This article was downloaded by: [134.148.10.12] On: 22 January 2017, At: 02:23
Publisher: Institute for Operations Research and the Management Sciences (INFORMS)
INFORMS is located in Maryland, USA

Transportation Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Simulation-Based Optimization: Achieving Computational Efficiency Through the Use of Multiple Simulators

Carolina Osorio, Krishna Kumar Selvam

To cite this article:
Carolina Osorio, Krishna Kumar Selvam (2017) Simulation-Based Optimization: Achieving Computational Efficiency Through the Use of Multiple Simulators. Transportation Science

Published online in Articles in Advance 17 Jan 2017

. http://dx.doi.org/10.1287/trsc.2016.0673

Full terms and conditions of use: http://pubsonline.informs.org/page/terms-and-conditions

This article may be used only for the purposes of research, teaching, and/or private study. Commercial use or systematic downloading (by robots or other automatic processes) is prohibited without explicit Publisher approval, unless otherwise noted. For more information, contact permissions@informs.org.

The Publisher does not warrant or guarantee the article's accuracy, completeness, merchantability, fitness for a particular purpose, or non-infringement. Descriptions of, or references to, products or publications, or inclusion of an advertisement in this article, neither constitutes nor implies a guarantee, endorsement, or support of claims made of that product, publication, or service.

Copyright © 2017, INFORMS

Please scroll down for article—it is on subsequent pages

INFORMS is the largest professional society in the world for professionals in the fields of operations research, management science, and analytics.
For more information on INFORMS, its publications, membership, or meetings visit http://www.informs.org

RIGHTSLINK

http://pubsonline.informs.org/journal/trsc/

TRANSPORTATION SCIENCE
Articles in Advance, pp. 1–17
ISSN 0041-1655 (print), ISSN 1526-5447 (online)

Simulation-Based Optimization: Achieving Computational Efficiency Through the Use of Multiple Simulators

Carolina Osorio,[a] **Krishna Kumar Selvam**[a]

[a] Department of Civil and Environmental Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139
Contact: osorioc@mit.edu (CO); kkselvam@mit.edu (KKS)

Received: August 2014
Revised: June 2015
Accepted: August 2015
Published Online in Articles in Advance: January 17, 2017

https://doi.org/10.1287/trsc.2016.0673

Copyright: © 2017 INFORMS

Abstract. Transportation agencies often resort to the use of traffic simulation models to evaluate the impacts of changes in network design or network operations. They often have multiple traffic simulation tools that cover the network area where changes are to be made. These multiple simulators may differ in their modeling assumptions (e.g., macroscopic versus microscopic), in their reliability (e.g., quality of their calibration), as well as in their modeling scale (e.g., city-scale versus regional-scale). The choice of which simulation model to rely on, let alone of how to combine their use, is intricate. A larger-scale model may, for instance, capture more accurately the local-global interactions; yet may do so at a greater computational cost. This paper proposes an optimization framework that enables multiple simulation models to be jointly and efficiently used to address continuous urban transportation optimization problems.

We propose a simulation-based optimization algorithm that embeds information from both a high-accuracy low-efficiency simulator and a low-accuracy high-efficiency simulator. At every iteration, the algorithm decides which simulator to evaluate. This decision is based on an analytical approximation of the accuracy loss as a result of running the lower-accuracy model. We formulate an analytical expression that is based on a differentiable and computationally efficient to evaluate traffic assignment model. We evaluate the performance of the algorithm with a traffic signal control problem on both a small network and a city network. We show that the proposed algorithm identifies signal plans with excellent performance, and can do so at a significantly lower computational cost than when systematically running the high-accuracy simulator.

The proposed methodology contributes to enable large-scale high-resolution traffic simulation models to be used efficiently for simulation-based optimization. More broadly, it enables the use of multiple simulation models that may differ, for instance, in their scale, their resolution, or their computational costs, to be used jointly for optimization.

Funding: This research was partly funded by the Ford Motor Company, the New England University Transportation Centers Program, and a Dwight David Eisenhower Graduate Fellowship of the Federal Highway Administration.

Keywords: simulation-based optimization • stochastic traffic simulation • traffic signal control

1. Introduction

Consider a subnetwork within a larger network (e.g., an arterial within a city, a city within a region) where local changes to the supply of the subnetwork are being considered. Transportation agencies often resort to the use of traffic simulators to determine the changes to be carried out (e.g., changes in network design or network operations), and to evaluate the impacts of these changes both locally (i.e., within the subnetwork) as well as globally (i.e., at the larger-scale).

Transportation agencies often have multiple simulators that cover the subnetwork of interest. These multiple simulators may differ in their modeling resolution (e.g., macroscopic, microscopic), in their reliability (e.g., quality of their calibration), as well as in their modeling scale (e.g., city-scale, regional-scale).

Most often, transportation experts will consider the advantages and disadvantages of each model, and will ultimately choose one model to rely on to determine and study the impact of the subnetwork changes. The choice of a model is not an easy task. A larger-scale model may, for instance, capture more accurately the local-global interactions, yet may do so at a greater computational cost. This paper proposes an optimization framework that enables multiple simulation models to be jointly and efficiently used to address continuous urban transportation optimization problems.

Recent reviews of traffic simulation models include Barceló (2010) and Ratrout and Rahman (2009). The main families of models are known as macroscopic, mesoscopic, and microscopic. Microscopic simulation models provide a high-resolution representation of both network supply (e.g., dynamic and traffic-respon-

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

RIGHTSLINK

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

sive traffic management strategies), and network demand (e.g., vehicle technologies and performance, traveler behavior such as routing, car-following, lane-changing). They are the most detailed, yet also the most computationally inefficient, models. Their computational inefficiency limits their use to address large-scale transportation optimization problems. On the other hand, macroscopic models provide a low-resolution representation of supply and demand (e.g., with aggregate flow-based network models). Macroscopic models are computationally more efficient than their microscopic counterparts and hence are often used for the analysis of large-scale problems.

To achieve a suitable tradeoff between modeling detail and computational efficiency, the transportation community has mostly resorted to the formulation of mesoscopic models that achieve a suitable detail-efficiency tradeoff by formulating a single model that combines ideas from both macroscopic and microscopic models. For a review of mesoscopic models, see Hoogendoorn and Bovy (2001). Nonetheless, the choice of which microscopic modeling assumptions to relax may be problem dependent. Hence, the difficulty of developing a general-purpose mesoscopic model that achieves a good detail-efficiency tradeoff.

A second approach has been to develop frameworks that combine the use of multiple traffic simulation models. These approaches can themselves be classified into two categories. The first category models different areas of the network with different levels of resolution (e.g., high-fidelity and low-fidelity models). Work in this field includes: Sewall, Wilkie, and Lin (2011), Burghout (2004), Horowitz (2004), and Bourrel and Lesort (2003). A detailed review is given by Burghout (2004). Much of the past work in this category focuses on the issue of model consistency at the multiresolution boundaries, and in particular, on the topic of aggregation and disaggregation of vehicle flows and densities.

The second category, which is the focus of this paper, allows for areas of the network to be simultaneously modeled with multiple simulators. Typically, a large region is modeled with a low-resolution (e.g., macroscopic) model, and a smaller subnetwork within the larger region is separately modeled with a higher resolution (e.g., microscopic/mesoscopic) model. This means that the smaller subnetwork is modeled with both the low- and high-resolution models. The boundary conditions of the smaller scale model (e.g., origin-destination (OD) matrix) are typically estimated based on large-scale low-resolution outputs.

Table 1 summarizes some of the work in this second category. The last line of the table considers the approach presented in this paper. The first column of the table states the corresponding work, columns 2 to 4 state which types of models are jointly used (microscopic, mesoscopic, or macroscopic), and column 5 indicates whether there is feedback between the multiple models. The last column indicates whether the multiple models are used to address an optimization problem.

Montero et al. (1998) propose a graphical user interface to build both macroscopic (EMME/2) and microscopic (AIMSUN) network models in a consistent way. The models are jointly used to inform road network designs in the Barcelona metropolitan area. Bunch et al. (1999) jointly use a macroscopic (EMME/2) model of the Seattle metropolitan region with a microscopic (INTEGRATION) model of a specific corridor within the region. Oh et al. (2000) consider the use of a microscopic model for the analysis of a large-scale network. They consider that microscopic route choice for large-scale networks is a computationally demanding task (due mainly to the detailed network representation), whereas mesoscopic models can calculate route choice in a more efficient manner. Hence, they consider a single network modeled at both a microscopic (Paramics) and a mesoscopic (DYNASMART) resolution. The microscopic model provides accurate aggregate link costs to the mesoscopic model, these are used as inputs for the mesoscopic route choice calculations, which are then fed back to the microscopic model. Although the authors do not present a quantitative comparison of the computation time between the inbuilt microscopic route choice and their proposed micro-meso route choice, they conclude that for large networks, their proposal of using a mesoscopic model to make route choice decisions would be advantageous. Rousseau et al. (2008) consider the integrated use of a macroscopic regional Atlanta model (CUBE), a macroscopic

Table 1. Works That have Jointly Used Multiple Traffic Models with Overlap in the Underlying Networks

	Microscopic	Mesoscopic	Macroscopic	Model feedback	Optimization
Montero et al. (1998)	AIMSUN		EMME/2		
Bunch et al. (1999)	INTEGRATION		EMME/2	✓	
Oh et al. (2000)	Paramics	DYNASMART		✓	
Rousseau et al. (2008)	Vissim		CUBE, Visum		
Osorio and Selvam (this paper)	AIMSUN[a]				✓

[a]Two AIMSUN models are used: one large-scale and one medium-scale model.

RIGHTSLINK

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

downtown Atlanta model (Visum), and a microscopic downtown core area of Atlanta (Vissim). The main differences between the modeling approaches (modeling assumptions, data challenges, computational challenges) are stated. Challenges in enabling full model feedback (both macroscopic to microscopic and microscopic to macroscopic) are mentioned. Their case study focuses on the development of the microscopic model from both field data and macroscopic model data.

This paper considers a subnetwork that is modeled with multiple models. It uses two high-resolution microscopic traffic models. That is, both the larger region and the smaller subnetwork are modeled with the same high resolution (both models are microscopic models). The larger region model is computationally costly to evaluate, whereas the subnetwork model is more efficient but less accurate because it considers fixed subnetwork boundary conditions. Hence, this paper proposes a framework that combines the use of high-accuracy low-efficiency models with low-accuracy high-efficiency models to address transportation optimization problems in an accurate and computationally efficient way. This combination leads to an algorithm that can identify points (e.g., network designs, traffic management strategies) with good performance at a reduced computational cost.

As indicated in Table 1, past work in this area has been limited to the joint use of multiple models for what-if analysis (i.e., scenario-based analysis), rather than for optimization. To the best of our knowledge, this paper is the first to embed the multiple simulation models within an optimization framework to address a variety of transportation problems.

In the broader field of optimization, the areas of surrogate-based optimization, also called multi-fidelity optimization or structural optimization, formulate multimodel optimization frameworks. For a review, see Robinson (2007). Foundational multimodel work with a focus on trust region algorithms (such as the one used in this paper) is presented by Carter (1986). The multiple models have different levels of fidelity or resolution, and also different levels of computational efficiency. The idea is to limit the use of the higher-fidelity inefficient model within the iterative optimization algorithm, and rather use frequently the lower-fidelity more efficient model. Past work in this area has considered either the use of several analytical models with varying computational costs, or the use of several deterministic simulation-based models with various computational costs. This paper embeds multiple stochastic simulation-based models within an optimization framework.

Additionally, the proposed methodology contributes to enable large-scale high-resolution stochastic simulation models to be used efficiently for simulation-based optimization (SO). Transportation agencies and researchers are increasingly resorting to the use of high-resolution simulators (e.g., stochastic, microscopic). Nonetheless, the computational cost of these models, along with their stochasticity and the typically large number of simulation evaluations required by traditional optimization methods, makes their direct use for optimization computationally inefficient. In past work, the efficiency of SO algorithms that embed microscopic models has been achieved by combining information from the large-scale high-resolution inefficient simulator with information from analytical and highly efficient traffic models (Chong and Osorio 2016, Osorio and Chong 2015, Osorio and Nanduri 2015, Osorio and Bierlaire 2013, Chen, Osorio, and Santos 2013). The SO approach proposed in this paper achieves efficiency by combining information from the high-resolution large-scale, and hence inefficient, simulator with information from a high-resolution small-scale, and hence more efficient, simulator.

The proposed algorithm allows for the use of multiple simulation models that may differ, for instance, in their scale, their resolution, or their computational costs, to be used jointly for optimization. This allows transportation experts to jointly use their available models in a systematic way. Most often, transportation agencies have access to large-scale (e.g., regional) low-resolution (macroscopic) models and small-scale (e.g., a few intersections) high-resolution (microscopic) models. The proposed algorithm can be used to combine information from both types of models. In this paper, the algorithm is illustrated by using two high-resolution (microscopic) models: one is large-scale (city-wide) and the other is medium-scale (city center). This choice is motivated by ongoing work in collaboration with the New York City Department of Transportation, where large-scale high-resolution models are being used for traffic optimization (Osorio et al. 2015). For such work, there is a need for algorithms that need not evaluate the large-scale model at every iteration.

The proposed methodology is a simulation-based optimization algorithm that embeds information from multiple models. At every iteration, the algorithm decides which model to evaluate. This decision is based on an analytical approximation of the accuracy loss. In Section 2 the methodology is formulated. It is then used to address a traffic signal control problem for both a simple network and a city network (Section 3). The main conclusions are presented in Section 4. The appendix presents implementation details.

2. Methodology

This section is structured as follows. Section 2.1 presents the general problem statement. A general formulation of the proposed multimodel framework is given in Section 2.2. The analytical traffic model used

RIGHTSLINK

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

in the framework is formulated in Section 2.3. We then consider a specific optimization problem, namely a traffic signal control problem, which is formulated in Section 2.4. A detailed description of how the general framework is applied to this specific signal control problem is given in Section 2.5. The SO algorithm used is presented in Section 2.6.

2.1. General Problem Statement

The methodology is defined for a general continuous simulation-based transportation optimization problem. It is formulated and illustrated for a specific problem in this paper: a traffic signal control problem. Hence, in the optimization algorithm a point corresponds to a given signal plan. This paper embeds, within an SO framework, multiple microscopic stochastic traffic simulation models. The computational inefficiency of microscopic models is a result of multiple factors. First, as mentioned in Section 1, they simulate individual travelers. For instance, in the case study of Section 3.2 the expected number of trips is over 12,300. They provide a high-resolution representation of traveler behavior: they can model pre-trip travel choices (e.g., mode, departure time, route) as well as en-route travel decisions (e.g., re-route, car-following, lane-changing). They also describe in detail both the interactions between vehicles, and the interactions between vehicles and the network supply. Second, the evaluation of the performance of a point (e.g., a traffic management strategy) with a stochastic microscopic simulator involves running multiple simulation replications, where each replication is computationally costly to run. Third, transportation agencies are interested in the evaluation of the performance of a transportation strategy under equilibrium assignment conditions. Evaluating an equilibrium for a given strategy, involves sequentially running the simulator multiple times and at each time running several replications, such as to derive consistent travel costs and path choices (i.e., such as to achieve an equilibrium). Hence, there is currently a need to design optimization methodologies that enable an efficient use of these inefficient models.

We assume that we have access to two simulation models that cover the subnetwork of interest and both have the same modeling resolution and assumptions (e.g., same traveler behavior models). Let R denote the larger scale simulation model (R stands for regional), and C denote the smaller scale simulation model (C stands for city). We assume that the subnetwork of interest, where network changes are to be carried out, is the entire network of C. The R model covers a larger area that includes the subnetwork of interest. This is a scenario which is often encountered in practice: R is an available large-scale model, and C is a smaller model extracted from R, and calibrated based on R outputs and, when available, on empirical data. Compared to model C, model R is assumed to lead to more accurate estimates of both local and global performance; yet is significantly more computationally expensive to evaluate.

Let us describe why model C may lead to less accurate performance estimates than those of R. For instance, the OD matrix of C may be calibrated based on model R flow outputs, which are derived for a given (e.g., the prevailing) supply/demand scenario. Hence, some (and typically most) OD pairs of C do not represent actual origins or destinations of trips, but rather locations where trips entered/exited the subnetwork C. Following a change in the supply in the subnetwork, drivers may revise their route choices. This can impact whether they actually enter the subnetwork, and if they still enter the subnetwork this can impact the location where they enter the subnetwork. This leads to a different OD matrix for C. If the C model is run with a fixed OD matrix for all supply scenarios, then the accuracy of the performance estimates will depend on the distance between the actual OD matrix (which would account for rerouting) and the fixed OD matrix. In summary, there are points (i.e., supply scenarios) where model C can yield accurate estimates and points where the estimates are inaccurate. A fixed (i.e., supply invariant) OD matrix is what we refer to as fixed boundary conditions for C.

The family of transportation problems that we consider are continuous and generally constrained problems. The objective function is estimated via a stochastic simulator, whereas the constraints are available in closed-form and are differentiable. Such a problem can be formulated as follows:

$$\min_{x} \quad f(x,\theta;\tilde{p}) = E[F(x,\theta;\tilde{p})] \tag{1}$$

$$\text{subject to} \quad h(x,\theta;\tilde{p}) = 0, \tag{2}$$

$$x \in \mathbb{R}^n. \tag{3}$$

In this formulation, x represents the decision vector (e.g., signal plans), F is the random variable that describes subnetwork performance (e.g., trip travel time). The objective function is the expected value of F. The objective function is an unknown function. We can only obtain estimates of it via stochastic simulation. The simulation model is also a function of exogenous parameters (e.g., network topology, calibrated behavioral models) which are represented by $\tilde{p}$, and of subnetwork boundary conditions θ. The constraints represented by the function h are differentiable and are available in closed form. For instance, in the signal control problem considered in this paper they represent green time constraints for every intersection (e.g., bounds, linear constraints).

In this paper, we consider an objective function that only accounts for the subnetwork performance, i.e., the aim is to improve local traffic conditions. The aim of

RIGHTSLINK

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

this paper is to derive a transportation strategy (e.g., a signal control plan, a network design alternative; hereafter called a point) that provides subnetwork improvement when evaluated with R. We assume we have a fixed computational (or simulation) budget (e.g., limited number of simulation runs, or limited simulation run-time). The objective is to identify a point with improved performance within this budget.

Three types of techniques can be used to address Problem (1)–(3):

- T1: use only R.
- T2: use only C.
- T3: use a combination of R and C.

The first technique, T1, will definitely lead to a point with improved performance, yet is inefficient since R takes longer to execute. Technique T2 is the most efficient, yet may not lead to a point with improved performance when evaluated with R. This is because the boundary conditions of C are fixed, i.e., they do not vary across points. Technique T3 is that proposed in this paper. We propose an SO technique that achieves a good tradeoff between obtaining accurate subnetwork performance estimates and doing so at a low computational cost.

At every iteration of an SO algorithm, a trial point is considered and its performance is estimated via simulation. We propose an SO algorithm that, at every iteration, can choose between evaluating the point with the model R or with the model C. If, for a given point, model C can yield an accurate performance estimate, then it should be chosen since the estimate can be obtained at a low computational cost. Otherwise, if model C leads to an inaccurate estimate, then model R should be chosen.

The main challenge is to determine, for a given point, whether model C can yield an accurate performance estimate. To address this challenge, we propose an analytical approximation of the estimation error of C. This analytical approximation is used for every trial point to decide which simulation model to run. This analytical approximation can be embedded and used within any iterative SO algorithm.

2.2. General Framework

We assume we have access to a calibrated large-scale model R. We calibrate the subnetwork model C based on the outputs of R (e.g., calibration of behavioral parameters, of OD matrix). If empirical traffic data is available, it can also be used to calibrate C. This is done once, before starting the optimization algorithm. A one-shot calibration of C before running the optimization algorithm implies that the boundary conditions of C are fixed.

At each iteration of the SO algorithm, the main decision to be made is which simulator to call (R or C). Given the varying accuracy of C, we propose an approach that approximates the accuracy loss of C. In other words, we analytically approximate the changes in the boundary conditions of C, as well as their effect on the objective function. We do this with an analytical traffic model that covers the full R network.

At iteration k of the SO algorithm, let x_k denote the trial point (e.g., signal plan) that is to be simulated. Let θ_0 denote the fixed (i.e., exogenous) subnetwork boundary conditions. These are obtained through the one-shot calibration mentioned previously. Recall that they are assumed fixed throughout the entire optimization process, i.e., they do not depend on x_k. Let θ_k denote the true (unknown) point-specific value of the subnetwork boundary conditions. Then, the absolute error made by running model C with fixed boundary conditions is

$$e(x_k) = \left| f(x_k, \theta_k; \tilde{p}) - f(x_k, \theta_0; \tilde{p}) \right|, \tag{4}$$

where f is given in Equation (1).

Let $\hat{e}(x_k)$ denote an analytical approximation of $e(x_k)$. At every iteration of the SO algorithm, the choice of the simulation model is given by

$$\begin{cases} \text{if } \hat{e}(x_k) < \eta, & \text{then run model } C; \\ \text{otherwise,} & \text{run model } R. \end{cases} \tag{5}$$

In other words, if the absolute error, as approximated analytically, is below a threshold, then we expect C to provide a sufficiently accurate performance estimate; and since it can do so at a lower computational cost, then it is the chosen approach. Otherwise, we run the more computationally expensive, yet more accurate, R model.

The analytical approximation, $\hat{e}(x_k)$, is defined by

$$\hat{e}(x_k; \alpha_k) = \alpha_{0,k}\tilde{e}(x_k) + \alpha_{1,k} + \sum_{i=1}^{o} \alpha_{i+1,k} x_k(i) + \sum_{i=1}^{o} \alpha_{i+o+1,k} x_k(i)^2, \tag{6}$$

where $x_k(i)$ denotes the ith element of the decision vector x_k, o denotes the dimension of x_k, α_k is a vector of parameters, $\alpha_{i,k}$ denotes the ith element of α_k, and the function $\tilde{e}(x_k)$ denotes an analytical approximation of $e(x_k)$ as derived from an analytical traffic model. In other words the analytical error approximation, $\hat{e}(x_k; \alpha_k)$, is a linear combination of an approximation derived by a traffic model, $\tilde{e}(x_k)$, and a correction term that is quadratic in x. The function $\tilde{e}(x_k)$ is defined next by Equation (7). A description of how the parameters α_k are estimated is given in Section 2.6.2.

We consider an analytical traffic model that covers the region of R. We use it to (analytically) approximate θ_k. Let $\hat{\theta}_k$ denote the analytical approximation of θ_k. Let $g(x_k, \hat{\theta}_k; \tilde{p})$ denote the approximation of the objective function ($f(x_k, \theta; \tilde{p})$ of Equation (1)) provided by

RIGHTSLINK

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

the analytical traffic model for point x_k, approximated subnetwork boundary conditions $\hat{\theta}_k$, and exogenous network parameters $\tilde{p}$. The analytical approximation of $e(x_k)$ is given by

$$\tilde{e}(x_k) = \left| g(x_k, \hat{\theta}_k; \tilde{p}) - g(x_k, \theta_0; \tilde{p}) \right|. \tag{7}$$

The first term on the right-hand side of Equation (7) (the term $g(x_k, \hat{\theta}_k; \tilde{p})$) approximates the objective function, f, assuming endogenous subnetwork boundary conditions. The second term approximates the objective function assuming fixed subnetwork boundary conditions. In other words, the first term approximates the estimate of f obtained by running model R, whereas the second term approximates the estimate obtained by running model C. The difference of these terms approximates the error made by model C in the estimation of the objective function.

2.3. Analytical Traffic Model

In this section, we formulate the analytical traffic model used to approximate the error $\tilde{e}(x)$ of Equation (7). The formulation builds on the traffic model of Osorio (2010, Chapter 4). First, we present the initial model of Osorio (2010), then we present its extension.

In Osorio (2010), a road network is modeled as a finite (space) capacity queueing network. Each lane is modeled as one (or a set) of queues. Each queue is considered an $M/M/1/\ell$ queue, where ℓ is the space capacity of the queue. This finite space capacity represents an upper bound on the queue-length, and is used to capture the spatial propagation of congestion (e.g., vehicular spillbacks).

For a given queue i, the following notation is used:

γ_i: external arrival rate;
λ_i: total arrival rate;
μ_i: service rate;
$\tilde{\mu}_i$: unblocking rate;
$\hat{\mu}_i$: effective service rate;
ρ_i: traffic intensity;
P_i^f: probability of being blocked after service at queue i;
ℓ_i: space capacity;
N_i: number of vehicles in queue i;
$P(N_i = \ell_i)$: probability of queue i being full (i.e., spillback probability);
p_{ij}: turning probability from queue i to queue j;
$\mathcal{D}_i$: set of downstream queues of queue i.

For a given road network represented as a queueing network, the marginal queue-length distributions of each queue are obtained by simultaneously solving for all queues the following system of equations:

$$\lambda_i = \gamma_i + \frac{\sum_j p_{ji} \lambda_j (1 - P(N_j = \ell_j))}{1 - P(N_i = \ell_i)}, \tag{8a}$$

$$\frac{1}{\hat{\mu}_i} = \frac{1}{\mu_i} + P_i^f \frac{1}{\tilde{\mu}_i}, \tag{8b}$$

$$\frac{1}{\tilde{\mu}_i} = \sum_{j \in \mathcal{D}_i} \frac{\lambda_j (1 - P(N_j = \ell_j))}{\lambda_i (1 - P(N_i = \ell_i)) \hat{\mu}_j}, \tag{8c}$$

$$P(N_i = \ell_i) = \frac{1 - \rho_i}{1 - \rho_i^{\ell_i + 1}} \rho_i^{\ell_i}, \tag{8d}$$

$$P_i^f = \sum_j p_{ij} P(N_j = \ell_j), \tag{8e}$$

$$\rho_i = \frac{\lambda_i}{\hat{\mu}_i}. \tag{8f}$$

We briefly describe the interpretation of these equations. Equation (8a) describes the conservation of flow between upstream and downstream queues. For queue i, its total arrival rate, λ_i, is related to its external arrival rate, γ_i and to the arrivals arising from upstream queues (second term in the right-hand side of the equation). The turning probabilities p_{ij} are known as routing probabilities in queueing theory. The expected time a vehicle occupies a server is given by $1/\hat{\mu}_i$ (Equation (8b)). This time is composed of two phases. First, the vehicle undergoes an initial service for an expected time of $1/\mu_i$. The queue has an underlying service rate, μ_i, that is determined by its underlying supply (e.g., flow capacity of the downstream intersection). After receiving service, a vehicle that is at queue i and is ready to proceed to queue j may do so if queue j is not full. If queue j is full (i.e., if there is a spillback at queue j), then the vehicle is forced to remain at queue i. This is known in queueing theory as blocking. This occurs with probability P_i^f and this second service is referred to as blocking time; the expected blocking time is given by $1/\tilde{\mu}_i$. The rate $\hat{\mu}_i$ is known as the *effective* service rate because it accounts for the potential occurrence of blocking (i.e., spillback), and hence it is used to approximate the expected time that a vehicle "effectively" undergoes service.

Equation (8c) describes the expected blocking time, which is a function of the effective service rate of downstream queue j, $\hat{\mu}_j$. Equation (8d) describes the probability that queue i is full; it is referred to as the blocking probability in queueing theory. In vehicular traffic this represents the spillback probability. The expression of Equation (8d) is obtained by assuming that queue i is an $M/M/1/\ell$ queue (e.g., Bocharov et al. 2004). Equation (8e) describes the probability that a vehicle at queue i gets blocked (i.e., that it cannot proceed downstream of queue i because of downstream spillbacks). Equation (8f) defines the traffic intensity, which is a ratio of expected demand to expected supply.

The main limitation of the model of Osorio (2010) for the purpose of this work is that it assumes exogenous turning probabilities, p_{ij} (i.e., traffic assignment is fixed, it does not vary with traffic conditions). In this paper, the purpose of the analytical model is to approximate how subnetwork boundary conditions

RIGHTSLINK

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

may change because of changes in supply, and in particular to approximate how the subnetwork OD matrix changes. Hence, accounting for endogenous assignment is important. We extend the aforementioned formulation (System of Equations (8)) by considering endogenous traffic assignment (i.e., the turning probabilities, p_{ij}, are endogenous). The analytical formulation of the assignment is formulated as follows.

d_s: demand for OD pair s;
c_t: expected travel cost of path t;
y_t: expected flow on path t;
l_{st}: probability that a vehicle traveling the OD pair s takes path t;
$E[T_i]$: expected travel time of queue i;
$E[N_i]$: expected number of vehicles in queue i;
l^{veh}: average vehicle length;
$v^{\text{free flow}}$: free flow speed;
κ: route choice model parameter;
$\mathcal{S}$: set of OD pairs;
$\mathcal{T}$: set of paths;
$\mathcal{Q}$: set of queues;
$\mathcal{P}_s$: set of paths of OD pair s;
$\mathcal{G}_{ij}$: set of paths that consecutively go through queues i and j;
$\mathcal{H}_i$: set of paths that go through queue i.

$$p_{ij} = \frac{\sum_{t\in\mathcal{G}_{ij}} y_t}{\sum_{t\in\mathcal{H}_i} y_t}, \quad \forall\, i \in \mathcal{Q}, \forall\, j \in \mathcal{Q}, \tag{9a}$$

$$y_t = \sum_{s\in\mathcal{S}} d_s l_{st}, \quad \forall\, t \in \mathcal{T}, \tag{9b}$$

$$l_{st} = \frac{e^{-\kappa c_t}}{\sum_{j\in\mathcal{P}_s} e^{-\kappa c_j}}, \quad \forall\, s \in \mathcal{S}, \forall\, t \in \mathcal{P}_s, \tag{9c}$$

$$c_t = \sum_{i\in\mathcal{Q}} r_{ti} E[T_i], \quad \forall\, t \in \mathcal{T}, \tag{9d}$$

$$E[T_i] = \frac{E[N_i]}{\lambda_i(1-P(N_i=\ell_i))} + \frac{l^{\text{veh}}(\ell_i - E[N_i])}{v^{\text{free flow}}} \quad \forall\, i \in \mathcal{Q}, \tag{9e}$$

$$E[N_i] = \frac{\rho_i}{1-\rho_i} - \frac{(\ell_i+1)\rho_i^{\ell_i+1}}{1-\rho_i^{\ell_i+1}}, \quad \forall\, i \in \mathcal{Q}, \tag{9f}$$

$$\gamma_i = \sum_{t\in\mathcal{T}} a^*_{ti} r_{ti} y_t, \quad \forall\, i \in \mathcal{Q}. \tag{9g}$$

The mapping of paths to queues is defined through the following exogenous parameters, which consider a given path t, and queues i and j:

$$r_{ti} = \frac{a_{ti}}{\sum_{j\in\mathcal{Q}} a_{tj} z_{ij}}, \quad \forall\, t \in \mathcal{T}, \forall\, i \in \mathcal{Q}, \tag{10}$$

$$z_{ij} = \begin{cases} 1 & \text{if } i = j; \\ 1 & \text{if } (i \neq j) \text{ and queues } i \text{ and } j \text{ are parallel queues;} \\ 0 & \text{otherwise,} \end{cases} \tag{11}$$

$$a_{ti} = \begin{cases} 1 & \text{if queue } i \text{ is part of path } t, \\ 0 & \text{otherwise,} \end{cases} \tag{12}$$

$$a^*_{ti} = \begin{cases} 1 & \text{if queue } i \text{ is part of the first link (road) of path } t, \\ 0 & \text{otherwise.} \end{cases} \tag{13}$$

Equation (9a) defines the probability of turning from queue i to queue j as the ratio of the total flow along paths that have queues i and j as consecutive queues and of the total flow that goes through queue i. Equation (9b) defines the flow along a path t, it is a function of the total demand of a given OD-pair s, denoted d_s, and the probability of choosing path t for OD-pair s, denoted l_{st}. Note that the OD-pair demand d_s of the full R network is exogenous, and obtained from the OD matrix of the R model. The path choice probability is given by the multinomial logit expression (Equation (9c)). The deterministic component of the utility function for a given route t is defined as a function of a single (exogenous) parameter κ and a route cost c_t. More details regarding the route choice model of both this analytical traffic model and of the traffic simulators are given in the appendix. The route cost c_t is defined by Equation (9d) as the expected travel time for route t. The route travel time is a function of queue travel time $E[T_i]$, which is given by Equation (9e). In Equation (9e), the first term on the right-hand side represents the (expected) delay at queue i, it is obtained by applying Little's law (Little 1961, 2011) to a finite (space) capacity queue. The second term is an approximation of the travel time to reach the physical queue of vehicles: the numerator approximates the available road-space length not occupied by a stationary vehicular queue, the denominator is the roads free-flow speed. Equation (9f) represents the expected number of vehicles in queue i, $E[N_i]$. The derivation of this closed-form expression is given in Osorio and Chong (2015, Appendix A). Equation (9g) gives the expression for the external arrival rate of queue i, γ_i. In the model of Osorio (2010), this rate is exogenous. In this paper, because we account for endogenous assignment, the external arrival rates of a given queue depend on the (endogenous) path choice probabilities, and hence are themselves endogenous.

Equations (10)–(13) describe the mapping of links to sets of queues. A link with multiple lanes is mapped as a set of parallel (i.e., side-by-side) queues. For paths that flow through multi-lane links, the exogenous parameter r_{ti} is used to distribute the path flow uniformly across the multiple lanes of the road. More specifically, for a given multi-lane road and a given path, the path flow is distributed across the lanes where the traffic movement of the path is allowed. The parameter r_{ti} can be interpreted as the proportion of flow on path t that goes through queue i.

In summary, the System of Equations (8)–(9) describes the analytical traffic model with endogenous assignment. The model takes as input: a mapping of

RIGHTSLINK

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

the network topology as a queueing network (this determines $\ell_i, \mathcal{D}_i, a, a^*, z, r, \mathcal{Q}$), an OD-matrix for the R model with fixed total demand per OD-pair (this determines $\mathcal{S}$ and d_s), a set of path alternatives that connect each OD-pair (this determines $\mathcal{P}_s, \mathcal{T}, \mathcal{G}_{ij}$, and $\mathcal{H}_i$), fixed lane flow capacities (this determines μ_i), and the other exogenous parameters are $\kappa, l^{\text{veh}}, v^{\text{free flow}}$. All other variables that appear in the System of Equations (8)–(9) are endogenous. Given the exogenous (or input) parameters, the System of Equations (8)–(9) is solved. The main outputs of the model are performance measures of traffic congestion, such as spillback probabilities $P(N_i = \ell_i)$, expected queue travel times $E[T_i]$, and expected number of vehicles in a queue $E[N_i]$. For more details on how this system of equations is solved, we refer the reader to Selvam (2014, Section 2.3.3, Appendix B).

The analytical model uses a multinomial logit path choice model (Equation (9c)). If desired, this path choice model may be replaced by any other analytical and differentiable path choice model with a closed-form expression available. For the case studies in this paper (Sections 3.1 and 3.2) the analytical model uses the multinomial logit path choice model (Equation (9c)). The simulation models may use other path choice models; they use a C-logit model in the case study of Section 3.2 and they use a multinomial logit model in the case study of Section 3.1.

This traffic model is used to approximate the expression in Equation (7), which represents the accuracy loss as a result of approximating the objective function assuming fixed subnetwork boundary conditions. In other words, the function g of Equation (7) is obtained by evaluating the analytical traffic model (Equations (8)–(9)) and using it to approximate the (simulation-based) objective function f (of Equation (1)). In Section 2.4, we detail how this is done with a specific example of a signal control problem.

2.4. Signal Control Problem

To provide a detailed presentation of how the analytical error approximation is derived from the analytical traffic model (Equation (7)), we consider a specific optimization problem. We consider a fixed-time traffic signal control problem. To formulate the problem, we introduce the following notation.

- b_i: available cycle ratio of intersection i;
- x_L: vector of minimum green splits for each phase;
- $\mathcal{I}$: set of intersection indices;
- $\mathcal{PH}(i)$: set of phase indices of intersection i;
- T_{sub}: subnetwork travel time.

$$\min_x \quad f(x;\tilde{p}) = E[T_{\text{sub}}(x;\tilde{p})] \tag{14}$$

$$\text{subject to} \quad \sum_{j\in\mathcal{PH}(i)} x(j) = b_i, \quad \forall\, i \in \mathcal{I}, \tag{15}$$

$$x \geq x_L. \tag{16}$$

The objective function of this problem is the expected travel time in the subnetwork, the function T_{sub} represents the subnetwork travel time, and x is the decision vector of green splits (i.e., ratio of green times to cycle times) for all endogenous signal phases. The parameter b_i is an exogenous parameter that represents the proportion of cycle time that can be allocated. This proportion excludes any fixed times (e.g., all-red times). Equation (15) ensures that for each intersection all available green time is allocated across all phases. In this equation $x(j)$ is the jth element of x; it represents the green split of signal phase j. Lower bounds for the green splits are ensured through (16).

The traffic model defined in Section 2.3 (Equations (8)–(9)) considers exogenous network supply and in particular exogenous flow capacities for each lane, and hence exogenous service rates μ_i for each queue. When considering a signal control problem, a change in the total green time of a signalized lane leads to a change in its flow capacity. Hence, the following equation is added to the traffic model such as to account for endogenous flow capacities for the lanes (or queues) with endogenous green splits:

$$\mu_i = \sum_{j\in\mathcal{P}_L(i)} x(j)\tilde{s}, \quad \forall\, i \in \mathcal{L}, \tag{17}$$

where $\mathcal{L}$ denotes the set of queues with endogenous green splits, $\mathcal{P}_L(i)$ denotes the set of signal phase indices of queue i, and $\tilde{s}$ is the saturation flow rate, which is an exogenous parameter. For queues that represent lanes that are either not controlled by traffic lights, or are controlled by lights with exogenous green splits, μ_i remains exogenous.

2.5. Analytical Approximation of the Accuracy Loss

We use the analytical traffic model presented in Section 2.3 to approximate the objective function error as defined in Equation (7). We first describe how the term $g(x_k, \theta_0; \tilde{p})$ of Equation (7) is derived. This term represents the analytical approximation of the objective function (Equation (14)) for signal plan x_k, assuming the boundary conditions of the subnetwork are unchanged. This means that the subnetwork OD-demand is equal to its initial value (this initial value is obtained through the one-shot calibration of C). In the analytical traffic model this means that the subnetwork external arrival rates γ_i take their initial values. We denote these initial values as $\gamma_{i,0}$. We consider the analytical model of the subnetwork (network C) with exogenous external arrival rates $\gamma_{i,0}$; we solve the System of Equations (8), (9a)–(9f), and (17).

In other words, we use the analytical model of the subnetwork, we consider an exogenous OD matrix, and allow for endogenous assignment. Then, the analytical

RIGHTSLINK

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

approximation of the objective function (Equation (14)) is given by

$$g(x_k, \theta_0; \tilde{p}) = \frac{\sum_{i\in\mathcal{A}} E[N_i]}{\sum_{i\in\mathcal{A}} \gamma_{i,0}(1 - P(N_i = \ell_i))}, \qquad (18)$$

where $\mathcal{A}$ represents the set of queues in the subnetwork, and $E[N_i]$ is given by Equation (9f). Equation (18) is derived by applying Little's law (Little 1961, 2011) to the subnetwork.

We now describe how the term $g(x_k, \hat{\theta}_k; \tilde{p})$ of Equation (7) is derived. This term represents the analytical approximation of the objective function (Equation (14)) for signal plan x_k accounting for endogenous subnetwork boundary conditions (e.g., the subnetwork OD-matrix is endogenous). We consider the analytical model of the full network (network R). We solve the System of Equations (8)–(9), (17). Then, the analytical approximation of the objective function (Equation (14)) is given by

$$g(x_k, \hat{\theta}_k; \tilde{p}) = \frac{\sum_{i\in\mathcal{A}} E[N_i]}{\sum_{i\in\mathcal{A}} \gamma_i(1 - P(N_i = \ell_i))}. \qquad (19)$$

In this equation the subnetwork demand (represented by γ_i) is endogenous, and is given by Equation (9g). Equation (19) differs from Equation (18) in the use of endogenous subnetwork demand (γ_i) as opposed to exogenous subnetwork demand ($\gamma_{i,0}$).

In summary, we use two analytical models: one of the subnetwork with fixed subnetwork boundary conditions, and one of the full network with endogenous subnetwork boundary conditions. With each model we approximate the objective function, and then subtract the respective approximations to derive an analytical approximation of the accuracy loss, which is represented by $\tilde{e}(x_k)$ in Equation (7).

2.6. Multimodel SO Algorithm

This approach can be embedded within any SO algorithm. In this paper, we choose the algorithm of Osorio and Bierlaire (2013), which is a metamodel SO algorithm. In this paper, we extend the algorithm of Osorio and Bierlaire (2013) to allow for the use of multiple simulation models. We present the proposed algorithm below. We then comment on how it differs from that of Osorio and Bierlaire (2013). For algorithmic details, we refer the reader to Osorio and Bierlaire (2013).

In this section, we denote $f(x, \theta; \tilde{p})$ as $f(x)$. The following notation of Osorio and Bierlaire (2013) is defined for a given iteration k of the algorithm.

$m_k(x; v_k)$: metamodel;
x_k: current iterate;
Δ_k: trust region radius;
v_k: vector of parameters of m_k;
n_k: total number of simulation runs carried out up until and including iteration k;
u_k: number of successive trial points rejected.

The constants $\eta_1, \bar{\gamma}, \gamma_{\text{inc}}, \bar{\tau}, \bar{d}, \bar{u}, \Delta_{\max}$ are given such that: $0 < \eta_1 < 1$, $0 < \bar{\gamma} < 1 < \gamma_{\text{inc}}$, $0 < \bar{\tau} < 1$, $0 < \bar{d} < \Delta_{\max}$, $\bar{u} \in \mathbb{N}^*$. Set the total number of simulation runs permitted, $n_{\max}$, this determines the computational budget. Set the number of simulation replications per point $\bar{r}$.

The following additional notation is introduced to extend the algorithm to a multimodel context.

$\hat{e}_k(x; \alpha_k)$: error model at iteration k (defined by Equation (6));
q_k: indicator of the simulation model choice at iteration k: $q_k = 0$ if model R is chosen, $q_k = 1$ if model C is chosen;
$\hat{f}_{q_k}(x)$: estimate of the objective function using the simulation model q_k;
η: accuracy threshold (defined by Equation (5));
n^*: number of points sampled to initially fit the error model.

2.6.1. Algorithm.

1. *Error model initialization*. If multiple simulation models are to be used then: sample n^* feasible points with each simulation model, and fit an initial error model $\hat{e}_0$ (defined by Equation (6)). Points are sampled randomly uniformly from the feasible region. Set $n_0 = 2\bar{r}n^*$.

If only one simulation model is to be used, then set $n_0 = 0$.

2. *Initialization*. Set values for the aforementioned constants. Set $k = 0, u_0 = 0$. Determine x_0 and Δ_0 ($\Delta_0 \in (0, \Delta_{\max}]$).

Compute $\hat{f}_0$ at x_0, fit an initial metamodel m_0 (i.e., compute v_0), set $n_0 = n_0 + \bar{r}$.

3. *Step calculation*. Compute a step s_k that reduces the metamodel m_k and such that $x_k + s_k$ (the trial point) is in the trust region (i.e., approximately solve the trust region subproblem).

4. *Simulation model selection*. This step selects the simulation model that is to be used to estimate the performance of the trial point. Calculate $\hat{e}_k(x_k + s_k)$. Following Equation (5): if $\hat{e}_k(x_k + s_k) < \eta$, then $q_k = 1$ (i.e., model C is selected), otherwise $q_k = 0$ (i.e., model R is selected).

5. *Acceptance of the trial point*. Compute $\hat{f}_{q_k}(x_k + s_k)$, i.e., evaluate the point $x_k + s_k$ using the selected simulation model. Compute

$$\rho_k = \frac{\hat{f}_{q_k}(x_k) - \hat{f}_{q_k}(x_k + s_k)}{m_k(x_k) - m_k(x_k + s_k)}.$$

If the current iterate, x_k, has not yet been evaluated by model q_k, then compute $\hat{f}_{q_k}(x_k)$, set $n_k = n_k + \bar{r}$, and update $\hat{e}_k$.

RIGHTSLINK

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

—If $\rho_k \geq \eta_1$ and $(\hat{f}_{q_k}(x_k) - \hat{f}_{q_k}(x_k + s_k)) > 0$, then accept the trial point: $x_{k+1} = x_k + s_k$, $u_k = 0$.

—Otherwise, reject the trial point: $x_{k+1} = x_k$, $u_k = u_k + 1$.

Include the new trial point observation in the set of sampled points ($n_k = n_k + \bar{r}$); fit the new metamodel m_{k+1}.

6. *Model improvement.* Compute $\tau_{k+1} = \|v_{k+1} - v_k\|/\|v_k\|$. If $\tau_{k+1} < \bar{\tau}$, then improve the metamodel by simulating with model R the performance of a new point x, which is sampled randomly uniformly from the feasible region. Include this new observation in the set of sampled points ($n_k = n_k + \bar{r}$). Update m_{k+1}.

7. *Trust region radius update.*

$$\Delta_{k+1} = \begin{cases} \min\{\gamma_{\text{inc}} \cdot \Delta_k, \Delta_{\max}\} & \text{if } \rho_k > \eta_1, \\ \max\{\bar{\gamma} \cdot \Delta_k, \bar{d}\} & \text{if } \rho_k \leq \eta_1 \text{ and } u_k \geq \bar{u}, \\ \Delta_k & \text{otherwise.} \end{cases}$$

If $\rho_k \leq \eta_1$ and $u_k \geq \bar{u}$, then set $u_k = 0$.

Set $n_{k+1} = n_k$, $u_{k+1} = u_k$, $k = k + 1$.

If $n_k < n_{\max}$, then go to "Step calculation" (step 3). Otherwise, stop.

2.6.2. Main Differences with the Basic SO Algorithm.

We now describe how the previous algorithm differs from that of Osorio and Bierlaire (2013).

- *New algorithmic steps.* The main difference is the inclusion of two new steps: the *error model initialization* step and the *simulation model selection* step.
- *Metamodel formulation and fit.* We use a quadratic polynomial as the metamodel, whereas Osorio and Bierlaire (2013) use a more complex metamodel formulation. The metamodel we use is defined, for a given iteration k, as

$$m_k(x, v_k) = v_{1,k} + \sum_{i=1}^{o} v_{i+1,k} x(i) + \sum_{i=1}^{o} v_{i+o+1,k} x(i)^2, \quad (20)$$

where x is the decision vector of dimension o, $x(i)$ is the ith element of x, v_k is the vector of metamodel parameters, and $v_{i,k}$ is the ith element of v_k.

The parameters of the metamodel, v_k, are fit by solving the following least squares problem:

$$\min_{v_k} \Bigg\{ \sum_{x \in \mathcal{S}_k^R} \{w_k(x)(\hat{f}^R(x) - m_k(x, v_k))\}^2 + \frac{1}{10} \sum_{x \in \mathcal{S}_k^C} \{w_k(x)(\hat{f}^C(x) - m_k(x, v_k))\}^2 + \sum_{j=1}^{2o+1} (w_0 . v_{j,k})^2 \Bigg\}, \quad (21)$$

where $\mathcal{S}_k^R$ (respectively $\mathcal{S}_k^C$) is the set of points evaluated with model R (respectively C) up until iteration k, $\hat{f}^R(x)$ (respectively $\hat{f}^C(x)$) is the estimate of the performance of point x simulated by model R (respectively C), w_0 is a constant weight, and $w_k(x)$ is a weight function defined by Equation (22). The first (respectively second) term of (21) corresponds to the weighted distance between the simulation estimates obtained from model R (respectively model C) and the corresponding metamodel predictions. Because we assume estimates obtained from R to be more accurate than those obtained from C, the second term is weighted by a factor of 1/10. The third term ensures that the least squares matrix is of full rank, even when no simulation observations have been obtained. This ensures that Problem (21) has a unique solution. This least-squares problem is solved using the Matlab routine *lsqlin*.

The same weight function as in Osorio and Bierlaire (2013) is used. It is defined by

$$w_k(x) = \frac{1}{1 + \|x_k - x\|_2}, \quad (22)$$

where x_k is the current iterate at iteration k.

- *Error model fit.* The new algorithmic steps require fitting an error model. We now detail how the parameters of the error model, $\hat{e}(x_k; \alpha_k)$ (Equation (6)) are estimated. At a given iteration k of the SO algorithm, the parameters α_k are fit by solving the following least squares problem:

$$\min_{\alpha_k} \Bigg\{ \sum_{x \in \mathcal{U}_k} (y(x) - \hat{e}(x; \alpha_k))^2 + \sum_{i=0}^{2o+1} (\tilde{w}_0 \cdot \alpha_{i,k})^2 \Bigg\}, \quad (23)$$

where $\mathcal{U}_k$ is the set of all points that have been evaluated by both model R and model C up until iteration k, $\tilde{w}_0$ is a constant weight value, $y(x)$ are simulation-based error estimates defined by

$$y(x) = \left| \hat{f}^R(x) - \hat{f}^C(x) \right|, \quad (24)$$

where $\hat{f}^R(x)$ and $\hat{f}^C(x)$ denote the performance estimates obtained by running models R and C, respectively.

The first term of (23) represents the distance between the simulated error estimates and the analytical model error approximations. The second term is used to ensure that the least squares matrix is of full rank, and hence Problem (23) has a unique solution. This least squares problem is solved with the Matlab routine *lsqlin*.

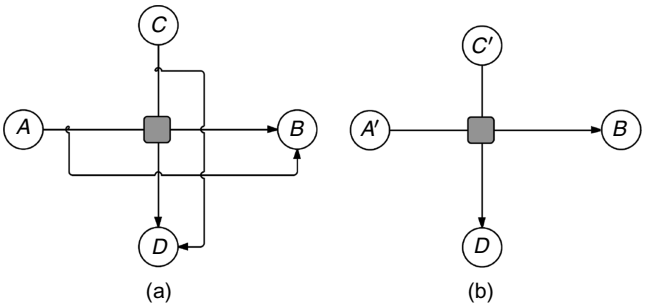
3. Case Studies

This section uses the proposed methodology to address a signal control problem for a small toy network (Section 3.1) and for the city of Lausanne, Switzerland, network (Section 3.2).

RIGHTSLINK

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

Figure 1. Network Topologies: For the *R* Network (a) and the *C* Network (b)



The performance of the algorithms are evaluated under tight computational budgets. In other words, few simulation runs are allowed (i.e., few points are evaluated, and for a given point few simulation replications are run). This is motivated by our interest in algorithms that perform well under tight computational budgets, which is how they are typically used in practice. For both case studies of this paper, the values of the computational budget (i.e., the total number of simulation runs) and of the number of simulation replications per point are chosen such as to be consistent with past SO work (Osorio and Chong 2015, Osorio and Nanduri 2015, Osorio and Bierlaire 2013, Chen, Osorio, and Santos 2013).

3.1. Toy Network

We illustrate the performance of the proposed methodology with a small toy network that is depicted in Figure 1. Figure 1(a) represents the full network *R*, which considers two OD pairs: $A \rightarrow B$ and $C \rightarrow D$. Each OD pair of network *R* has two path alternatives, one of which goes through a signalized intersection (denoted by the square in the center of each figure). The subnetwork *C* is displayed in Figure 1(b). It considers trips that start after the diverge intersections along the paths of network *R*. Hence, it considers two OD pairs, each with one path. A change in the signal plan of the intersection may affect the path choice probabilities. This change will be reflected when running simulator *R* but will not be reflected when running simulator *C*. The simulation model uses a multinomial logit path choice model, where the path cost increases with path travel time. For more details, see the appendix.

Recall that the objective function is the expected subnetwork travel time, i.e., the travel time in the links of subnetwork *C*. The signalized intersection has two endogenous signal phases: one for eastbound and westbound traffic and the second for northbound and southbound traffic. This leads to a one-dimensional signal control problem (i.e., the decision vector is of dimension one).

We compare the performance of the three techniques described in Section 2.1 (denoted T1–T3). For a given technique, we allow for a maximum of 20 simulation runs. That is, technique T1 (respectively, T2) allows for 20 runs of simulator *R* (respectively *C*), whereas technique T3 allows for a total of 20 runs that consist of a combination of runs from *R* and *C*.

For all three techniques, the SO algorithm defined in Section 2.6 is used. For techniques T1 and T2, the algorithm systematically chooses at every iteration the same simulation model, whereas for T3 either of the two models can be chosen.

Note that any steps of the SO algorithm that are carried out to select a simulation model, are not carried out for techniques T1 and T2. In particular, the initial error model fit step is not carried out by T1 or T2. Hence, for T1 and T2 all 20 simulation runs are devoted to evaluating the performance of trial points, whereas for T3 some of the simulation runs are used to improve the fit of the error model. For instance, in this case study six simulation runs are used to initially fit the error model (this corresponds to step 1 of the algorithm in Section 2.6.1). Hence, only 14 (i.e., $20-6$) simulation runs are available for T3 to use throughout the SO iterations. In other words, for a given computational budget, methods T1 and T2 evaluate the performance of a larger number of trial points than T3.

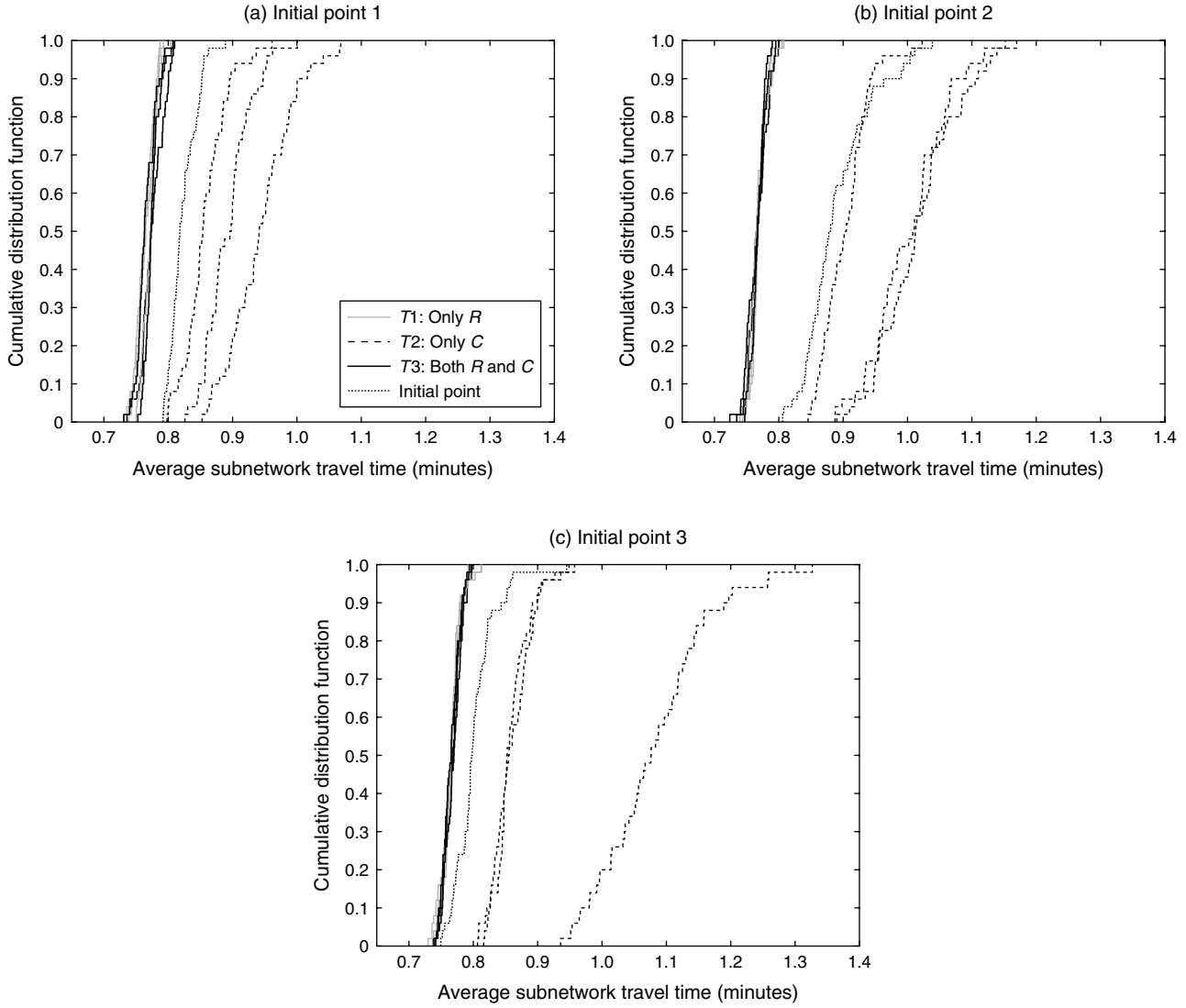
We consider three different initial signal plans. The initial signal plans are drawn randomly and uniformly from the feasible region, which is defined by Equations (15)–(16). Uniform sampling is carried out with the code of Stafford (2006).

For each initial signal plan and each technique, we run the SO algorithm three times, allowing each time for a maximum of 20 simulation runs. Once the maximum number of simulation runs is reached (i.e., the computational budget is depleted), we obtain a new signal plan as proposed by the algorithm. To evaluate the performance of a proposed signal plan, we embed it within the *R* simulator and run 50 simulation replications. For each replication, we obtain a realization of the objective function: the expected subnetwork travel time. For each signal plan, we use the 50 realizations of the average subnetwork travel time (ASTT) to construct a cumulative distribution function (cdf).

The plots of Figure 2 display several cdf curves. For each plot, the x-axis represents the ASTT. For a given x value the corresponding y value of the curve represents the proportion of replications (out of the 50 replications) where the simulated ASTT was smaller than x. Thus, the more the cdf curves are shifted to the left, the higher the proportion of simulated observations with low ASTT values, i.e., the better the performance of the corresponding signal plan.

RIGHTSLINK

Figure 2. Cumulative Distribution Functions of the Average Subnetwork Travel Time, for Each Signal Plan Proposed by Each of the Three Techniques for the City Center Network



Each plot of Figure 2 considers a different initial signal plan. Each plot contains 10 cdf curves:

- The dotted curve corresponds to the initial signal plan.
- The three dashed curves correspond to the three signal plans proposed by only running the *C* simulator. This is the least accurate yet also the least computationally-costly technique.
- The three solid grey curves correspond to the three signal plans proposed by only running the *R* simulator. This is the most accurate yet also the most computationally-costly technique.
- The three solid black curves correspond to the three signal plans proposed by our technique.

When running only the simulator *C*, all three plots of Figure 2 indicate that signal plans with poor performance are derived. For all initial points, all signal plans proposed by running only the simulator *C* perform worse than the initial signal plan.

When running only the simulator *R*, all three plots of Figure 2 indicate that signal plans with good performance are obtained. They all outperform the initial signal plan, and all lead to a subnetwork travel time average of approximately 0.75 minutes.

When running a combination of simulators *R* and *C*, the signal plans systematically yield a performance similar to the signal plans proposed by running only *R*. Additionally, for the proposed technique the *R* model was called on average 74% of the time, whereas the *C* model was called 26% of the time. The percentage of *R* calls for each of the nine runs (i.e., three SO runs for each of the three initial points) is displayed in Figure 3. These results indicate that the proposed tech-

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

RIGHTSLINK

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

Figure 3. Percentage of Calls to *R* Simulator for Each of the Nine Toy Network Experiments

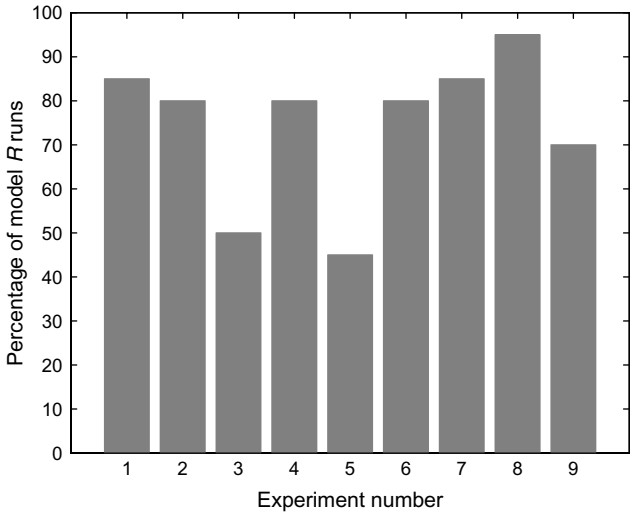
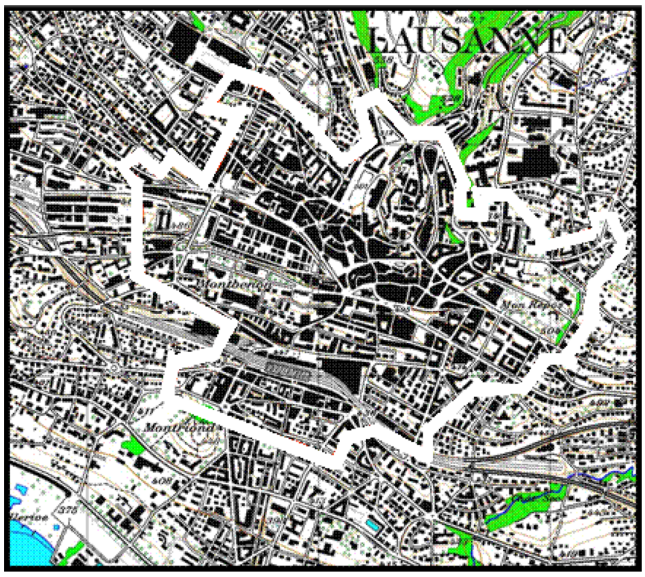


Figure 4. (Color online) Lausanne City Road Network Map



Source. Adapted from Dumont and Bert (2006).

nique identifies signal plans with good performance and does so at a lower computational cost.

3.2. Lausanne City Network

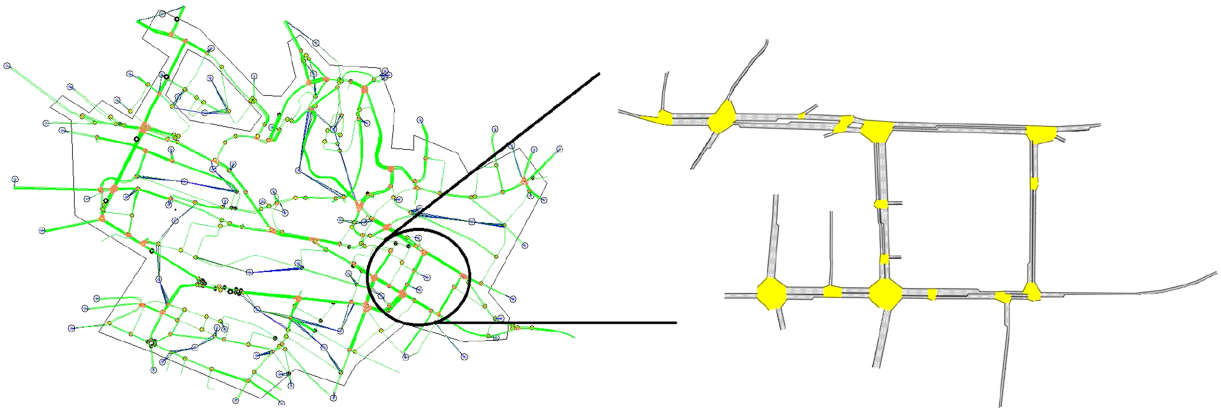
We consider a city-scale problem, focusing on the city of Lausanne. A map of the city is depicted in Figure 4. The corresponding network of the microscopic simulation model is displayed in the left plot of Figure 5. The microscopic simulator was developed and calibrated by Dumont and Bert (2006). The *R* network considers the full city, it is depicted in the left plot of Figure 5. The *C* network (also referred to as the subnetwork) considers an area within the city center, it is depicted in the right plot of Figure 5.

The *R* network consists of a total of 653 links (mapped as a network of 922 queues, certain links are represented by multiple queues in the queueing model). It has 2,075 OD pairs with non-zero demand, and an expected total demand of 12,328 trips. The subnetwork *C* has 48 links (mapped as a network of 102 queues). The *C* network has 121 OD pairs with non-zero demand, and an expected total demand of 417 trips. The simulation model uses a *C*-logit path choice model, whereas the analytical model assumes a multinomial logit model (Equation (9c)). For more route choice model details, see the appendix.

We consider a signal control problem, where the objective function is the expected travel time in the subnetwork. We control the signals of nine intersections in the subnetwork, this leads to a total of 51 endogenous signal phases, i.e., the decision vector is of dimension 51.

We proceed as for Section 3.1. We compare the performance of the three techniques T1, T2, and T3. We consider three different initial points, drawn uniformly from the feasible space. We allow for a maximum number of 200 simulation runs per experiment. That is, every time we run the SO algorithm, we allow for a total of 200 simulation runs (i.e., we have a computa-

Figure 5. (Color online) The Full (City) Network Model (Left) and the Subnetwork (City Center) Model (Right)



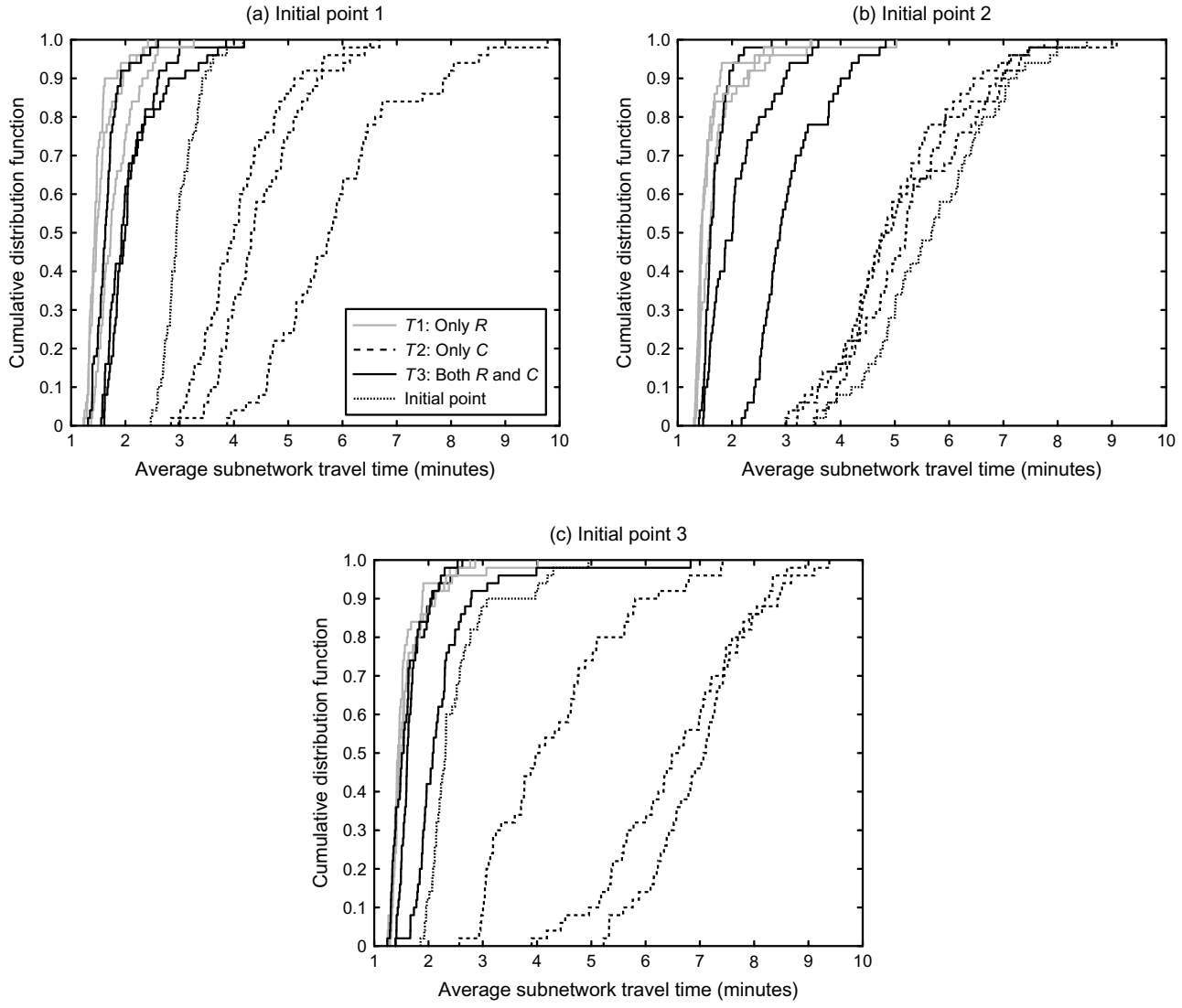
RIGHTSLINK

tional budget of 200). Once the computational budget is depleted, we terminate the algorithm, and consider the current iterate as the proposed signal plan. For each signal plan and each technique, we run the SO algorithm three times. To evaluate the performance of a proposed signal plan, we encode it within the *R* model, we run 50 simulation replications, and plot the cdf of the average subnetwork travel time.

As in Section 3.1, method T3 consumes part of the simulation budget to fit the error model. More specifically, 50 simulation runs are used to initially fit the error model (step 1 of the SO algorithm of Section 2.6.1). Hence, only 150 (i.e., 200 – 50) simulation runs are available for T3 to use throughout the SO iterations. In other words, for a given computational budget, methods T1 and T2 evaluate the performance of a larger number of trial points than T3.

Each plot of Figure 6 considers an initial signal plan. Each plot displays 10 different cdf curves. Figure 6(a) considers a first initial point. The three signal plans proposed by T2 (which only runs the *C* model) have a worse performance than the initial signal plan, and than all signal plans proposed by T1 and T3. The three signal plans proposed by T3 (both *R* and *C* models are run) outperform the initial signal plan and have a similar performance to the three plans proposed by T1 (only the *R* model is run). For the second initial signal plan (Figure 6(b)), the three signal plans of T2 perform slightly better than the initial signal plan, and worse than all signal plans proposed by T1 or T2. The three signal plans proposed by T1 have similar, and the best, performance. One of the three signal plans proposed by T3 has a similar performance to the signal plans

Figure 6. Cumulative Distribution Functions of the Average Subnetwork Travel Time, for Each Signal Plan Proposed by Each of the Three Techniques for the Lausanne City Network



Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

RIGHTSLINK

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

of T1. The other two have a significantly better performance than the initial plan and the plans proposed by T2, they are outperformed by the plans proposed by T1.

For the third initial signal plan (Figure 6(c)), similar conclusions hold. The three signal plans of T2 perform worse than all other plans, including the initial plan. Five plans have similar and the best performance: these are the three plans proposed by T1 and two out of the three plans proposed by T3. All plans proposed by T3 have a better performance than the initial signal plan.

Just as in the case of the toy network, using only model C results in signal plans with poor performance. In fact, for two of the three initial points, it leads to signal plans that perform worse than the initial point (Figures 6(a) and 6(c)). On the other hand, when only model R is used, or when both models R and C are used, signal plans with significantly better performance than the initial plan are identified.

Most signal plans derived by T3 have a similar performance to those derived by T1, and they are identified at a significantly lower computational cost. Table 2 displays simulation statistics for technique T3. Each row of the table corresponds to one of the above nine experiments (three algorithmic runs for each of the three initial points for a total of nine experiments). The last row displays the average across the above nine rows. Columns 2 and 3 display, respectively, the percentage of simulation runs where model R was chosen and the corresponding percentage of total simulation runtime. Columns 4 and 5 display, respectively, the total simulation runtime for model C and for model R. Column 6 displays the total simulation runtime, it is the sum of columns 4 and 5. Column 3 can be calculated as the ratio between columns 5 and 6. Note that the computation time of one simulation replication of model R is of the order of 90 seconds, while it is of the order of 2.8 seconds for model C, i.e., it is significantly faster to evaluate model C. Column 7 displays the percentage reduction in simulation runtime of technique T3 compared to technique T1. Recall that technique T1 only evaluates model R, hence for any experiment all 200 simulation runs are evaluated with model R, leading to a total of approximately 300 minutes in total simulation runtime. Hence, column 7 can be calculated as 100 minus the ratio of column 6 to 300.

Table 2. Simulation Statistics for Technique T3

Experiment index	Percentage of model R		Simulation runtime (min)			Simulation runtime reduction (%)
	Runs	Runtime	Model C	Model R	Total	
1	48	97	5	144	149	50
2	43	96	5	128	133	56
3	43	96	5	128	133	56
4	42	96	5	126	131	56
5	23	91	7	69	76	75
6	53	97	4	158	162	46
7	43	96	5	128	133	56
8	42	96	5	125	130	57
9	40	95	6	119	124	59
Average	42	96	5.4	125	130	57

This table indicates that, on average, technique T3 chooses model R 42% of the time. This leads to 96% of the total simulation runtime being devoted to evaluating this 42% of the runs. This emphasizes that model R is significantly costlier to evaluate than model C. By comparison to technique T1, technique T3 achieves, on average, a 57% reduction in total simulation runtime. Using both models R and C to design signal plans, leads to plans with a similar performance than the plans obtained by using only model R. This similar performance is achieved in spite of model R being called on average only 42% of the time, and with a reduction in simulation runtime of 57%. This 57% average runtime reduction corresponds to an average of 170 minutes (i.e., 300 – 130) per experiment.

For the aforementioned experiments, one single simulation replication is run when evaluating a given point (i.e., $\bar{r} = 1$). If two simulation replications are carried out ($\bar{r} = 2$), this would lead to a simulation runtime reduction per experiment of the order of six hours. Similarly, if 5, 10 or 20 simulation replications are carried out, this would lead, respectively, to a reduction of the order of 14, 28, and 57 hours.

4. Conclusions

This paper presents a simulation-based optimization methodology that allows continuous problems to be addressed with computationally inefficient simulators in a computationally efficient manner. The main idea is to avoid running the inefficient model at every iteration of the SO algorithm. This is achieved through the combined use of multiple stochastic traffic simulators. This combination allows to tradeoff the high computational costs of running accurate large-scale simulators with the lower costs of running less accurate smaller-scale simulators.

To choose a simulation model, at every iteration of the algorithm, we formulate an analytical differentiable traffic assignment model that is computationally efficient to evaluate. This analytical model is used to approximate, for a given point, the accuracy loss as a result of running the smaller-scale model. It allows us to identify points where the smaller-scale model would yield an accurate performance estimate.

We illustrate the proposed technique with a signal control problem on both a small toy network example and on a city-scale network. The proposed technique identifies signal plans with good performance and does so at a significantly lower computational cost

RIGHTSLINK

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

than when systematically running the accurate larger-scale simulator.

This approach allows for large-scale SO problems to be efficiently addressed with accurate yet inefficient simulators. These ideas can be used to enable the use of these inefficient simulators for real-time traffic control. More broadly, they constitute a first step toward the objective of combining different types of simulation models (e.g., macroscopic, mesoscopic, and microscopic) to solve transportation optimization problems in a holistic manner.

One extension of this framework is the formulation of an analytical expression for the error provided by the lower-accuracy model. This would allow the algorithm to evaluate the lower-accuracy model even when it is not expected to provide an accurate objective function estimate. This would further improve the computational efficiency of the algorithm.

The design of this multimodel SO algorithm is driven by an interest in algorithms that perform well under tight computational budgets. An open question that is of great importance to the simulation-based optimization community is how to allocate a given computational budget. In other words, how to achieve a suitable tradeoff between the number of points to evaluate and the accuracy of the evaluation per point. This is known as the *exploration versus exploitation* problem. The formulation of suitable exploration versus exploitation approaches for challenging transportation problems is a topic of ongoing work.

Acknowledgments

The authors thank Dr. Emmanuel Bert and Professor André-Gilles Dumont (LAVOC, EPFL) for providing the Lausanne simulation model.

Appendix. Implementation Details

In the case studies of this paper (Section 3), the analytical traffic model considers a multinomial logit route choice model (Equation (9c)). Let P_k denote the probability of choosing route k, and let S denote the route choice set. Then the multinomial logit choice probability is given by

$$P_k = \frac{e^{\kappa V_k}}{\sum_{\ell \in S} e^{\kappa V_\ell}}, \tag{25}$$

where V_k is known as the deterministic part of the utility function. In the analytical model V_k is defined as the expected travel time of route k.

The simulation model of the toy network case study (Section 3.1) also considers a multinomial logit, with the route utility V_k defined as a decreasing function of route travel time. For more details regarding the definition and calculation of V_k for the simulation model, we refer the reader to TSS-Transport Simulation Systems (2010, Section 12.4.1.2). The simulation model of the Lausanne case study (Section 3.2) considers a *C*-logit route choice model (Cascetta et al. 1996), which accounts for the effect of path overlapping. It is defined as

$$P_k = \frac{e^{\kappa(V_k - CF_k)}}{\sum\limits_{\ell \in S} e^{\kappa(V_\ell - CF_\ell)}}, \tag{26}$$

where CF_k is known as the commonality factor, and is proportional to the degree of overlap between path k and other path alternatives. It is given by

$$CF_k = \beta \cdot \ln\left(\sum_{\ell \in S} \frac{L_{\ell k}}{\sqrt{L_\ell L_k}}\right)^{\tilde{\beta}}, \tag{27}$$

where $L_{\ell k}$ is the length of the links common to paths ℓ and k while L_ℓ and L_k are the lengths of paths ℓ and k, respectively. The model parameters are β and $\tilde{\beta}$. For more details regarding the calculation of the *C*-logit path choice probability by the simulation model, see TSS-Transport Simulation Systems (2010, Section 12.4.2.2.6).

In the case studies of Section 3, the analytical traffic model considers for each OD pair a maximum of three path alternatives. These alternatives are chosen as the three distance-based shortest paths. Although the simulation models account for a much larger path choice set, the choice set of the analytical model is limited to three fixed paths. This contributes to ensure that the analytical model remains both scalable and computationally tractable.

The values of all algorithmic parameters are given in Tables A.1 and A.2.

Table A.1. Numerical Values of Algorithmic Parameters, Part 1

Network	x_L (sec)	l^{veh} (m)	s (veh/hr)	$v^{\text{free flow}}$ (km/hr)	o	η	r	Δ_0	$\Delta_{\max}$
Toy	4	4	2,300	60	1	2	1	1,000	1e10
Lausanne	4	4	1,800	60	42	2	1	1,000	1e10

Table A.2. Numerical Values of Algorithmic Parameters, Part 2

Network	η_1	$\bar{\gamma}$	γ_{inc}	$\bar{\tau}$	$\bar{d}$	$\bar{u}$	w_0	$\tilde{w}_0$	$n_{\max}$	n^*	κ	β	$\tilde{\beta}$
Toy	1e–3	0.9	1.2	0.1	1e–2	10	0.1	0.001	20	3	60	NA	NA
Lausanne	1e–3	0.9	1.2	0.1	1e–2	10	0.1	0.001	200	25	7	0.15	1

RIGHTSLINK

Downloaded from informs.org by [134.148.10.12] on 22 January 2017, at 02:23 . For personal use only, all rights reserved.

References

Barceló J (2010) *Fundamentals of Traffic Simulation*, Internat. Series Oper. Res. Management Sci., Vol. 145 (Springer, New York).

Bocharov PP, D'Apice C, Pechinkin AV, Salerno S (2004) *Queueing Theory*, Modern Probability Statist. (Brill Academic Publishers, Zeist, Netherlands), 96–98.

Bourrel E, Lesort J-B (2003) Mixing microscopic and macroscopic representations of traffic flow: Hybrid model based on Lighthill–Whitham–Richards theory. *Transportation Res. Record: J. Transportation Res. Board* 1852(1):193–200.

Bunch JA, Hatcher SG, Larkin J, Nelson GG, Proper AT, Roberts DL, Shah V, Wunderlich KE (1999) Incorporating ITS into corridor planning: Seattle case study. Technical report MTR 1999-40, Center for Telecommunications and Advanced Technology, McLean, VA.

Burghout W (2004) Hybrid microscopic-mesoscopic traffic simulation. Unpublished doctoral thesis, KTH Royal Institute of Technology, Stockholm.

Carter RG (1986) Multi-model algorithms for optimization. Unpublished doctoral thesis, Rice University, Houston.

Cascetta E, Nuzzolo A, Russo F, Vitetta A (1996) A modified logit route choice model overcoming path overlapping problems: Specification and some calibration results for interurban networks. *Proc. 13th Internat. Sympos. Transportation Traffic Theory* (Elsevier, Oxford, UK), 697–711.

Chen X, Osorio C, Santos B (2013) Travel time reliability in signal control problem: Simulation-based optimization approach. *Proc. Transportation Res. Board (TRB) Conf., Washington, DC*.

Chong L, Osorio C (2016) A simulation-based optimization algorithm for dynamic large-scale urban transportation problems. *Transportation Sci.* Forthcoming.

Dumont AG, Bert E (2006) Simulation de l'agglomération Lausannoise SIMLO. Technical report, Laboratoire des voies de circulation, ENAC, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland. http://web.mit.edu/osorioc/www/papers/dumont06BertRapport.pdf.

Hoogendoorn S, Bovy P (2001) State-of-the-art of vehicular traffic flow modelling. *Proc. Institution Mechanical Engineers. Part I: J. Systems Control Engrg.* 215(4):283–303.

Horowitz R (2004) Development of integrated meso/microscale traffic simulation software for testing fault detection and handling in AHS. Technical Report UCB-ITS-PRR-2004-19, California PATH Research Report, University of California, Berkeley.

Little JDC (1961) A proof for the queuing formula: $L = \lambda W$. *Oper. Res.* 9(3):383–387.

Little JDC (2011) Little's law as viewed on its 50th anniversary. *Oper. Res.* 59(3):536–549.

Montero L, Codina E, Barceló J, Barceló P (1998) Combining macroscopic and microscopic approaches for transportation planning and design of road networks. *Proc. 19th Conf. Australian Road Res. Board, Sydney*.

Oh JS, Cortés CE, Jayakrishnan R, Lee D (2000) Microscopic simulation with large-network path dynamics for advanced traffic management and information systems. *Proc. 6th ASCE Internat. Conf. Appl. Adv. Tech. Transportation Engrg.*

Osorio C (2010) Mitigating network congestion: Analytical models, optimization methods and their applications. Unpublished doctoral thesis, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland.

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

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

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

Osorio C, Chen X, Marsico M, Talas M, Gao J, Zhang S (2015) Reducing gridlock probabilities via simulation-based signal control. *Transportation Res. Procedia, Internat Sympos. Transport Simulation (ISTS)*, Vol. 6, 101–110.

Ratrout NT, Rahman SM (2009) A comparative analysis of currently used microscopic and macroscopic traffic simulation software. *Arabian J. Sci. Engrg.* 34(1B):121–133.

Robinson TD (2007) Surrogate-based optimization using multifidelity models with variable parameterization. Unpublished doctoral thesis, Massachusetts Institute of Technology, Cambridge.

Rousseau G, Scherr W, Yuan F, Xiong C (2008) An implementation framework for integrating regional planning model with microscopic traffic simulation. *Logist.: Emerging Frontiers Transportation Development China: Proc. 8th Internat. Conf. Chinese Logist. Transportation Professionals*.

Selvam KK (2014) Multi-model simulation-based optimization applied to urban transportation. Unpublished Master's thesis, Massachusetts Institute of Technology, Cambridge.

Sewall J, Wilkie D, Lin MC (2011) Interactive hybrid simulation of large-scale traffic. *ACM Trans. Graphics (TOG)* 30(6):Article 135.

Stafford R (2006) The theory behind the "randfixedsum" function. http://www.mathworks.com/matlabcentral/fileexchange/9700.

TSS-Transport Simulation Systems (2010) Microsimulator and mesosimulator, Aimsun 6.1 user's manual.

RIGHTSLINK