic link fundamental diagram. Both Zhang *et al.* (2017) and the proposed approach consider an analytical network model with endogenous assignment. Both models are formulated as a system of nonlinear equations. Nonetheless, the implementation of Zhang *et al.* (2017) explicitly implements the link-to-link turning probabilities. Hence, the full network model is less scalable and less computationally efficient than the proposed approach.

As mentioned previously, we consider computationally efficient offline calibration algorithms to be essential components for the design of real-time calibration algorithms. For online calibration, we view the method of Zhang and Osorio (2017) and the proposed method as suitable building blocks for real-time algorithms. For networks with elaborate traffic dynamics, where assignment varies significantly within short time intervals (e.g., networks with dynamic congestion pricing or with real-time traffic management strategies), we expect the proposed method to be suitable (since it has endogenous assignment). For networks with simpler traffic dynamics, we expect the method of Zhang and Osorio (2017) to be a suitable choice.

3 Validation

We now validate the proposed method with a simple synthetic toy network. We use the network topology of Astarita *et al.* (2001) displayed in Figure 2. It consists of 3 origin nodes (labeled o_1, o_2, o_3), 3 destination nodes (labeled d_1, d_2, d_3) and 9 OD's (i.e., all origin-destination combinations are feasible). Hence, the OD calibration problem is a nine-dimensional problem. All links are one-way. The network mimics a multi-lane highway (top links) with on-ramps and off-ramps and a single-lane arterial (below horizontal links). The network is modeled as a set of 28 links and 43 lanes. We assume that all links have sensors. Note that even when all sensors have links, there are multiple OD matrices that can lead to the same link counts. In this section and in Section 4, we use the mesoscopic dynamic simulator DynaMIT (DYnamic Network Assignment for the Management of Information to Travelers) (Ben-Akiva *et al.* 2010). We first validate the analytical traffic model (Section 3.1). We then benchmark the performance of the proposed approach

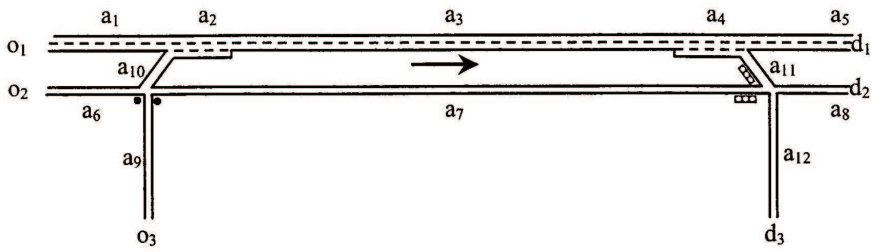


Figure 2: Synthetic simple network topology proposed by Astarita *et al.* (2001)

versus other optimization algorithms (Section 3.2).

3.1 Validation of the analytical traffic model

We consider three scenarios with increasing levels of demand. For a given demand level, we compute synthetic link counts via simulation, and assume these to be the "true" field counts (i.e., term y_i of Eq. (1)). Each element of the prior OD matrix (term $\tilde{d}$ of Eq. (1)) is obtained as the sum of the corresponding true OD value and a randomly drawn normal distributed error term with an expectation of zero and a standard deviation of 20% of the true OD value. Hereafter, when performing optimization, estimates of the simulation-based functions are obtained from a single simulation replication. Experiments based on multiple simulation replications were carried out on a subset of experiments, and led, for all methods, to the same trends than those presented hereafter. Nonetheless, the use of multiple replications may be warranted for other case studies.

Each plot of Figure 3 considers a given demand level. Each plot displays a one-dimensional cut of the simulation-based summation term of the objective function (i.e., first summation of Eq. (1)). The cut is obtained by assuming that all OD's have common value (i.e., all elements of the decision vector d have the same value). The x-axis displays the common OD value, and the y-axis displays the first summation term of the objective function. The blue circles represent the simulation-based estimates of objective function term. The red crosses represent the analytical approximations derived by the analytical traffic model, f_A of Eq. (5). For all three demand levels, the analytical model yields an accurate approximation of the simulation-based summation term of objective function.

Figure 4 compares in a single plot all points of the 3 plots of Figure 3. Each point is represented by a black circle. For each point, it considers the simulation-based term of the objective function (i.e., the first summation of Eq. (1)) and displays the analytical traffic model approximation along the x-axis (i.e., term f_A of Eq. (5)) and the corresponding simulation-based estimate along the y-axis. It also displays the diagonal line ($y = x$) in red. All points are along the diagonal. This indicates an accurate approximation of the simulation-based objective function provided by the analytical network model. Recall from Section 2.2 that the goal of the analytical traffic model (and of the metamodel) is to enable the SO algorithm to efficiently search in high-dimensional feasible regions. In other words, the role of the analytical traffic model is to identify subregions of the feasible region that have points with good simulation-based performance. Note that this can be achieved without the analytical traffic model providing an accurate (or even a positively

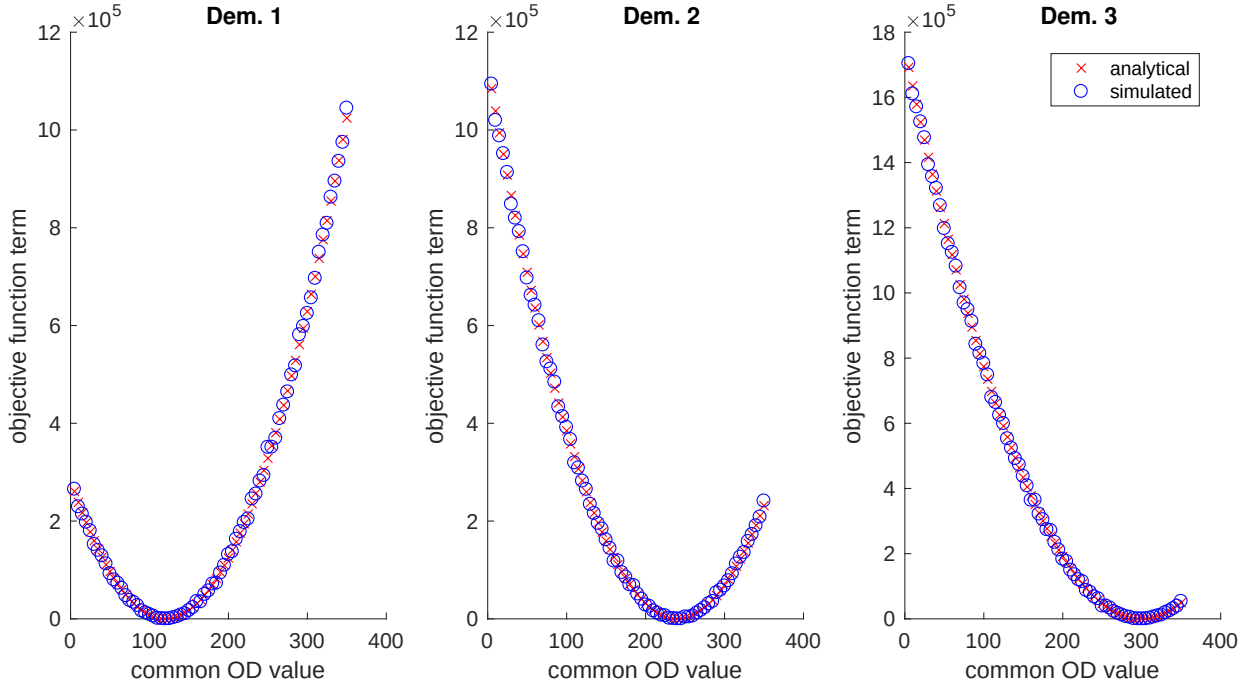


Figure 3: Comparison of the simulation-based estimates of a one-dimensional cut of the simulation-based term of the objective function and of the corresponding analytical approximations, considering three levels of demand

correlated) approximation of the simulation-based objective function.

We now evaluate the performance of the proposed algorithm for this simple synthetic network. We apply the SO algorithm for each of the three levels of demand mentioned above. For each demand, we consider 10 random initial points, which are uniformly drawn from the feasible region (Constraint (2)). For each initial point, we run the algorithm and terminate it once a total of 20 points have been simulated (i.e., the computational budget is 20).

Each plot of Figure 5 considers a demand level. Each plot displays 10 curves that correspond to the 10 algorithmic runs (i.e., based on 10 different initial points). The x-axis displays the number of simulated points, and the y-axis displays the simulation-based objective function estimate of the current iterate (i.e., the point considered to have best performance so far). The plots indicate that regardless of the initial point and of the level of demand, the proposed approach identifies points with good performance as of the second simulated point ($x = 2$). For all runs, the initial points have bad performance (i.e., high objective function estimates). There is also high variability across the performance of the initial points. Nonetheless, for all initial points the proposed approach identifies a point with good performance as of the second simulation. This shows the robustness of the proposed approach to the quality of the initial points.

Recall that at every iteration of the SO algorithm, the metamodel optimization problem is solved (Problem (3)-(12)). A solution to that problem is known as a trial point. Figure 6 considers the trial points of all the 30 SO runs (10 runs for each of the 3 demand levels). The figure compares, for each trial point, the objective function approximation provided

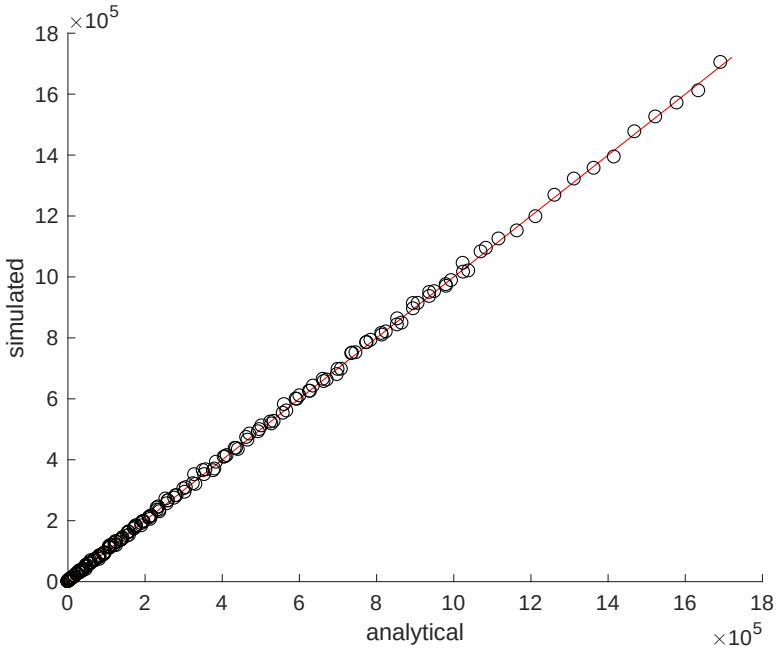


Figure 4: Comparison of the simulation-based estimates and the analytical approximations of a one-dimensional cut of the simulation-based term of the objective function

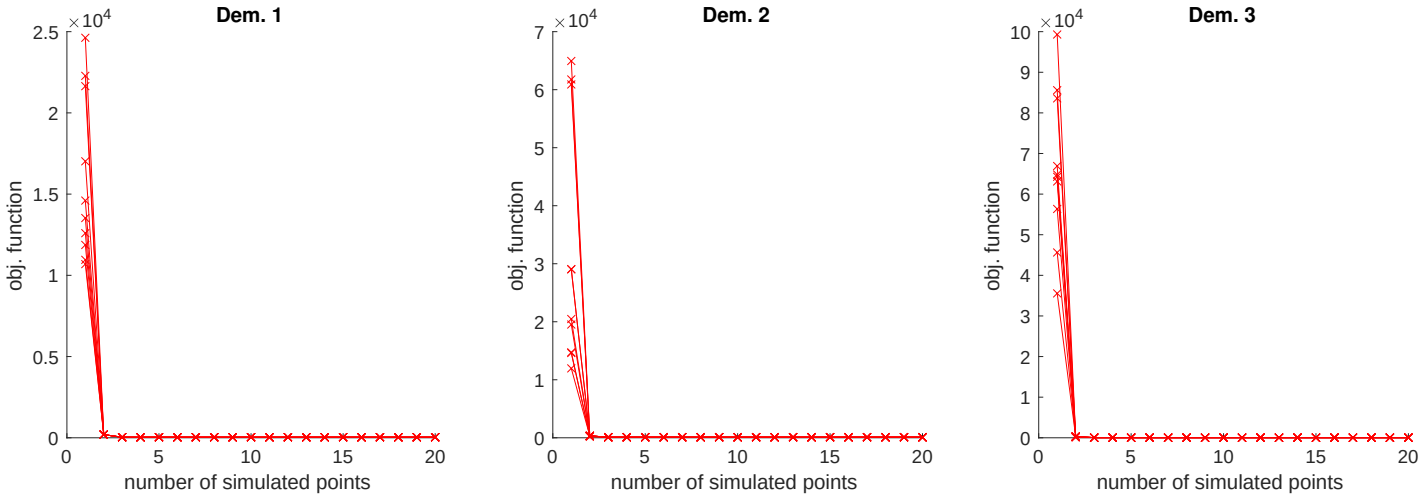


Figure 5: Performance of the SO algorithm as a function of the number of simulated points (i.e., the computational budget), for three demand levels

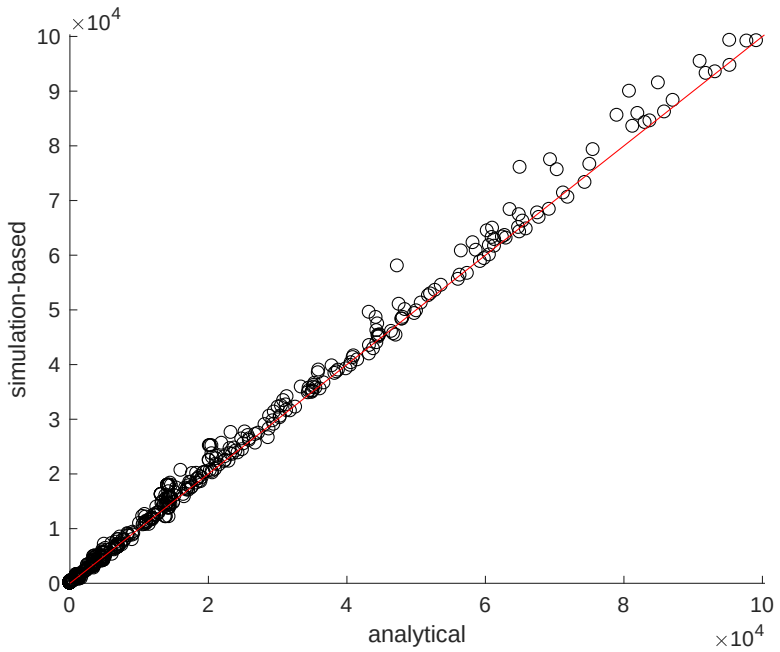


Figure 6: Comparison, for all the trial points of all the SO runs, of the simulation-based estimates of the objective function to the analytical approximations of the objective function derived by the analytical traffic model

by the analytical traffic model (f_A of Eq. (5)) along the x-axis to the simulation-based objective function along the y-axis. The trial points are represented as black circles. The diagonal line ($y = x$) is displayed in red. This figure illustrates the accuracy of the approximations derived by the analytical traffic model.

Each plot of Figure 7 considers the 10 solutions of a given demand level. Each blue cross represents the counts for a given solution and a given link with a sensor. The x-axis displays the "true" (synthetic) counts (term y_i of Eq. (1)) and the y-axis displays the simulated estimate for the corresponding OD solution (i.e., this is an estimate of the term $E[F_i(d, u_1; u_2)]$ of Eq. (1)). The diagonal line is displayed as a dashed red line. Note that each plot contains a total of 28*10 points (for each of the 10 solutions there are 28 links with sensors), yet there are typically 4 points visible in the plots. This is because many links have common counts. Hence, their corresponding points overlap in the plots. This figure indicates that for all 10 runs of all 3 demand levels, the solutions proposed by the algorithm accurately replicate the true counts for all links with sensors.

3.2 Validation versus benchmark methods

We now benchmark the performance of the proposed approach to that of two other methods: a derivative-free generalized pattern search algorithm (GPS) (Mathworks, Inc. 2016), and the Stochastic Perturbation Simultaneous Approximation (SPSA) algorithm (Spall 1992, 2003). The algorithmic parameters of the GPS are set based on standard guidelines as well as on its performance on toy network experiments. The SPSA parameters are set based on standard guidelines Spall (2003, Chap. 7) and based on insights from its past

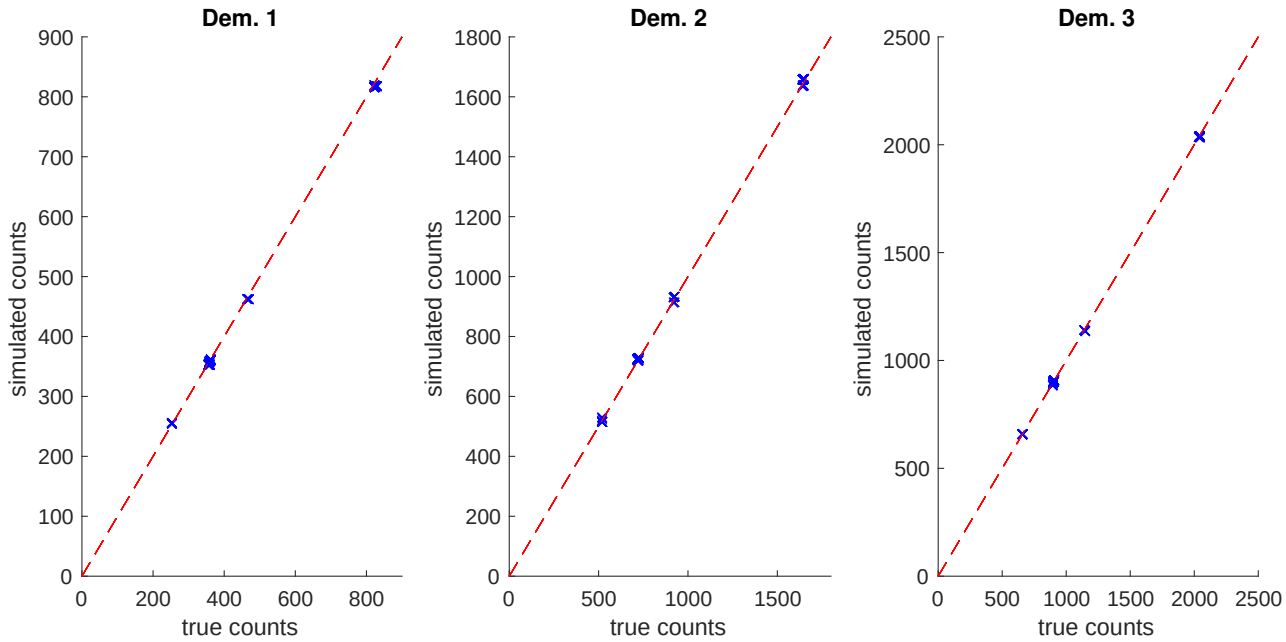


Figure 7: Comparison of the "true" link counts to the simulation-based count estimates obtained from the proposed OD solutions, for each of the 10 SO runs and 3 demand levels

use for calibration by MIT's ITS (Intelligent Transportation Systems) Lab members.

We consider here a broader set of demand scenarios in order to account for both time-varying demand conditions (e.g., transitions from uncongested to congested conditions and vice versa), as well transient traffic conditions. We consider three demand levels with low (denoted L), medium (denoted M) and high (denoted H) congestion levels, respectively. For each experiment, we define one demand level (denoted D_1) for 7-8am and a second demand level (denoted D_2) for 8-9am. This leads to a total of 9 demand scenarios, defined as $(D_1, D_2) \in \{L, M, H\}^2$. In other words, the demand scenario s result from all pairwise combinations of demands $\{L, M, H\}$. The simulated counts are estimated based on traffic conditions observed from 8:30-9am. When performing optimization, we assume D_1 is known and fixed, while D_2 is assumed unknown (it defines the decision vector).

For a given demand scenario (D_1, D_2), we proceed as in Section 3.1, we compute synthetic link counts via simulation, and assume these to be the "true" field counts (i.e., term y_i of Eq. (1)). We compute the prior OD matrix following the same procedure as in Section 3.1. We proceed as in Section 3.1 and consider a set of 10 different initial points, uniformly randomly drawn from the feasible region. For each of the 9 demand scenarios and each of the 3 methods (proposed, GPS and SPSA), we run it 10 times, once for each initial point. We terminate the method once a total of 20 points (i.e., 20 OD matrices) have been simulated. In other words, the computational budget is set to 20.

Figures 8, 9 and 10, consider the performance of the proposed, GPS and SPSA methods, respectively. Each figure contains 9 plots, one for each demand scenario. The top (resp. middle and bottom) row plots consider scenarios where the first demand D_1 is set to L (resp. M and H). The left-most (resp. middle and right-most) column plots consider scenarios where the second demand D_2 is set to L (resp. M and H). Each plot displays, for all links with sensors, the "true" synthetic counts along the x-axis and the counts of the OD matrix derived by the corresponding algorithm along the y-axis. Each plot displays the performance of 10 OD solutions, one for each initial point.

Figure 8 considers the proposed method. It shows that for all initial points and all 9 demand scenarios (i.e., all plots), the derived OD's yield an accurate fit to the true counts. Figure 9 considers the GPS method. It shows that for a given demand scenario (i.e., a given plot) the performance of the method varies across initial points. This holds for all 9 demand scenarios. The GPS solutions do not accurately fit the counts. Figure 9 considers the SPSA method. The same conclusions as for GPS hold.

Figure 11 evaluates the performance of the methods as a function of the computational budget. Just as for the previous figures, each plot considers one of the 9 demand scenarios. Each plot displays the performance of the proposed method (red solid lines with crosses), GPS (black dash-dotted lines with crosses) and SPSA (blue dashed lines with circles). For each method there are 10 lines, one for each initial point. Each line displays the estimate of the objective function of the current iterate as a function of the total number of simulated points. Each plot has a logarithmic scale. All plots have the following similar trends. GPS and SPSA have similar performance for all demand scenarios. The proposed method outperforms GPS and SPSA by 1 to 3 orders of magnitude. For a given demand scenario (i.e., a given plot) the performance of GPS and of SPSA varies by up to 1 order of magnitude, this shows their sensitivity to the performance of the initial point. The proposed method has a performance that is mostly insensitive to that of the initial points. The highest sensitivity is observed in the plot of the top row and middle column. For all other plots, the performance of the proposed method has similar performance across

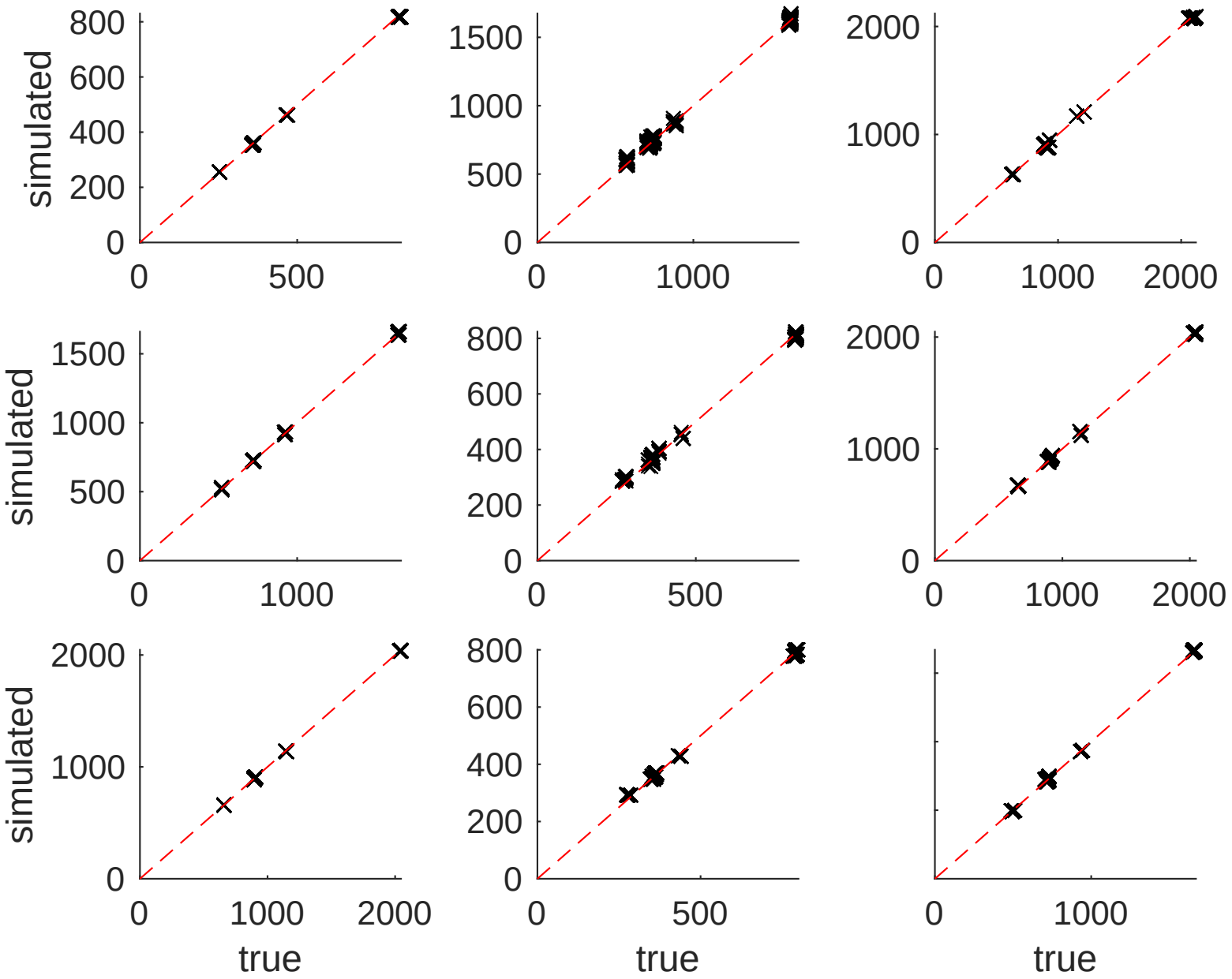


Figure 8: Fit to counts of the solutions derived by the proposed method, for all 9 demand scenarios (one for each plot) and all 10 initial points

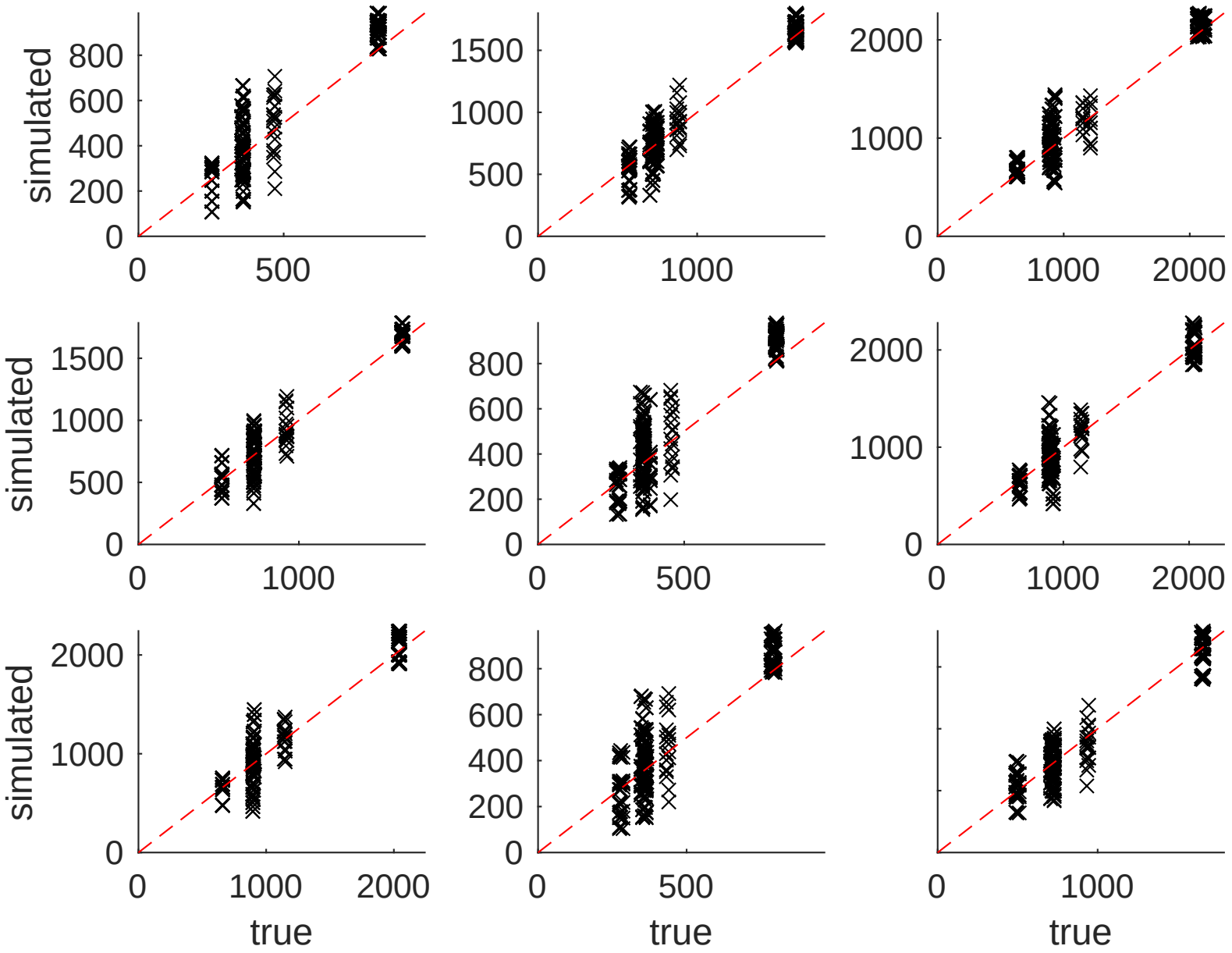


Figure 9: Fit to counts of the solutions derived by the GPS method, for all 9 demand scenarios (one for each plot) and all 10 initial points

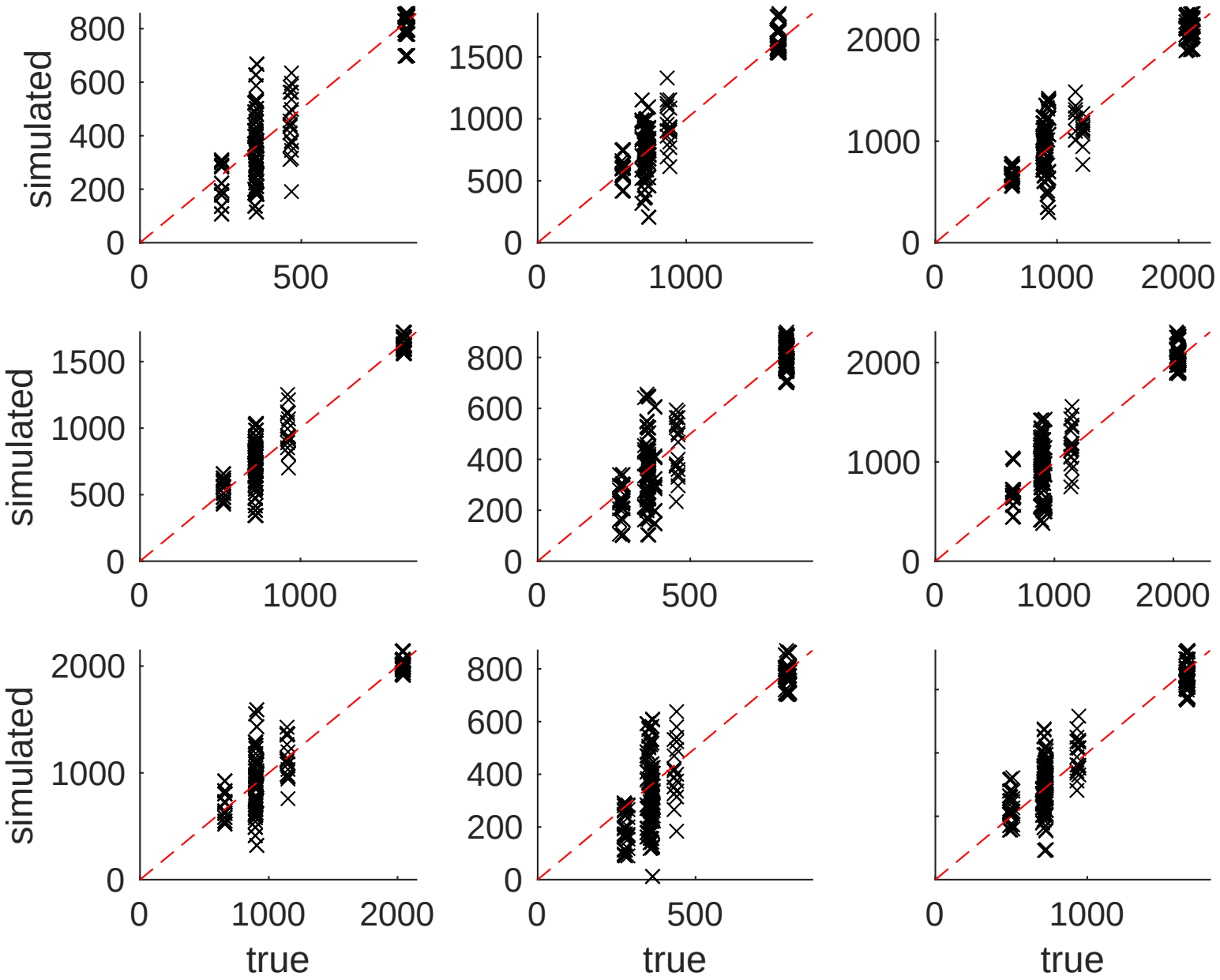


Figure 10: Fit to counts of the solutions derived by the SPSA method, for all 9 demand scenarios (one for each plot) and all 10 initial points

initial points. This illustrates its robustness to the quality of the initial points.

We now evaluate the performance of the methods as a function of the weight given to the prior OD information. We consider six different values of the weight parameter δ of Eq. (2), $\delta \in \{10, 1, 0.1, 0.01, 0.001, 0.0001\}$. Note that the above experiments were carried out with δ set to 0.01. For each value of δ, we run each method for a single (and common) initial point and for all 9 demand scenarios. As before, we terminate each method once the computational budget of 20 simulation evaluations is depleted The left (resp. middle and right) plot of Figure 12 consider the performance of the proposed (resp. GPS and SPSA) method. Each plot considers the performance of 54 solutions (i.e., each combination of 6 δ values and 9 demand scenarios). For each solution, the counts over all links are displayed. As before, the x-axis displays the true (synthetic) counts and the y-axis displays the simulated counts derived through optimization. Note that when disaggregating these results to create one plot for each method and for each of the 6 δ values, the fit to counts presents very similar trends. In other words, the performance of a method did not vary for different δ values. This holds for all methods. This is why we have grouped all 6 solutions in a single plot. The figure indicates that the proposed method slightly outperforms GPS and SPSA. This is systematically the case across δ values.

Figures 13 evaluates the performance of the methods as a function of the computational budget. Just as for the previous figures, each plot considers one of the 6 δ values. The highest δ value is that of the top-left plot, values decrease across columns for the first row of plots, followed by smaller values that decrease across columns for the second row of plots. Each plot displays the performance of the proposed method (red solid lines with crosses), GPS (black dash-dotted lines with crosses) and SPSA (blue dashed lines with circles). For each method there are 9 lines, one for each demand scenario. Note that each combination of demand scenario and δ value defines a different OD calibration problem. Thus, for a given method, the different runs need not have similar performance. As before, each line displays the estimate of the objective function of the current iterate as a function of the total number of simulated points. Each plot has a logarithmic scale along the y-axis. All plots have the following similar trends. GPS and SPSA have similar performance for most δ values, with SPSA slightly outperforming GPS for 3 of the 6 δ values (plots in the top-right, top-left and bottom-middle). The proposed method outperforms GPS and SPSA by 1 to 4 orders of magnitude.

We now evaluate the performance of the methods as a function of the bias in the prior OD. For a given true OD value, denoted d_z^*, the corresponding prior OD value is sampled from a normal distribution with expectation sd_z^* and with standard deviation $0.2sd_z^*$. The scalar s denotes the scaling factor of the prior OD matrix. We consider three types of experiments with $s \in \{1, 0.7, 1.3\}$. Note that all past experiments in this section have considered an unbiased prior (i.e., $s = 1$). For each scaling factor value, we run each method 10 times with 10 different initial points and terminate the run once the computational budget of 20 simulation evaluations is depleted. All experiments consider the same demand scenario, which was defined previously as having initially medium levels of congestion followed by high levels of congestion (i.e., $(D_1, D_2) = (M, H)$).

The left (resp. middle and right) plot of Figure 14 consider the performance of the proposed (resp. GPS and SPSA) method. Each plot considers the performance of 30 solutions (i.e., for each of the 3 s values we perform 10 SO runs). Each plot has the same layout as those of Figure 12. Note that when disaggregating these results to create one plot for each method and for each of the 3 s values, the fit to counts presents very similar

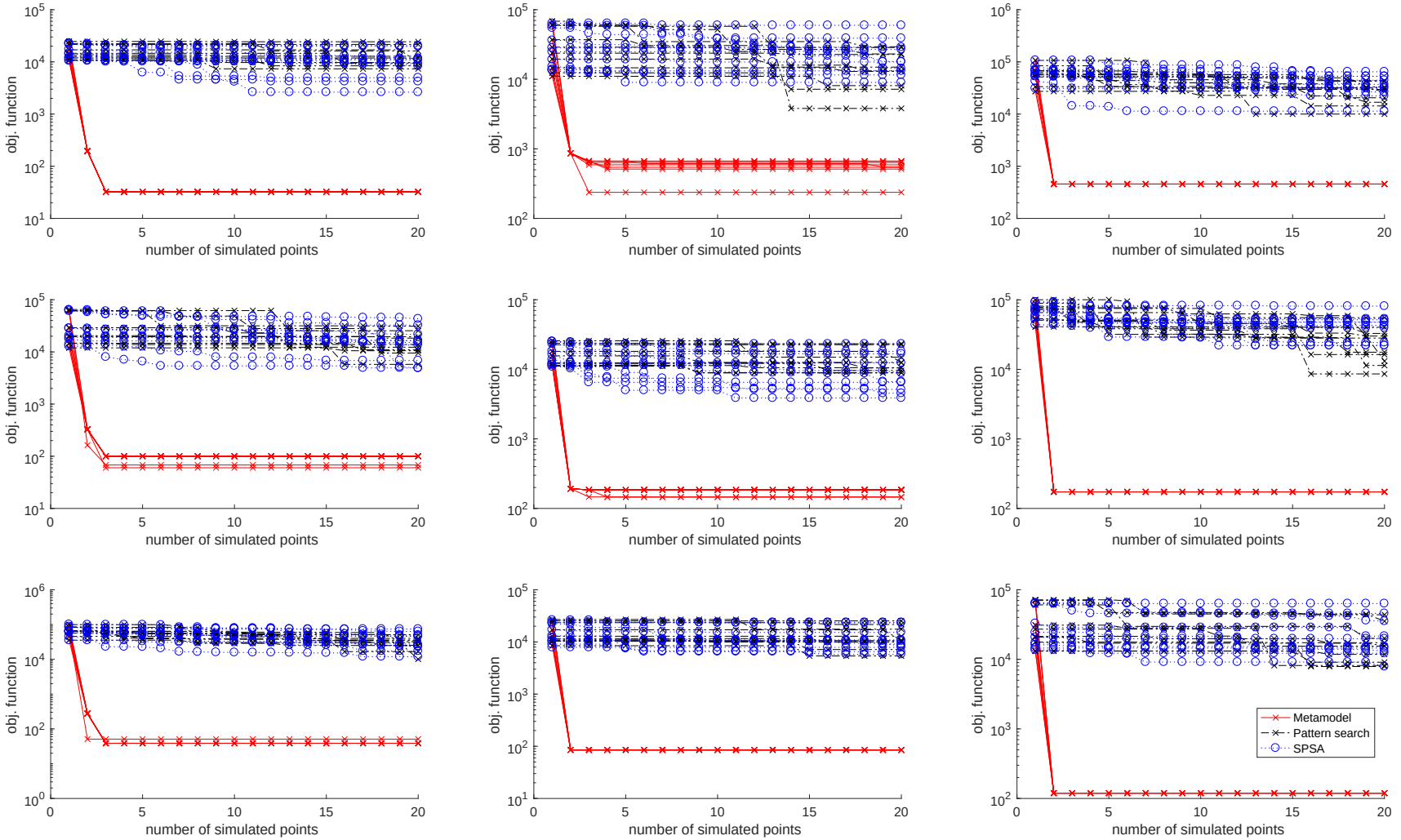


Figure 11: Performance of each method as a function of the number of simulated points, for all 9 demand scenarios (one for each plot) and all 10 initial points

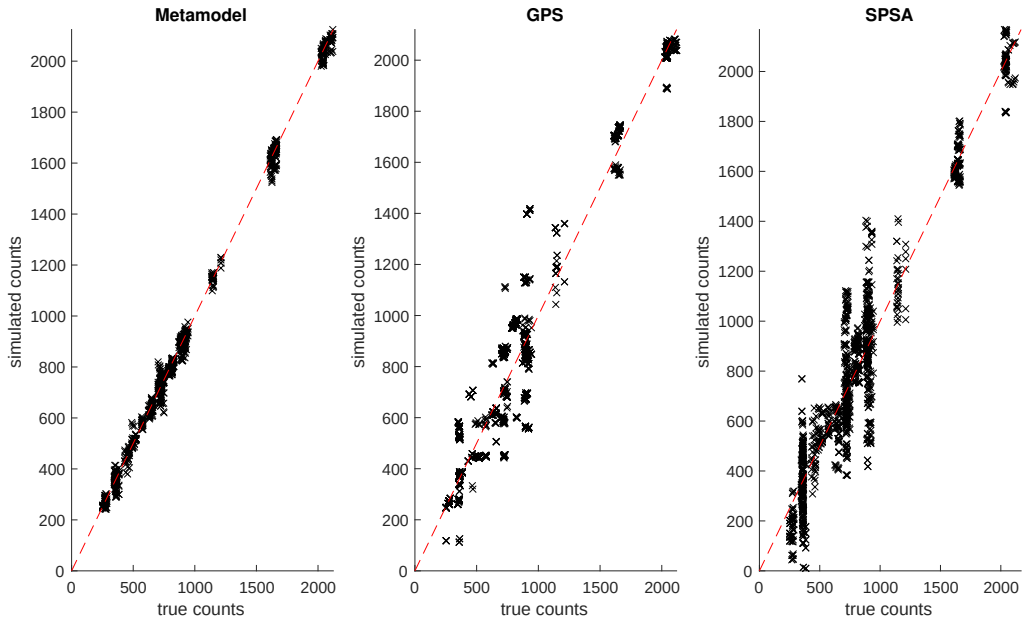


Figure 12: Fit to counts of the solutions derived by the proposed method, GPS and SPSA, considering all combinations of the 6 δ values and the 9 demand scenarios.

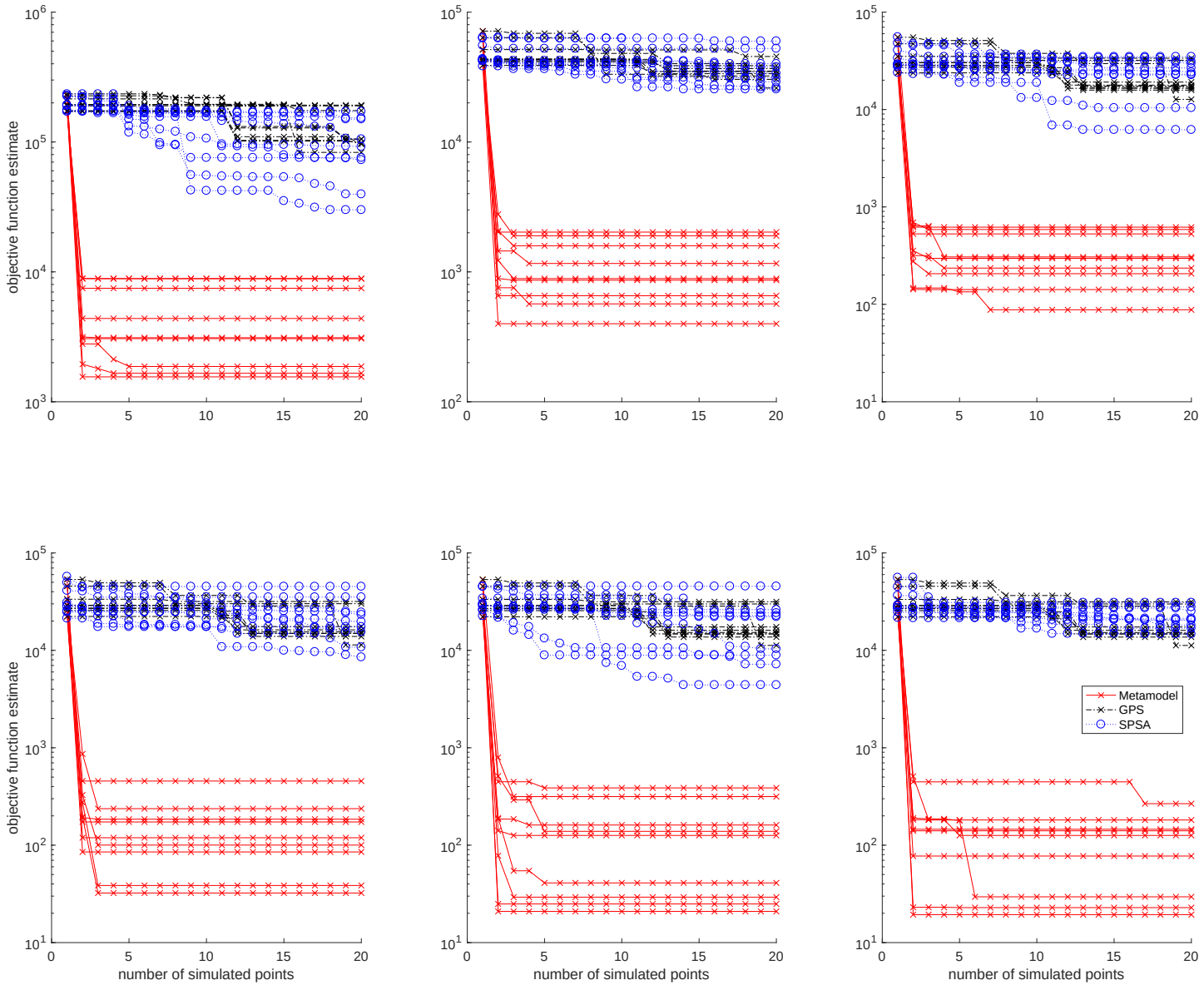


Figure 13: Performance of each method as a function of the number of simulated points, for all 6 δ values (one for each plot) and all 9 demand scenarios

trends. In other words, the performance of a method did not vary for different s values. This holds for all methods. This is why we have grouped the results of the 3 experiments in a single plot. The figure indicates that GPS and SPSA have similar performance, and are outperformed by the proposed method. Again, these observations across all s values.

Figure 15 evaluates the performance of the methods as a function of the computational budget. The top (resp. middle and bottom) plot considers s set to 1 (resp. 0.7 and 1.3). Each plot displays the performance of the proposed method (red solid lines with crosses), GPS (black dash-dotted lines with crosses) and SPSA (blue dashed lines with circles). For each method there are 10 lines, one for each initial point. As before, each line displays the estimate of the objective function of the current iterate as a function of the total number of simulated points. Each plot has a logarithmic scale along the y-axis. All plots have the following similar trends: (i) GPS and SPSA have similar performance, (ii) the proposed method outperforms GPS and SPSA by 2 orders of magnitude.

4 Singapore case study

We now evaluate the performance of the proposed approach with a large-scale network model. We consider the network of major arterials and expressways of Singapore. Figure 16a displays a map of the network, with the location of the tolls indicated in red, the expressway links represented in orange and the major arterials represented in yellow. The corresponding network model is displayed below the map. The network model is defined by 1150 links, over 2300 lanes, 4050 OD pairs with over 18000 routes. Of the 1150 links of the simulator, only 860 are modeled by the analytical network model. These are the links that are defined in the pre-determined route choice set.

We calibrate demand for all 4050 OD pairs (i.e., the dimension of the decision vector is 4050). We consider a single fixed OD matrix for a weekday 7:00-7:30am. We use 6-7am as the warm-up period. More specifically, when simulating the performance of a given OD matrix, we use it to define the demand for 6:00-7:30am, yet only extract simulation statistics for the 7:00-7:30am period. For this Singapore network, we do not have access to field measurements (terms y_i of Eq. (1)). Hence, we use synthetic simulated data to estimate the field measurements. More specifically, we use a pre-calibrated OD matrix as the "true" OD matrix. We embed it within the simulator and extract link counts on a set of 172 links (i.e., 15% of the links have measurements). These synthetic link counts are assumed to be field counts. In other words, they are used to estimate the terms y_i of Eq. (1). A given element of the prior OD matrix ($\tilde{d}_z$ of Eq. (1)) is defined as the corresponding "true" OD value plus a random normally distributed error term with expectation 0 and standard deviation 20% of its value. We set the upper bound ($d^{\max}$ of Eq. (2)) to 2000 vehicles per hour.

As before, we compare our proposed approach with GPS and SPSA. Their algorithmic parameters are fine-tuned as described in the previous Section. Recall that unlike the proposed method, these benchmark methods are not designed to be used under tight computational budgets. Instead, they are designed based on asymptotic performance properties. As mentioned in Section 1, there are no SO methods that are designed to address high-dimensional problems under tight computational budgets. This is precisely the literature gap that this paper contributes to.

We consider 10 different initial points. The initial points are uniformly drawn from the feasible region (Eq. (2)) and such that the total OD demand is equal to that of the prior

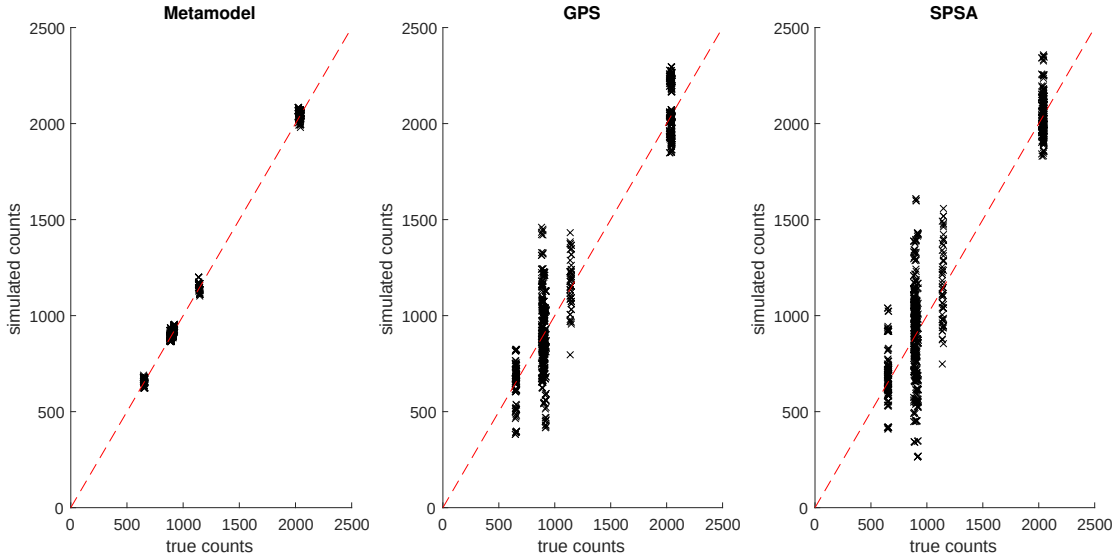


Figure 14: Fit to counts of the solutions derived by the proposed method, GPS and SPSA, for all 3 scaling factors of the expectation of the prior OD and 10 initial points

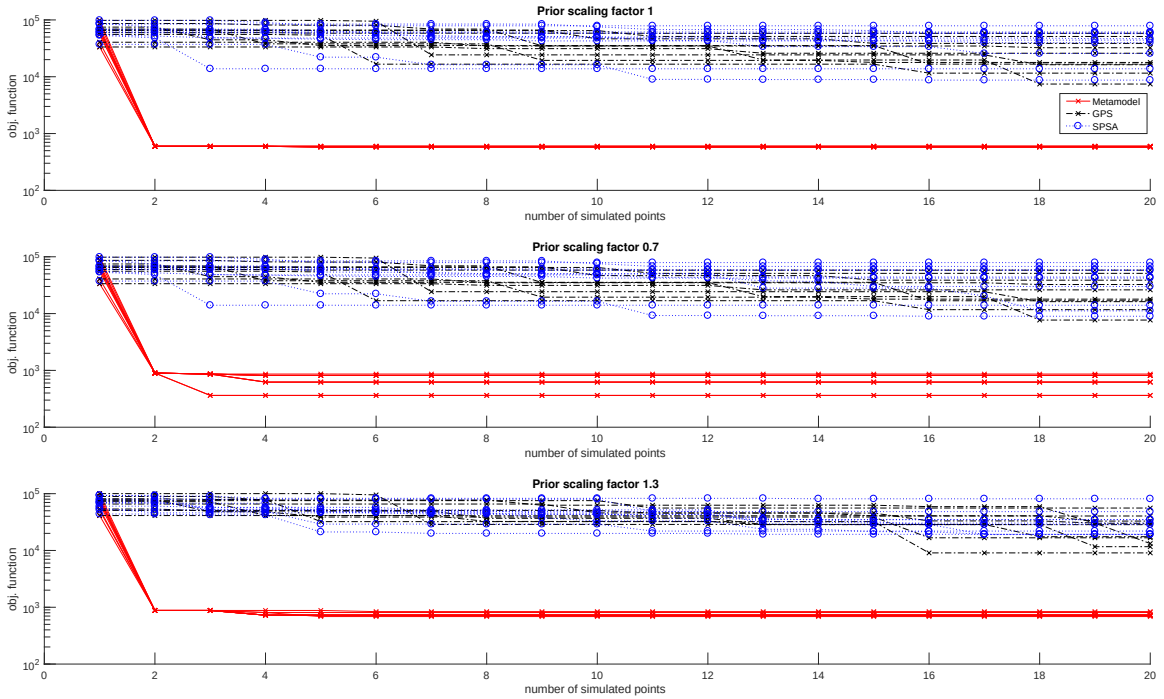
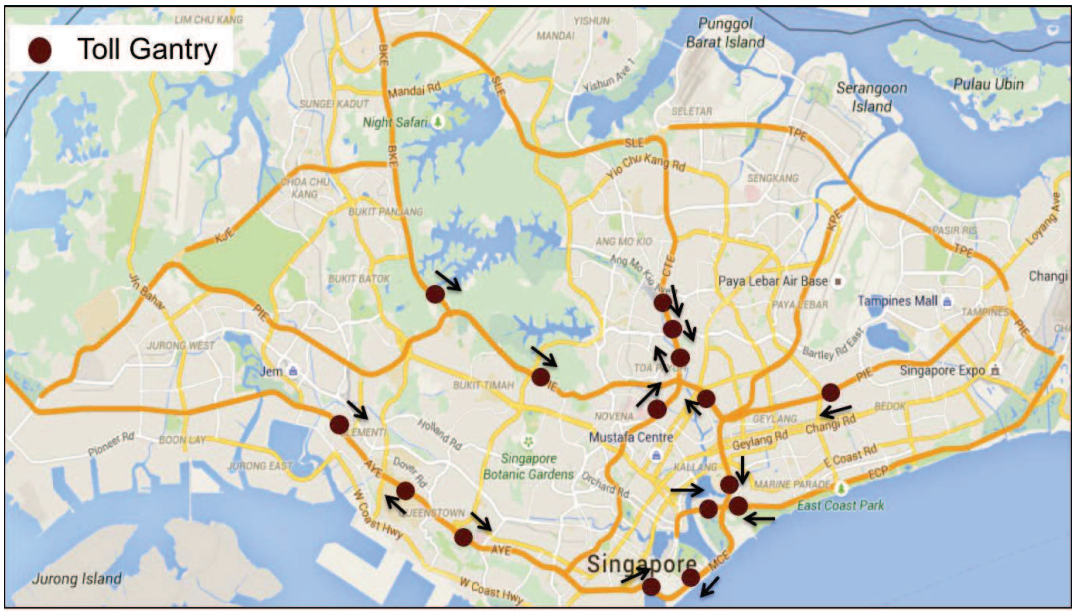
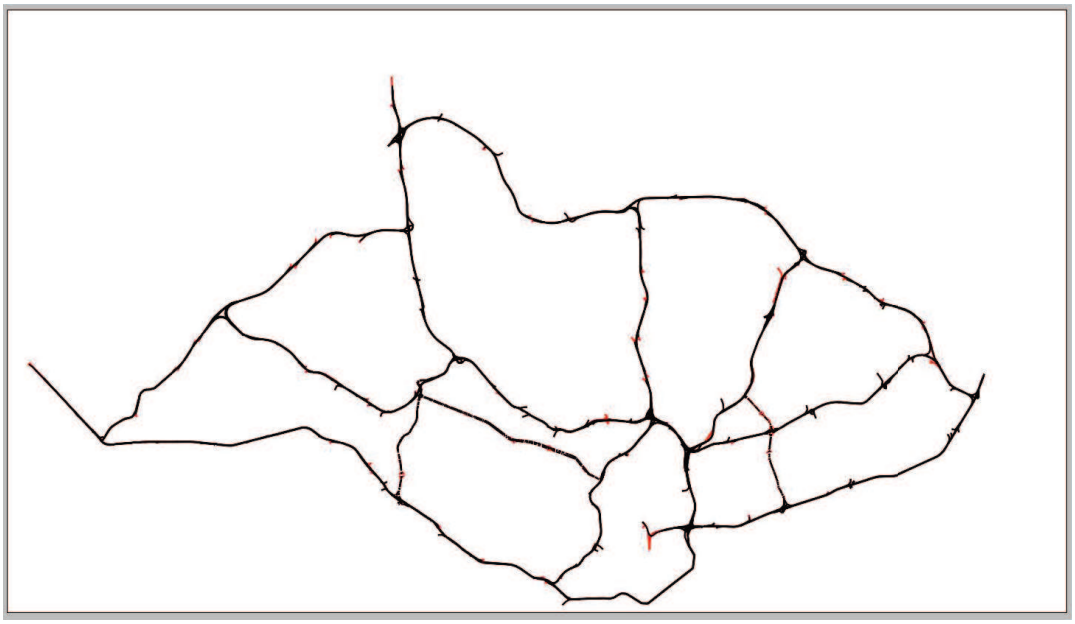


Figure 15: Performance of each method as a function of the number of simulated points, for all 3 values of the scaling factor of the prior OD expectation (one for each plot) and all 10 initial points



(a) Singapore expressway network (map data: Google Maps (2017))



(b) Simulation network model

Figure 16: Singapore network

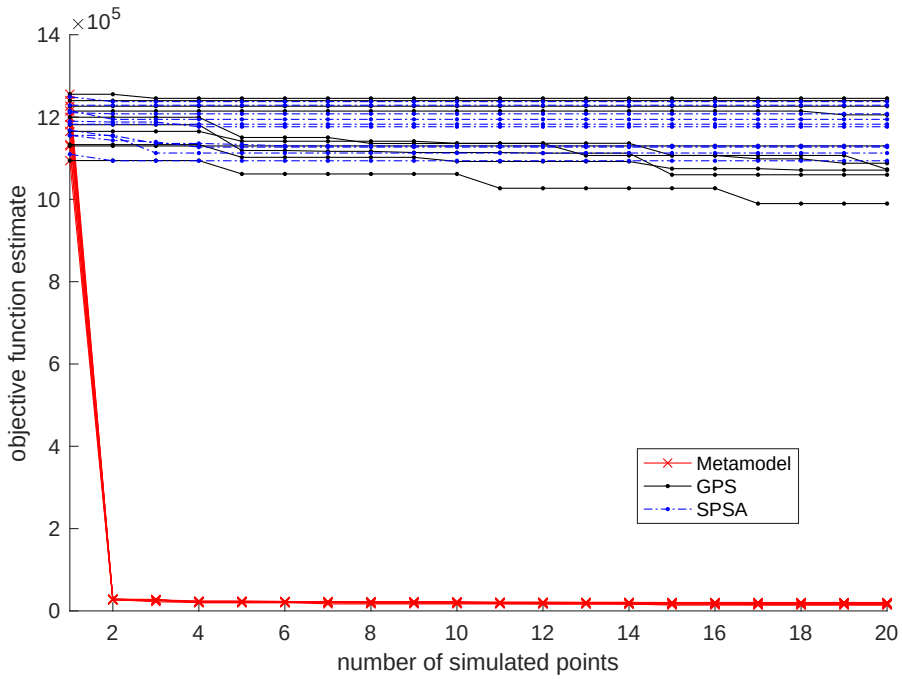


Figure 17: Objective function estimate of the current iterate as a function of the number of simulated points

OD. For each initial point, we run each method and terminate it once 20 simulations have been evaluated. In other words, the computational budget is 20.

Figure 17 displays for each algorithmic run, the objective function of the current iterate as a function of the number of simulation points. There are 10 solid red (resp. solid black and dashed blue) lines that correspond to the 10 runs of the proposed (resp. GPS and SPSA) approach. Recall that all methods share the same set of 10 initial points. Almost all lines of the metamodel approach overlap. Once the second point is simulated, they all have a significant improvement in the objective function estimate. More specifically, the improvement is of 2 orders of magnitude (the average initial point performance is $1.18e6$ and the average performance of the second simulated point is $2.8e4$). This indicates the added value of the analytical structural information provided by the analytical model. In particular, for all initial points the proposed method yields a current iterate at sample size 2 with similar performance. This shows how the analytical structural information of the analytical network model enables the method to become robust to the quality of the initial points.

On the other hand, both benchmark methods (GPS and SPSA) gradually identify points with improved performance. Nonetheless, none of the runs yield a significant improvement compared to the proposed approach. Upon depletion of the simulation budget (i.e., $x = 20$), the solutions of the GPS method yield an average improvement of 4.4% compared to the initial solutions (the average initial point performance is $1.18e6$ and the average performance of the final GPS solutions is $1.13e6$). For SPSA, the average improvement is 1.4% compared to the initial solutions ($1.18e6$ versus $1.17e6$).

Figure 18 presents, in more detail, the performance of the proposed method. It displays the same 10 red lines of Figure 17 yet omits the initial point estimates (i.e., it starts at

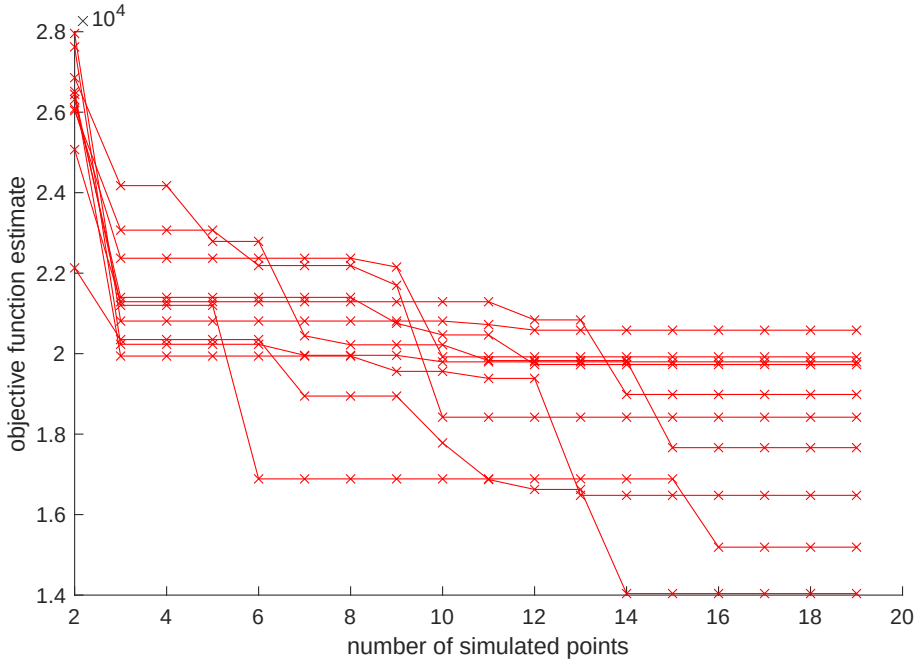


Figure 18: Objective function estimate of the current iterate as a function of the number of simulated points excluding the initial point simulations (i.e., x-axis starts at 2)

$x = 2$). This figure illustrates that, after the second simulation, the proposed method continues to yield points with improved performance. The average performance improvement between $x = 2$ and $x = 20$ is 35% (i.e., $2.77e4$ versus $1.80e4$). While Figure 17 indicated that, for all initial points, the metamodel proposed solutions with performance estimates that have the same order of magnitude, this figure shows that there is variability in the performance. This variability can be can be due to the initial points. It can also be due to the stochasticity of the simulator or to the stochasticity of the sampling strategy of the algorithm (i.e., Step 2b of Figure 1 involves random sampling).

We now compare the performance of the methods in terms of computation time. The computation times include all computations, not just simulation times. In particular, for the proposed method the computation times account for the time needed to solve the metamodel optimization problem. Note that the various computations are carried out on several servers. For a given server, the allocation of CPU resources for a job can vary over time because it depends on what other jobs are running simultaneously. Hence, these computation statistics can give a general idea of runtimes, yet their differences cannot be solely attributed to algorithmic performance. Figure 19 displays for each algorithmic run, the objective function of the current iterate as a function of the computation time. As before, the 10 solid red (resp. solid black and dashed blue) lines correspond to the 10 runs of the proposed (resp. GPS and SPSA) approach.

To put these computation times into context, note that one simulation run (i.e., the evaluation of the simulator) takes approximately 2.5 hours, hence the computation of 20 sequential simulations is in the order of 2.1 days. This high computation time of a single simulation illustrates the need for computationally efficient algorithms, i.e., algorithms that can identify points with good performance within few simulation runs. This figure

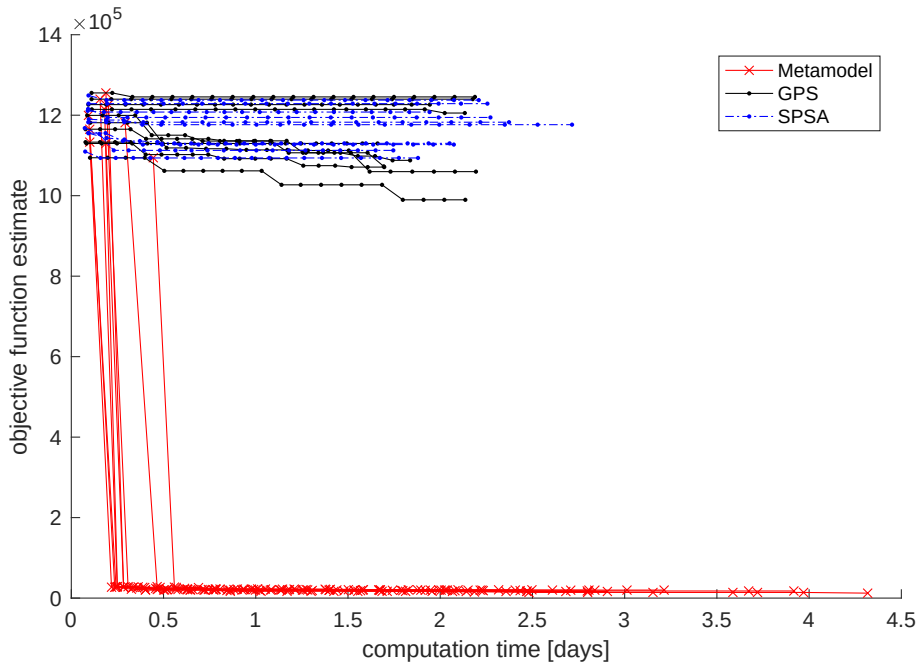


Figure 19: Objective function estimate of the current iterate as a function of the computation time

indicates that when comparing the performance as a function of computation time, the proposed method also outperforms the benchmark methods. The average computation time needed to deplete the computational budget (20 points) is 2.8 days for the proposed method versus 2.0 days for GPS and 2.2 days for SPSA. The longest computation time needed to deplete the computational budget is 4.3 days for the proposed method, 2.2 days for GPS and 2.7 days for SPSA.

Figure 20 is similar to Figure 19, yet it zooms into the performance of the proposed method. It displays the performance of the proposed method excluding the initial points (i.e., these are accounted for in the total computation time, but the corresponding initial objective function is not displayed). This figure illustrates that even after the initial significant improvement in the objective function (which is displayed in Figure 19), the proposed method gradually identifies points with further improved performance.

Figure 21 compares the performance of the methods in terms of their ability to replicate the link counts. Each plot of Figure 21 compares the "true" counts (these are the y_i values of Eq. (1)) to the estimated simulation-based link counts of a given solution (these are estimates of the terms $E[F_i]$ of Eq. (1)). The 10 plots in the top (resp. middle and bottom) row correspond to the 10 solutions obtained by the proposed (resp. GPS and SPSA) method for each of the 10 initial points and allowing for a computational budget of 20 simulations. Each column of plots corresponds to the solutions obtained for a given initial point. Each plot also displays the diagonal axis (function $y = x$) as a red dashed line. For a given solution, the closer the distance between the "true" counts and the simulated counts (i.e., the closer the points are to the diagonal axis), the better the performance of the solution in terms of fit to the counts. For a given row, all plots have the same y-axis limits, yet those across rows differ. For a given row, only the y-axis limits of the

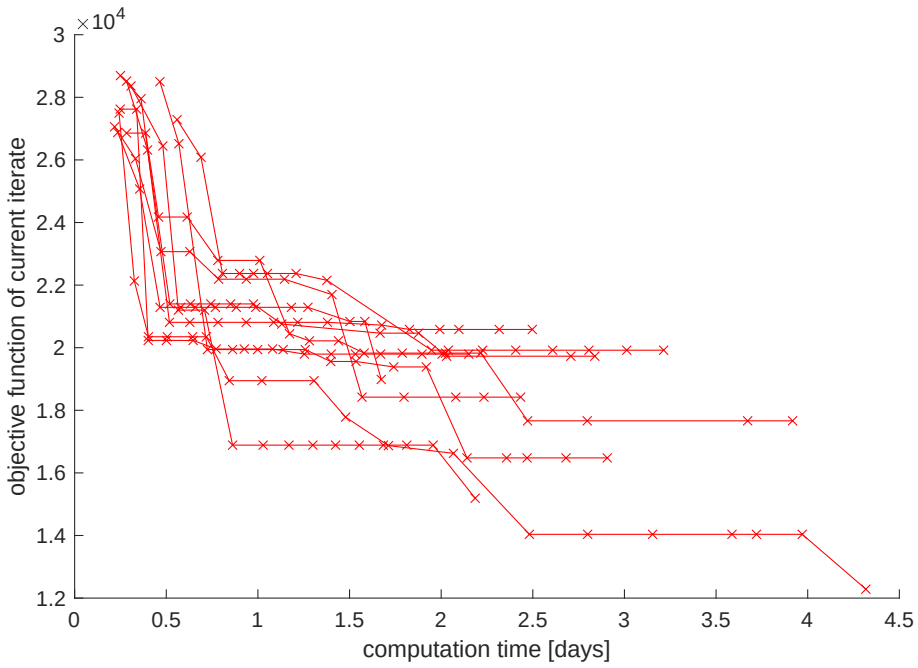


Figure 20: Objective function estimate of the current iterate as a function of the computation time excluding the initial point simulations

left-most plot is displayed, this improves the legibility of the plot. These figures show that all 10 solutions derived by the proposed method have good performance. The similarity in performance of the solutions derived by the proposed method illustrates the robustness of the method to the quality of the initial point. Both benchmark methods fail to identify solutions with good performance within the tight computational budget.

Figure 22 evaluates the ability of each method to recover the "true" OD. From left to right, the four plots of Figure 22 consider the performance of the metamodel approach, GPS, SPSA and the initial ODs. For each plot, the x-axis displays the true OD value, and the y-axis displays the proposed OD value. Each plot displays the 10 OD values proposed by each method (there is one for each of the 10 initial points). For the right-most plot, the y-axis displays the 10 initial points used to initialize the algorithms. The plots also display a red diagonal line with equation $y = x$. The closer the points are to the diagonal line, the better the solution. The main insights from these plots are as follows. None of the methods yield accurate ODs. Given the tight computational budget (20 simulation observations for a problem of dimension 4050) this is not surprising. The conclusions from Figure 21 were that the proposed approach was capable of identifying solutions with good fit to counts, while the left-most plot of Figure 22 indicates that it does not accurately identify the true OD. As discussed before, the OD calibration problem is underdetermined, hence the ability of an OD to replicate traffic counts does not indicate that the proposed OD is similar to the true OD. The second (from left to right), third and fourth plots of Figure 22 are similar. This highlights how the algorithms that treat the simulator as a black-box (GPS and SPSA) are not designed to perform well under tight computational budgets and remain close to the initial ODs. As part of ongoing work, we are studying how the use of higher-resolution mobility data can mitigate this underdetermination issue,

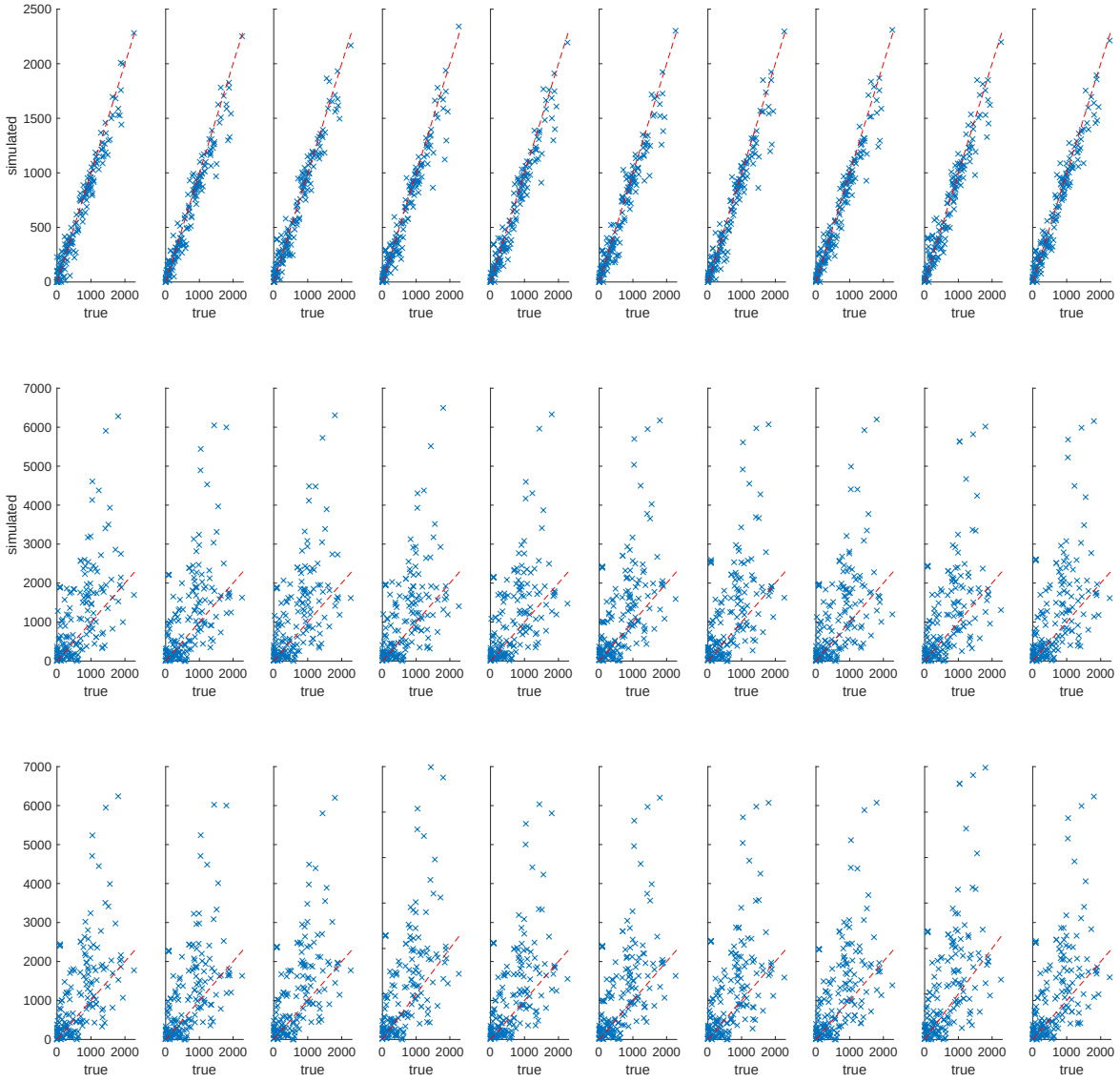


Figure 21: Comparison of the simulated counts of the proposed solutions to the "true" counts, for each method (i.e., each row) and each initial point (i.e., each column)

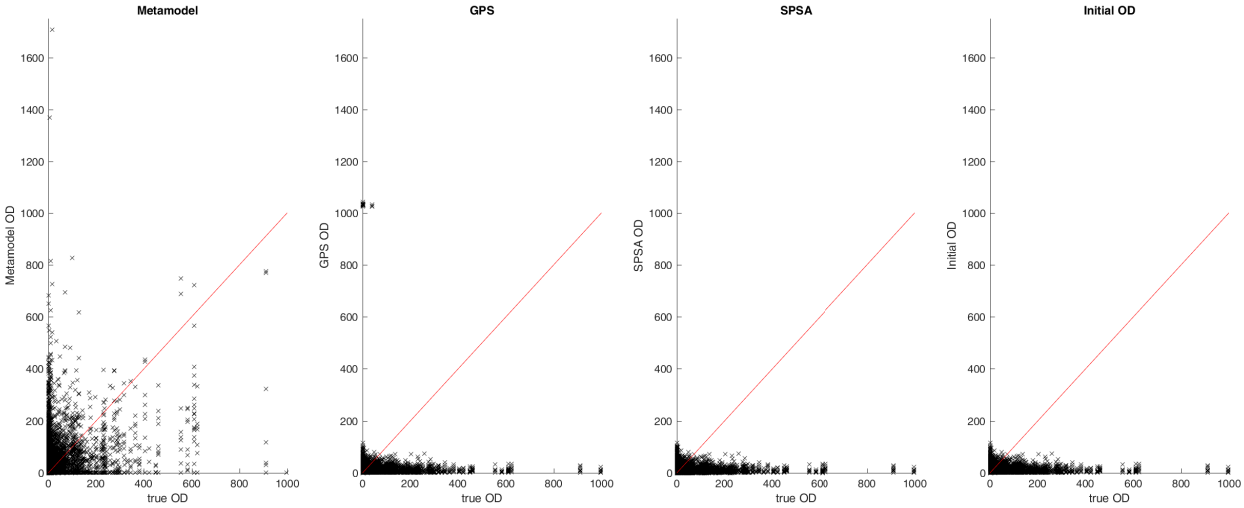


Figure 22: Comparison of the OD values of the proposed solutions to the "true" values. From left to right, the plots consider the proposed method, GPS, SPSA and the initial ODs

while preserving the computational efficiency of the proposed calibration algorithm.

Given that the computational budget is depleted faster for the benchmark methods than for the proposed method, Figure 23 allows the benchmark methods to compute a larger set of simulation points, such that the total computation time exceeds that of the proposed method. The red lines are the same as those of Figure 19. In other words, additional computational resources are not given to the proposed method. The benchmark methods are allowed to compute for an average of 7 days (compared to the average 2.8 days of the proposed method). This figure indicates that even when allowing for more than double the computation time of the proposed method, the benchmark methods are outperformed by the proposed method by almost 2 orders of magnitude.

The congestion levels in this Singapore case study vary depending on the OD value considered. The main trends that hold over all points (i.e., all simulated ODs) is that average congestion levels (computed as averages across links) gradually increase throughout the 7:00-7:30am time period. During this period, transient traffic conditions are observed and congestion is not uniformly distributed across space: typically a small set of links are highly congested. To give an order of magnitude, considering an SPSA solution, the average (across links) density increases from 48 veh/mile at 7am to 64 veh/mile at 7:30am. The percentage of links that have unstable traffic conditions (defined as densities above 30 veh/mile) varies from 20% to 23% and the percentage of links with high levels of density (defined as above 67 veh/mile) increases from 13% at 7am to 18% at 7:30pm.

5 Conclusions

This paper proposes a metamodel simulation-based optimization (SO) approach for offline calibration problems that are high-dimensional and consider large-scale simulation-based network models. The main component of the metamodel is an analytical network model.

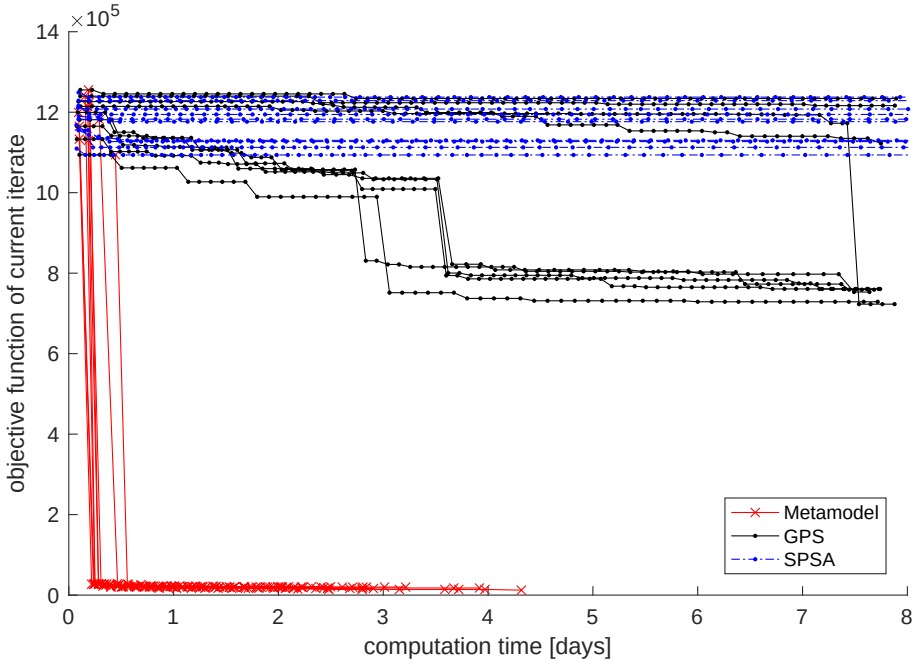


Figure 23: Objective function estimate of the current iterate as a function of the computation time

The latter is specified as a nonlinear system of equations, the dimension of which scales linearly with the number of links in the network. It is tractable: it can be efficiently evaluated with standard solvers for systems of equations. It is also scalable: for a network with n links, it is implemented as a system of n nonlinear equations. This makes it suitable for large-scale network analysis. The proposed method is validated with a simple synthetic toy network. It is then applied to a high-dimensional problem for a large-scale network of Singapore. It is benchmarked versus a general-purpose derivative-free pattern search algorithm and versus the SPSA (Simultaneous Perturbation Stochastic Approximation) algorithm. Experiments with a computational budget of 20 simulation runs are carried out. The proposed method yields an average improvement of the objective function of 2 orders of magnitude, while the benchmark methods, GPS and SPSA, yield average improvements of between 4.4% and 1.4%, respectively. Experiments with larger computational budgets are also carried out. They indicate that even when allowing for more than double the computation time of the proposed method, the benchmark methods are still outperformed by the proposed method by 2 orders of magnitude.

This remarkable computational efficiency is achieved thanks to the structural information provided by the analytical network model. The latter is differentiable, tractable and scalable. Hence, it enables high-dimensional calibration problems to be addressed efficiently. Moreover, the structural information it provides the SO algorithm enables it to become robust to the quality of the initial points.

For general OD estimation problems, the travel behavior that underlies the traffic measurements (e.g., route choice, mode choice) is unknown. This contributes to the observability issue. Nonetheless for OD calibration problems of traffic models, where the goal is to estimate the parameter inputs of the traffic simulator, the behavioral models of the traffic simulators are specified. This specification information can be directly used

in the calibration algorithm to design efficient algorithms. For example, in this paper, Equations (7) and (10), respectively, are approximations of the route choice model and of the fundamental diagram of the specific simulator used. Since the simulator specifications tend to be intricate (e.g., rely on dynamic and traveler-specific information), the challenge lies in the formulation of approximate expressions that are also analytical, differentiable and computationally tractable to evaluate.

The general OD estimation problem is underdetermined. This makes the problem particularly difficult. A recent review and discussion on this is included in Yang *et al.* (2018), Yang and Fan (2015). The use of, more recently available, disaggregate travel data (e.g., trajectory, license plate, cell phone) has great potential to address this issue (Castillo *et al.* 2008). As part of ongoing work, we are extending the proposed methodology to account for turning proportions. The metamodel approach then relates these field measurements to route choice probabilities, which significantly improves the underdetermination problem. In the study of data traffic across computer networks, this issue has been extensively studied under the name of passive network tomography or inferential network monitoring (reviews include Xi *et al.* (2015), Lawrence *et al.* (2006), Castro *et al.* (2004)). Similar to the transportation literature, the underdetermination issue has been addressed through the use of parametric statistical models. Most OD estimation problems for data networks are formulated as stochastic OD estimation problems, with similar ideas than those recently proposed by Yang *et al.* (2018). In both fields a major challenge is the design of scalable and computationally efficient techniques. Just like the transportation literature has inspired the computer science literature (Xi *et al.* 2015), our transportation metamodel ideas could be used to enhance the scalability and efficiency of network tomography methods. Moreover, the various analytical queueing models formulated for data traffic could be readily used to formulate similar metamodels.

Recent ideas proposed by Yang and Fan (2015), indicate that the use of only partial information from the prior OD matrix, rather than using the full matrix, can contribute to improve the quality of the OD estimation and the uniqueness of the underlying solution. A natural extension to this research is the design of efficient online calibration algorithms for both high-dimensional problems and large-scale networks. One approach is to combine an efficient offline sampling strategy with an efficient online optimization strategy. The proposed approach can be used both as the basis of the offline sampling strategy and to identify a small set of points to simulate in real-time. The use of a time-dependent analytical network model may be of interest, a time-dependent metamodel formulation was proposed for signal control in Chong and Osorio (2018). The use of metamodels to perform joint calibration of demand and supply parameters is also of interest. The analytical structural information provided by these metamodels can also enable the use of higher-resolution mobility data for calibration. For instance, analytical approximations of, link or path, travel time distributions can be used in combination with trajectory data for enhanced calibration. The extension of this framework for efficient online calibration could also by achieved by embedding structural assumptions on the traffic dynamics. For instance, quasi-dynamic assumptions (Cascetta *et al.* 2013, Marzano *et al.* 2015) could contribute to account for dynamics in a tractable way. The computational budgets considered in this manuscript are considered tight even for the recent extensions of SPSA, such as c-SPSA (Tympakianaki *et al.* 2015) and and W-SPSA (Lu *et al.* 2015). Of ongoing interest is the combination of these metamodel ideas with these recent SPSA formulations, such as to further enhance their computational efficiency.

Acknowledgments

The work of C. Osorio is partially supported by the U.S. National Science Foundation under Grant No. 1351512. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. We acknowledge the support of the Intelligent Transportation Systems (ITS) Lab both at MIT and SMART (Singapore-MIT Alliance for Research and Technology). Special thanks to Dr. Bilge Atasoy and Ravi Seshadri for their help in using the Singapore network model, which was developed and calibrated by the DynaMIT team at SMART.

Appendices

A Metamodel fitting process

This appendix details how the metamodel parameters are fitted. This corresponds to Step 1 of Figure 1. This occurs at every iteration of the algorithm. At iteration k, the metamodel parameters β_k are fitted by solving the following least squares problem.

$$\min_{\beta_k} \sum_{d \in \mathcal{S}_k} \left\{ w_k(d)(\hat{E}[F_i(d, u_1; u_2)] - (\beta_{k,0} f_A(d) + \phi(d; \beta_k)) \right\}^2 + w_0^2 \left((\beta_{k,0} - 1)^2 + \beta_{k,1}^2 + \sum_{z=1}^{\text{card}(\mathcal{Z})} \beta_{k,z+1}^2 \right) \tag{13}$$

where $\hat{E}[F_i(d, u_1; u_2)]$ denotes the simulation-based estimate of the expected flow on link i for point d (i.e., OD matrix d), $\mathcal{S}_k$ represents the set of points simulated up until iteration k, $\phi(d; \beta_k)$ is defined by Eq. (4), w_0 is an exogenous (fixed) scalar weight coefficient (which is set to 0.001) and $w_k(d)$ is a scalar weight for point d defined, as in Osorio and Bierlaire (2013), by the following equation:

$$w_k(d) = \frac{1}{1 + \|d - d^k\|_2}, \tag{14}$$

where d^k denotes the current iterate.

Problem (13) fits the metamodel parameters by solving a weighted least squares problem. The first term of Problem (13) represents the weighted distance between the link flows predicted by the metamodel and those estimated by the simulator. For a given point d, its weight is proportional to its distance from the current iterate, d^k. This aims to improve the local (in the vicinity of the current iterate) fit of the metamodel. The second term of Problem (13) accounts for the distance between the parameter, β_k, and initial (or prior) values. This second term ensures that the least square matrix is of full rank. The initial values used correspond to an initial metamodel that is solely based on the analytical network model (i.e., $\beta_{k,0} = 1$ and for $j \geq 1 \;\; \beta_{k,j} = 0$).

If the analytical network model does not provide a good approximation of the simulation-based function (i.e., if f_A does not approximate well the first summation term of Equation (1)), then the scalar $\beta_{k,0}$ will asymptotically tend to zero, and the metamodel predictions will be governed by the functional term of the metamodel (i.e., the function ϕ). The

functional term also serves to asymptotically guarantee that the metamodel can provide a good local approximation to an arbitrary simulation-based function. This paper focuses on the non-asymptotic (e.g., small-sample, tight computational budget) performance of the SO algorithm, which reflects the way transportation researchers and practitioners alike use these algorithms. A discussion of the asymptotic properties and performance of the underlying SO algorithm can be found in Osorio and Bierlaire (2013).

References

Ankenman, B., Nelson, B. L., and Staum, J. (2010). Stochastic Kriging for simulation metamodeling. *Operations Research*, **58**(2), 371–382.

Antoniou, C. (2004). *On-line Calibration for Dynamic Traffic Assignment*. Ph.D. thesis, Massachusetts Institute of Technology.

Antoniou, C., Lima Azevedo, C., Lu, L., Pereira, F., and Ben-Akiva, M. (2015). W-SPSA in practice: approximation of weight matrices and calibration of traffic simulation models. *Transportation Research Part C*, **59**, 129–146.

Astarita, V., Er-Rafia, K., Florian, M., Mahut, M., and Velan, S. (2001). Comparison of three methods for dynamic network loading. *Transportation Research Record*, **1771**, 179–190.

Balakrishna, R., Ben-Akiva, M. E., and Koutsopoulos, H. N. (2007). Offline calibration of dynamic traffic assignment: simultaneous demand-and-supply estimation. *Transportation Research Record*, **2003**, 50–58.

Barton, R. R. and Meckesheimer, M. (2006). Metamodel-based simulation optimization. In S. G. Henderson and B. L. Nelson, editors, *Handbooks in operations research and management science: Simulation*, volume 13, chapter 18, pages 535–574. Elsevier, Amsterdam.

Ben-Akiva, M., Koutsopoulos, H. N., Antoniou, C., and Balakrishna, R. (2010). *Traffic Simulation with DynaMIT*, pages 363–398. Springer New York, New York, NY.

Ben-Akiva, M., Gao, S., Wei, Z., and Wen, Y. (2012). A dynamic traffic assignment model for highly congested urban networks. *Transportation Research Part C*, **24**, 62–82.

Bierlaire, M. and Crittin, F. (2004). An efficient algorithm for real-time estimation and prediction of dynamic OD tables. *Operations Research*, **52**(1), 116–127.

Cantelmo, G., Cipriani, E., Gemma, A., and Nigro, M. (2014). An adaptive bi-level gradient procedure for the estimation of dynamic traffic demand. *IEEE Transactions on Intelligent Transportation Systems*, **15**(3), 1348–1361.

Cascetta, E., Papola, A., Marzano, V., Simonelli, F., and Vitiello, I. (2013). Quasi-dynamic estimation of od flows from traffic counts: formulation, statistical validation and performance analysis on real data. *Transportation Research Part C*, **55**, 171–187.

Castillo, E., Menendez, J. M., and Jimenez, P. (2008). Trip matrix and path flow reconstruction and estimation based on plate scanning and link observations. *Transportation Research Part B*, **42**, 455–481.

Castro, R., Coates, M., Liang, G., Nowak, R., and Yu, B. (2004). Network tomography: recent developments. *Stastical Science*, **19**(3), 499–517.

Chong, L. and Osorio, C. (2018). A simulation-based optimization algorithm for dynamic large-scale urban transportation problems. *Transportation Science*, **52**(3), 637–656.

Cipriani, E., Florian, M., Mahut, M., and Nigro, M. (2011). A gradient approximation approach for adjusting temporal origin-destination matrices. *Transportation Research Part C*, **19**(2), 270–282.

Conn, A. R., Scheinberg, K., and Vicente, L. N. (2009). Global convergence of general derivative-free trust-region algorithms to first- and second-order critical points. *SIAM Journal on Optimization*, **20**(1), 387–415.

Djukic, T., van Lindt, J. W. C., and Hoogendoorn, S. P. (2012). Application of principal component analysis to predict dynamic origindestination matrices. *Transportation Research Record*, **2283**, 81–89.

Flötteröd, G., Bierlaire, M., and Nagel, K. (2011). Bayesian demand calibration for dynamic traffic simulations. *Transportation Science*, **45**(4), 541–561.

Frederix, R., Viti, F., Corthout, R., and Tampère, C. M. (2011). New gradient approximation method for dynamic origin-destination matrix estimation on congested networks. *Transportation Research Record*, **2263**, 19–25.

Google Maps (2017). Singapore expressway network.

Hazelton, M. L. (2008). Statistical inference for time varying origin-destination matrices. *Transportation Research Part B*, **42**(6), 542–552.

Huang, E. (2010). *Algorithmic and implementation aspects of on-line calibration of dynamic traffic assignment*. Master's thesis, Massachusetts Institute of Technology.

Jha, M., Gopalan, G., Garms, A., Mahanti, B. P., Toledo, T., and Ben-Akiva, M. E. (2004). Development and calibration of a large-scale microscopic traffic simulation model. *Transportation Research Record*, **1876**, 121–131.

Jones, D. R., Schonlau, M., and Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, **13**(4), 455–492.

Kattan, L. and Abdulhai, B. (2006). Noniterative approach to dynamic traffic origin-destination estimation with parallel evolutionary algorithms. *Transportation Research Record*, **1964**, 201–210.

Kim, H., Baek, S., and Lim, Y. (2001). Origin-destination matrices estimated with a genetic algorithm from link traffic counts. *Transportation Research Record*, **1771**, 156–163.

Kleijnen, J. P. C., van Beers, W., and van Nieuwenhuyse, I. (2010). Constrained optimization in expensive simulation: Novel approach. *European Journal of Operational Research*, **202**(1), 164–174.

Lawrence, E., Michailidis, G., Nair, V. N., and Xi, B. (2006). Network tomography: a review and recent cevelopments. *Frontiers in Statistics*, pages 345–366.

Lee, J. B. and Ozbay, K. (2009). New calibration methodology for microscopic traffic simulation using enhanced simultaneous perturbation stochastic approximation approach. *Transportation Research Record*, **2124**, 233–240.

Lu, L., Xu, Y., Antoniou, C., and Ben-Akiva, M. (2015). An enhanced SPSA algorithm for the calibration of dynamic traffic assignment models. *Transportation Research Part C*, **51**, 149–166.

Marzano, V., Papola, A., Cascetta, E., and Simonelli, F. (2015). Towards online quasi-dynamic o-d flow estimation/updating. In *IEEE International Conference on Intelligent Transportation Systems*, pages 1471–1476. IEEE.

Mathworks, Inc. (2016). *Global Optimization Toolbox User's Guide Matlab (R2016b)*. Natick, MA, USA.

Nie, Y. (2006). *Variational inequality approach for inferring dynamic origin-destination travel demand*. Ph.D. thesis, University of California, Davis.

Osorio, C. (2010). *Mitigating network congestion: analytical models, optimization methods and their applications*. Ph.D. thesis, Ecole Polytechnique Fédérale de Lausanne.

Osorio, C. (2019). Dynamic OD (origin-destination) calibration for large-scale network simulators. *Transportation Research Part C*, **98**, 186–206.

Osorio, C. and Atastoy, B. (2017). Efficient simulation-based toll optimization for large-scale networks. Technical report, Massachusetts Institute of Technology. Under review. Available at: http://web.mit.edu/osorioc/www/papers/osoAtaTollSO.pdf .

Osorio, C. and Bierlaire, M. (2013). A simulation-based optimization framework for urban transportation problems. *Operations Research*, **61**(6), 1333–1345.

Osorio, C. and Chong, L. (2015). A computationally efficient simulation-based optimization algorithm for large-scale urban transportation. *Transportation Science*, **49**(3), 623–636.

Osorio, C. and Nanduri, K. (2015a). Energy-efficient urban traffic management: a microscopic simulation-based approach. *Transportation Science*, **49**(3), 637–651.

Osorio, C. and Nanduri, K. (2015b). Urban transportation emissions mitigation: coupling high-resolution vehicular emissions and traffic models for traffic signal optimization. *Transportation Research Part B*, **81**, 520–538.

Osorio, C. and Selvam, K. (2017). Simulation-based optimization: achieving computational efficiency through the use of multiple simulators. *Transportation Science*, **51**(2), 395–411.

Osorio, C., Chen, X., and Santos, B. F. (2017). Simulation-based travel time reliable signal control. *Transportation Science*. Forthcoming. Available at: http://web.mit.edu/osorioc/www/papers/osoCheSanReliableSO.pdf .

Prakash, A. A., Seshadri, R., Antoniou, C., Pereira, F. C., and Ben-Akiva, M. E. (2018). Improving scalability of generic online calibration for real-time dynamic traffic assignment systems. *Transportation Research Record*. forthcoming.

Søndergaard, J. (2003). *Optimization using surrogate models - by the Space Mapping technique*. Ph.D. thesis, Technical University of Denmark.

Spall, J. (1992). Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Transactions on Automatic Control*, **37**(3), 332–341.

Spall, J. C. (2003). *Introduction to stochastic search and optimization: estimation, simulation, and control*. Wiley-Interscience series in discrete mathematics and optimization. John Wiley & Sons, New Jersey, USA.

Stathopoulos, A. and Tsekeris, T. (2004). Hybrid meta-heuristic algorithm for the simultaneous optimization of the O-D trip matrix estimation. *Computer-Aided Civil and Infrastructure Engineering*, **19**(6), 421–435.

Tavana, H. (2001). *Internally-consistent estimation of dynamic network origin-destination flows from intelligent transportation systems data using bi-level optimization*. Ph.D. thesis, University of Texas at Austin.

Tympakianaki, A., Koutsopoulos, H., and Jenelius, E. (2015). c-SPSA: Cluster-wise simultaneous perturbation stochastic approximation algorithm and its application to dynamic origin-destination matrix estimation. *Transportation Research Part C*, **55**, 231–245.

Vaze, V., Antoniou, C., Wen, Y., and Ben-Akiva, M. (2009). Calibration of dynamic traffic assignment models with point-to-point traffic surveillance. *Transportation Research Record*, **2090**, 1–9.

Verbas, I. O., Mahmassani, H. S., and Zhang, K. (2011). Time-dependent origin-destination demand estimation: challenges and methods for large-scale networks with multiple vehicle classes. *Transportation Research Record*, **2263**, 45–56.

Wang, W., Wan, H., and Chang, K. H. (2016). Randomized block coordinate descendant STRONG for large-scale stochastic optimization. In *2016 Winter Simulation Conference (WSC)*, pages 614–625.

Wild, S. M., Regis, R. G., and Shoemaker, C. A. (2008). ORBIT: Optimization by radial basis function interpolation in trust-regions. *SIAM Journal on Scientific Computing*, **30**(6), 3197–3219.

Xi, B., Yang, X., Nair, V. N., and Michailidis, G. (2015). Statistical issues in computer networks and traffic analysis. Technical Report 15-01, Purdue University.

Yang, Y. and Fan, Y. (2015). Data dependent input control for origindestination demand estimation using observability analysis. *Transportation Research Part B*, **78**, 385–403.

Yang, Y., Fan, Y., and Wets, R. J. B. (2018). Stochastic travel demand estimation: Improving network identifiability using multi-day observation sets. *Transportation Research Part B*, **107**, 192–211.

Zhang, C. and Osorio, C. (2017). Efficient offline calibration of origin-destination (demand) for large-scale stochastic traffic models. Technical report, Massachusetts Institute of Technology. Under review. Available at: http://web.mit.edu/osorioc/www/papers/zhaOsoODcalib.pdf .

Zhang, C., Osorio, C., and Flötteröd, G. (2017). Efficient calibration techniques for large-scale traffic simulators. *Transportation Research Part B*, **97**, 214–239.

Zhang, H. M., Nie, Y., and Qian, Z. (2008). Estimating time-dependent freeway origin-destination demands with different data coverage sensitivity analysis. *Transportation Research Record*, **2047**, 91–99.

Zhou, X. and Mahmassani, H. S. (2006). Dynamic origin-destination demand estimation using automatic vehicle identification data. *IEEE Transactions on Intelligent Transportation Systems*, **7**(1), 105–114.

Zhou, X., Qin, X., and Mahmassani, H. S. (2003). Dynamic origin-destination demand estimation with multiday link traffic counts for planning applications. *Transportation Research Record*, **1831**, 30–38.