A simulation-based optimization framework for urban traffic congestion management

Carolina Osorio and Michel Bierlaire

Transport and Mobility Laboratory
Ecole Polytechnique Fédérale de Lausanne

Abstract

Microscopic simulators embed numerous traffic models that make them detailed and realistic tools appropriate to perform scenario-based or sensitivity analysis. This realism leads to nonlinear objective functions with no available closed form and containing potentially several local minima. As nonlinear, stochastic and evaluation-expensive models, their integration within an optimization framework remains a difficult and challenging task. We believe that in order to perform both fast and reliable simulation optimization for congested networks, information from the simulation tool should be combined with information from a network model that analytically captures the structure of the underlying problem.

This paper presents a surrogate that combines the information from a calibrated microscopic traffic simulation model of the Lausanne city center (Dumont and Bert, 2006), with an analytical queueing network model (Osorio and Bierlaire, 2009a) that resorts to finite capacity queueing theory to capture the key traffic dynamics and the underlying network structure, e.g. how upstream and downstream queues interact, how this interaction is linked to network congestion. This network model, which consists of a system of nonlinear equations, has been successfully used in past work to a solve traffic signal control problem (Osorio and Bierlaire, 2009b).

We integrate this surrogate within a derivative-free (DF) trust region optimization framework (Conn et al., 2009a). Resorting to a DF algorithm is particularly appropriate for noisy problems where the derivatives are difficult to obtain and often unreliable. This is also the case when the evaluation of the objective function is computationally expensive, or when the simulation source code is unavailable. In the field of transportation, the simulators typically fall into all three of these categories.

The framework is illustrated by solving a fixed-time signal control problem for a subnetwork of the Lausanne city center. The performance of the derived plans is compared to that of other methods, including an existing plan for the city of Lausanne.

Keywords: *Traffic control, Simulation-Optimization, Metamodel, Queueing*

1 Introduction

Microscopic urban simulators capture in detail the behavior of drivers as well as their interaction with the network infrastructure. They can provide accurate network performance estimates in the context of scenario-based analysis or sensitivity analysis. They are therefore often used to evaluate traffic management schemes. Nevertheless, using them to derive appropriate management schemes (i.e. to perform optimization) is a complex task.

An optimal traffic management scheme can be formulated as:

$$\min_{x,z\in\Omega} E[f(x,z;p)], \tag{1}$$

where the objective is to minimize the expected value of a suitable network performance measure, f. This performance measure is a function of a decision or control vector x, endogenous variables z and exogenous parameters p. The feasible space Ω consists of a set of constraints that link x to z, p and f.

For instance, a traffic signal control problem can take f as the average vehicle travel time and x as the green splits for the signalized lanes. Elements such as the total demand or the network topology will be captured by p, while z will account, for instance, for the capacities of the signalized lanes.

The various traffic models embedded within the simulator make it a detailed and realistic model, but lead to noisy nonlinear objective functions containing potentially several local minima. These objective functions have no available closed form; we can only derive estimates for $E[f(x,z;p)]$. Additionally, evaluating these estimates is computationally expensive because they involve running numerous replications. As a nonlinear, stochastic and computationally-expensive problem, it is complex to address.

Metamodel methods

As is detailed in Section 2, one approach to perform simulation-based optimization (SO) is to build an analytical model of the objective function based on a sample of simulated observations. This analytical model is referred to as a metamodel or a surrogate model. This family of approaches is known as metamodel methods. Once the metamodel is constructed (e.g. fitted) it is used to perform optimization.

This approach is depicted in Figure 1. The metamodel is denoted as m, whereas the simulation response is denoted as $\hat{f}$. This figure illustrates the two main steps of metamodel methods. Firstly, the metamodel is constructed based on simulated observations. Secondly, once the metamodel m has been fitted, the optimization process derives a trial point based on the predictions and properties of m.

These steps can then be iterated as follows. For a given sample the metamodel is fitted, the optimization problem is solved, deriving a trial point. Then the performance of this trial point can be evaluated by the simulator, which leads to a new observation. As new observations become available the accuracy of the metamodel can be improved, leading ultimately to more reliable trial points.

Metamodels are typically deterministic functions that are cheaper to evaluate. By replacing the stochastic response of the simulation by a deterministic function, deterministic optimization techniques can be used. Furthermore, by using metamodels that are cheap to evaluate, the number of objective function evaluations is no longer a limitation. The main limitation remains the number of simulation runs needed such that an accurate metamodel can be built and well-performing trial points can be derived.

The most common metamodels (also called surrogates) used to perform simulation-based optimization are general-purpose models (e.g. polynomials, splines) that can be used to approximate

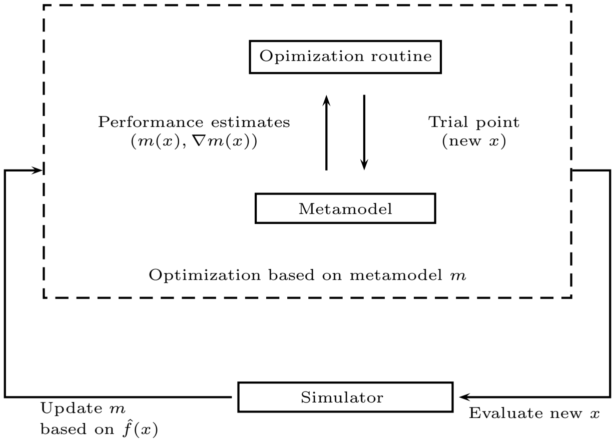


Figure 1: Metamodel simulation-based optimization methods. Adapted from Alexandrov et al. (1999).

any objective function, but capture little information about the structure of the underlying problem. Furthermore, they require a large initial sample to be fitted, and are thus inappropriate for applications with a tight computational budget.

We believe that in order to perform SO for congested urban networks given a limited computational budget, these generic metamodels should be combined with surrogate network models that analytically capture the structure of the underlying problem, and potentially improve the short-term behavior of SO algorithms.

Derivative-free optimization

Both the noise inherent in simulation outputs along with their high computational cost, makes the accurate estimation of derivatives an expensive and difficult task. When derivative information is either unavailable, available at a high cost or unreliable, then derivative-free (DF) optimization methods are an appropriate and common approach to tackle such problems.

Given the lack of derivative information, DF methods typically perform less well than their derivative-based counterparts. In particular, the scale of the problems that can be efficiently tackled is substantially reduced. Currently, the most recent DF methods are limited to around 20 variables for unconstrained problems and their convergence is typically rather slow (Conn et al., 2009b), not to mention their limitations in the presence of constraints. By using a surrogate that integrates structural information, we expect to be able to address both larger and constrained problems more efficiently.

Furthermore, DF methods are often used when function evaluations are computationally expensive. They therefore attempt to identify trial points with improved performance, given a fixed, and typically tight, computational budget. We expect the added structural information of the metamodel to allow the identification of good trial points even for tight computational budgets. In this paper, we will evaluate the performance of the proposed metamodel considering scenarios with tight computational budgets and assuming that there are initially no observations available.

This paper is structured as follows. Firstly, we present a literature review of the surrogate models used to perform SO and of the existing optimization algorithms that allow for arbitrary surrogates (Section 2). In Section 3, we present both the surrogate model and the optimization algorithm that will be used. We then show how this methodology applies to a fixed-time traffic signal control problem (Section 4), and present empirical results evaluating its performance (Section 5).

2 Literature review

There are two types of approaches to address SO problems. Firstly, there is the family of direct-search methods, which rely only on objective function evaluations and do not resort to any direct or indirect (i.e. explicit or implicit) derivative approximation or model building. In these methods, the search is based on a pre-determined geometric pattern or grid.

Secondly, there are the methods that do use gradient information. Barton and Meckesheimer (2006) review such methods and classify them into direct gradient and metamodel methods. Direct gradient methods estimate the gradient of the simulation response, whereas metamodel methods use an indirect-gradient approach by computing the gradient of the metamodel, which is a deterministic function.

In this paper, we focus on the second family of methods, and in particular on metamodel methods. Although there have been significant advances and novel approaches for gradient estimation (Fu, 2006; Fu et al., 2005), methods that rely on direct derivative information often require more function evaluations, and are therefore inappropriate for applications with a limited computational budget. Additionally, by resorting to a metamodeling approach the stochastic response of the simulation is replaced by a deterministic metamodel response function, such that deterministic optimization techniques can be used.

Recent reviews of metamodels are given by Conn et al. (2009b), by Barton and Meckesheimer (2006) and by Søndergaard (2003). Metamodels are classified in the literature as either physical or functional metamodels (Søndergaard, 2003; Serafini, 1998). Physical metamodels consist of application-specific metamodels, whose functional form and parameters have a physical or structural interpretation.

Functional metamodels are generic (i.e. general-purpose) functions, that are chosen based on their analytical tractability, but do not take into account any information with regards to the specific objective function, let alone the structure of the underlying problem. They are often a linear combination of basis functions from a parametric family.

The most common approach is the use of low-order polynomials (e.g. linear or quadratic). Quadratic polynomials are used as surrogates in most trust region methods (Conn et al., 2000). Spline models have also been used, although their use within an SO framework has focused on univariate or bivariate functions, and as Barton and Meckesheimer (2006) mention: “unfortunately, the most popular and effective multivariate spline methods are based on interpolating splines, which have little applicability for SO”. Radial basis functions (Oeuvray and Bierlaire, 2009; Wild et al., 2008) and Kriging surrogates (Booker et al., 1999) have also been proposed.

The existing metamodels consist of either physical or functional components. The metamodel proposed here goes beyond existing approaches by combining both a physical and a functional component. It combines an analytical network traffic model with a quadratic polynomial. The physical component is captured by the traffic model, whose parameters have a structural interpretation. For a given problem, the traffic model will yield a different functional form for the objective function. The functional component is captured by the general purpose quadratic polynomial.

In order to integrate the proposed metamodel within an existing optimization method, we review the algorithms that allow for an arbitrary metamodel. These methods are called multi-model or hybrid methods. They share a common motivation, which is to combine the use of models with varying evaluation costs (low versus high-fidelity models, or coarse versus fine models).

A trust region optimization framework for unconstrained problems allowing for multiple models was proposed by Carter (1986) (see references herein for previous multi-model frameworks). His work analyses the theoretical properties and derives a global convergence theory for several types of multi-model algorithms. It allows for nonquadratic models as long as at least one model is a standard quadratic with uniformly bounded curvature.

The Approximation and Model Management Optimization/Framework (AMMO or AMMF) is a trust region framework for generating and managing a sequence of metamodels. There are several versions of the algorithm: for unconstrained problems (Alexandrov et al., 1998), bound constrained (Alexandrov et al., 2000), inequality constrained (Alexandrov et al., 1999) and generally constrained problems (Alexandrov et al., 2001). Although no restrictions are imposed on the type of surrogates allowed, it is a first-order method that requires that the model and the objective function, as well as their first-order derivatives, coincide at each major (or accepted) iterate. Thus the metamodel must always behave as a first-order Taylor series approximation. This is a strong restriction if the function is noisy and expensive to evaluate.

The Surrogate-Management Framework (SMF) proposed by Booker et al. (1999) is a derivative-free method for bound constrained problems. It is based on a direct search technique called pattern search. Since direct search techniques typically require many function evaluations, they use a surrogate model of the objective function to improve the performance of the algorithm. The surrogate model used is an interpolated Kriging model.

The Space Mapping (SM) technique and its many versions (Bandler et al., 2006; Bandler et al., 2004) is a simulation-based optimization technique that uses two metamodels: a fine and a coarse model. Both models are often simulation-based. The coarse model is constructed based on a transformation of the endogenous variables ("space mapping") that minimizes the error for a sampled set of high-fidelity response values. Nevertheless, SM relies on the assumption that via a transformation of the endogenous variables the coarse model will exhibit the physical/mathematical properties of the fine model (Alexandrov and Lewis, 2001) and as Bandler et al. (2004) mention "the required interaction between coarse model, fine model, and optimization tools makes SM difficult to automate within existing simulators". Alexandrov and Lewis (2001) give a comparison of the AMMO, the SMF and the SM methods.

Conn et al. (2009a) recently proposed a trust region derivative-free framework for unconstrained problems. This framework allows for arbitrary metamodels and makes no assumption on how these metamodels are fitted (interpolation or regression). To ensure global convergence, a model improvement algorithm guarantees that the models achieve a uniform local behavior (i.e. satisfy Taylor-type bounds) within a finite number of steps.

Derivative-free (DF) methods do not require nor do they explicitly approximate derivatives. Resorting to a DF algorithm, rather than to first or second order algorithms, is appropriate when the derivatives are difficult to obtain, unreliable or computationally expensive to evaluate. This is the case for noisy problems, for problems where the evaluation of the objective function is computationally expensive, or for problems where the simulation source code is unavailable (Moré and Wild, 2009). In the field of transportation, most simulators fall into all three of these categories. Thus we will opt for a DF approach.

Among the two main strategies used to ensure global convergence, line search and trust region methods, the latter are more appropriate for our context since they "extend more naturally than line search methods to models that are not quadratics with positive Hessians" (Carter, 1986). The most common approach for fitting metamodels within a trust region (TR) framework is interpolation. Nevertheless, for noisy functions we believe that regression is more appropriate since it is less sensitive to the inaccuracy of the observations.

The framework proposed by Conn et al. (2009a), as a derivative-free TR method that allows for arbitrary models and does not impose interpolation, is therefore particularly appealing. We will therefore integrate the proposed metamodel within this framework.

3 Method

3.1 Metamodel

The metamodel combines information from two models: a simulation model and an analytical network model. We first present these two models, we then describe how they are combined.

Simulation model. We use a calibrated microscopic traffic simulation model of the Lausanne city center. This model (Dumont and Bert, 2006) is implemented with the AIMSUN simulator (TSS, 2008). Details regarding the network are given in Osorio and Bierlaire (2008). For a given decision vector x, the simulator provides a realization $\hat{f}(x, z; p)$ of the performance measure $f(x, z; p)$ (presented in Equation (1)).

Analytical queueing model. The model used in this framework is the analytical urban traffic model formulated in Osorio and Bierlaire (2008). Alternatively, for large-scale networks the model of Osorio (2010) can also be used.

In Osorio and Bierlaire (2008), this queueing model was used as a surrogate to perform optimization, and in particular to solve a fixed-time traffic control problem. In this paper, it will be used to formulate a metamodel to perform simulation-based optimization.

We briefly, recall the main components of this model. By resorting to finite capacity queueing theory, it captures the key traffic interactions and the underlying network structure, e.g. how upstream and downstream queues interact, and how this interaction is linked to network congestion. The model consists of a system of nonlinear equations. It is formulated based on a set of exogenous parameters q that capture the network topology, the total demand, as well as the turning probabilities. A set of endogenous variables y describe the traffic interactions, e.g. spillback probabilities, average rates at which a spillback diffuses. For a given decision vector x, the network model yields the objective function $T(x, y; q)$, which is a deterministic approximation of $E[f(x, z; p)]$.

We recall here the notation that we have introduced so far:

x	decision vector;
T	approximation of the objective function derived by the queueing model;
$\hat{f}$	performance measure observation derived by the simulation model;
y	endogenous queueing model variables;
z	endogenous simulation variables;
q	exogenous queueing model parameters;
p	exogenous simulation parameters.

We now describe how $\hat{f}$ and T are combined to derive the metamodel m. The main idea of trust region methods is to build, at each iteration, a model of the objective function which one "trusts" in a neighborhood of the current iterate, the *trust region*. The most common approach is to use a quadratic polynomial. The proposed metamodel combines a quadratic polynomial with a deterministic approximation of the objective function, provided by the analytical network model. The functional form of m is:

$$m(x, y; \alpha, \beta, q) = \alpha T(x, y; q) + \phi(x; \beta), \tag{2}$$

where ϕ is a quadratic polynomial in x, α and β are parameters of the metamodel.

The polynomial ϕ is quadratic in x with a diagonal second derivative matrix. This choice is based on existing numerical experiments for derivative-free TR methods, which show that these types of quadratic polynomials are often more efficient than full quadratic polynomials (Powell, 2003).

$$\phi(x; \beta) = \beta_1 + \sum_{j=1}^{d} \beta_{j+1} x_j + \sum_{j=1}^{d} \beta_{j+d+1} x_j^2, \tag{3}$$

where d is the dimension of x, x_j and β_j are the j^{th} components of x and β, respectively.

At each iteration of a trust region algorithm the objective function is evaluated at a set of points. The model is then constructed based on objective function observations. Traditionally, trust region methods fit the polynomial via interpolation. In this framework, we fit the metamodel via regression. At each iteration, the simulator and the queueing model are evaluated at one (in some cases two) point(s). The metamodel is fitted using the observations obtained at the current iteration, as well as all observations collected at previous iterations.

The parameters β and α of the metamodel are fitted by solving a least squares problem. At a given iteration, the model approximates the objective function in a neighborhood of the current iterate. In order to give more importance to observations that correspond to points that are near the current iterate, we associate weights to each observation. The least squares problem is formulated as follows.

$$\min_{\alpha,\beta} \sum_{i=1}^{n_k} \Big\{ w_{ki} \Big(\hat{f}(x^i, z^i; p) - m(x^i, y^i; \alpha, \beta, q) \Big) \Big\}^2 \; + \; (w_0.(\alpha - 1))^2 + \sum_{i=1}^{2d+1} (w_0.\beta_i)^2, \tag{4}$$

where x^i represents the i^{th} point in the sample, with corresponding endogenous simulation variables z^i, endogenous queueing model variables y^i and observation $\hat{f}(x^i, z^i; p)$. The sample size at iteration

k is n_k. The weight associated at iteration k to the the i^{th} observation is denoted w_{ki}. The parameter w_0 represents a fixed weight, its role will be discussed further on.

The first squared term of Equation (4) represents the weighted distance between the simulated observations and the metamodel predictions. The next two squared terms measure the distance between the parameters and their initial values. These terms ensure that the least squares matrix is of full rank. The initial values used here (one for α and zero for β) lead to an initial metamodel that is based only on the queueing model. This is of interest when starting off the algorithm with few or even no observations.

The weights w_{ki} capture the importance of each point with regards to the current iterate. The work of Atkeson et al. (1997) gives a survey of weight functions and analyzes their theoretical properties. We use what is known as the *inverse distance* weight function along with the Euclidean distance. This leads to the following weight parameters:

$$w_{ki} = \frac{1}{1 + \|x_k - x^i\|_2}, \tag{5}$$

where x_k is the current iterate, and x^i is the i^{th} sample point.

The weight of a given point is therefore inversely proportional to its distance from the current iterate. This allows us to approximately have a Taylor-type behavior, where observations corresponding to local points have more weight. The least squares problem is solved using the Matlab routine *lsqlin* (The Mathworks, 2008).

3.2 Algorithmic framework

For an introduction to trust region (TR) methods, we refer the reader to Conn et al. (2000). They summarize the main steps of a TR method in the *Basic trust region algorithm.* The method proposed by Conn et al. (2009a) builds upon the *Basic TR algorithm* by adding two additional steps: a model improvement step and a criticality step. For a detailed description, see Conn et al. (2009a).

A given iteration k of the algorithm considers a metamodel m_k, an iterate x_k and a TR radius Δ_k. Hereafter, the subscript k refers to the iteration. Each iteration consists of 5 steps:

- **Criticality step.** This step may modify m_k and Δ_k if the measure of stationarity is close to zero.
- **Step calculation.** Approximately solve the TR subproblem to yield a trial point.
- **Acceptance of the trial point.** The actual reduction of the objective function is compared to the reduction predicted by the model, this determines whether the trial point is accepted or rejected.
- **Model improvement.** Either certify that m_k is *fully linear* (i.e. satisfies Taylor-type bounds) in the TR or attempt to improve the accuracy of the metamodel.
- **TR radius update.**

3.3 Algorithm

The algorithm used in this framework follows. We then provide details regarding the implementation of each step of the algorithm.

0. **Initialization.** Set
   - an initial point x_0,
   - an upper bound for the trust region radius $\Delta_{max} > 0$,
   - an initial trust region radius $\Delta_0 \in (0, \Delta_{max}]$,
   - the parameters $\eta_1, \gamma, \gamma_{inc}, \epsilon_c, \bar{\tau}, \bar{d}, \bar{u}$ such that
     - $0 < \eta_1 < 1$,
     - $0 < \gamma < 1 < \gamma_{inc}$,
     - $\epsilon_c > 0$,
     - $0 < \bar{\tau} < 1$,
     - $0 < \bar{d} < \Delta_{max}$,
     - $\bar{u} \in \mathbb{N}^*$,
   - the maximum number of function evaluations (i.e. simulation runs) permitted n_{max}.
   - Define
     - α_k and β_k as the metamodel parameters at iteration k,
     - ν_k as the vector of parameters of m_k, $\nu_k = (\alpha_k, \beta_k)$,
     - n_k as the sample size,
     - u_k as the number of successive trial points rejected,
     - g_k the gradient of the Lagrangian evaluated at x_k.
   - Compute T and $\hat{f}$ at x_0, fit an initial model m_0, and compute ν_0.
   - Set $k = 0, n_0 = 1, u_0 = 0$.

1. **Criticality step.** If $\|g_k\| \leq \epsilon_c$, then switch to *conservative mode* (detailed in Section 3.4).

2. **Step calculation.** Compute a step s_k that "sufficiently reduces the model" m_k and such that $x_k + s_k \in B(x_k; \Delta_k)$ (i.e. approximately solve the TR subproblem).

3. **Acceptance of the trial point.** Compute $\hat{f}(x_k + s_k)$ and

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

   - If $\rho_k \geq \eta_1$, 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 observation in the sample set ($n_k = n_k + 1$), and fit the new model m_{k+1}.

4. **Model improvement.** Compute

$$\tau_{k+1} = \frac{\|\nu_{k+1} - \nu_k\|}{\|\nu_k\|}. \tag{6}$$

If $\tau_{k+1} < \bar{\tau}$, then improve the model by sampling a new point x, evaluate T and $\hat{f}$ at x. Include this point in the sample set ($n_k = n_k + 1$). Update m_{k+1}.

5. **Trust region radius update.**

   - If $\rho_k > \eta_1$, then increase the trust region radius: $\Delta_{k+1} = \min\{\gamma_{inc}\Delta_k, \Delta_{max}\}$.
   - Otherwise,
     - if $u_k \geq \bar{u}$, then reduce the trust region radius: $\Delta_{k+1} = \max\{\gamma\Delta_k, \Delta_{min}\}, u_k = 0$,
     - otherwise, $\Delta_{k+1} = \Delta_k$.
   - If $\Delta_{k+1} \leq \bar{d}$, then switch to *conservative mode.*

Set $n_{k+1} = n_k, u_{k+1} = u_k$.
Set $k = k + 1$.
If $n_k < n_{max}$, then go to Step 1.

3.4 Algorithmic details

Criticality step The criticality step of the algorithm ensures that if the measure of stationarity goes under a given threshold, then the model can be improved so that its stationarity measure can be trusted. The model is then said to be *certifiably fully linear* (i.e. it satisfies Taylor-type bounds). We assume throughout that we cannot certify whether the model is fully linear. If at a given iteration, the measure of stationarity does go under the criticality threshold then a purely quadratic metamodel along with an appropriate sampling strategy (e.g. Monte Carlo, Quasi-Monte Carlo) can be used in order to obtain an accurate gradient estimate and to certify full linearity. This is denoted as the *conservative mode* in the algorithm.

Step calculation Details regarding the TR subproblem are given for the traffic signal control problem in Section 4.2. .

Model improvement step At each iteration, we run the simulator at the trial point, $x_k + s_k$ (Step 3 of the algorithm). In order to diversify the set of sampled points, we may sample points other than the trial points. This step attempts to improve the accuracy of the model, by improving the geometric properties of the sampled space (e.g. attempting to fully span the feasible space such that a full rank least squares matrix is obtained, or in the case of interpolation methods improving the poisedeness of the sample (Conn et al., 2009b)). We do so if the condition $\tau_{k+1} < \bar{\tau}$ is satisfied. To sample we draw uniformly from the feasible space.

TR radius update In the Conn et al. (2009a) algorithm the TR radius can be reduced if the model is *fully linear* but has not performed well. Since we assume throughout that we cannot certify whether the model is *fully linear*, we reduce the TR radius after $\bar{u}$ successive trial points have been rejected. If the TR radius reaches a lower bound $\bar{d}$, then a quadratic polynomial with an appropriate sampling strategy is used, and as mentioned previously, we can ensure that within a uniformly bounded number of sampling steps the model will be *fully linear*.

Algorithmic parameters The following values are used for the parameters of the TR algorithm:

- $\Delta_{max} = 10^{10}$,
- $\Delta_0 = 10^3$,
- $\eta_1 = 10^{-3}$,
- $\gamma = 0.9$,
- $\gamma_{inc} = 1.2$,
- $\epsilon_c = 10^{-6}$,
- $\bar{\tau} = 0.1$,
- $\bar{d} = 10^{-2}$,
- $\bar{u} = 10$,
- $w_0 = 0.1$.

Typical values for TR parameters are given in Carter (1986). For the algorithm used to solve the TR subproblem we set the tolerance for relative change in the objective function to 10^{-3} and the tolerance for the maximum constraint violation to 10^{-2}.

4 Optimization problem

4.1 Traffic signal control

We illustrate the use of this framework with a signal control problem for a subnetwork of the city of Lausanne. A review of the different formulations, as well as the definitions of the traffic signal terms used hereafter, is given in Osorio and Bierlaire (2008). We consider the same problem as in Osorio and Bierlaire (2008), i.e. we consider a fixed-time signal control problem where the offsets, the cycle times and the all-red durations are fixed. The stage structure is also given. In other words, the set of lanes associated with each stage as well as the sequence of stages are both known. To formulate this problem we introduce the following notation:

b_i	available cycle ratio of intersection i;
$x(j)$	green split of phase j;
x_L	vector of minimal green splits;
$\mathscr{I}$	set of intersection indices;
$\mathscr{P}_I(i)$	set of phase indices of intersection i.

The problem is traditionally formulated as follows:

$$\min_{x,z} E[f(x,z;p)] \tag{7}$$

subject to

$$\sum_{j \in \mathscr{P}_I(i)} x(j) = b_i, \ \forall i \in \mathscr{I} \tag{8}$$

$$x \geq x_L. \tag{9}$$

The decision vector x consists of the green splits for each phase. The objective is to minimize the expected travel time (Equation (7)). The linear constraints (8) link the green times of the phases with the available cycle time for each intersection. The bounds (9) correspond to minimal green time values for each phase. These have been set to 4 seconds according to the Swiss transportation norm (VSS, 1992).

As detailed by Conn et al. (2009b), DF TR methods are a relatively recent topic. The algorithms developed so far are derived based on sound theoretical properties that lead to a solid global convergence theory, but they are mostly formulated for unconstrained problems. Unfortunately, the optimization problems encountered in practice are rarely unconstrained. Conn et al. (2009b) review constrained DF algorithms, and confirm that for constrained problems "currently, there is no convergence theory developed for TR interpolation-based methods", not to mention TR methods that allow for regression models.

Conn et al. (1998) propose a method to solve problems with general constraints using an unconstrained TR algorithm. The traffic management problems that we are interested in solving fall into the category of what they denote as *easy* constraints. These are general constraints that are continuously differentiable and whose first order partial derivatives can be computed relatively cheaply (with regards to the cost of evaluating the objective function). In their approach, they include such constraints in the TR subproblem, which ensures that all trial points are feasible. Conn et al. (2009b) mention that such an approach is often sufficient in practice.

Here we use the TR algorithm proposed by Conn et al. (2009a) for unconstrained methods, and extend its use to constrained problems as Conn et al. (1998) suggest. That is, we include the constraints in the TR subproblem to ensure that all trial points are feasible. The next section formulates the TR subproblem.

4.2 Trust region subproblem

At a given iteration k the TR subproblem includes three more constraints than the previous problem. It is formulated as follows:

$$\min_{x,y} \; m_k = \alpha_k T(x, y; q) + \phi(x; \beta_k) \tag{10}$$

subject to

$$\sum_{j \in \mathscr{P}_I(i)} x(j) = b_i, \; \forall i \in \mathscr{I} \tag{11}$$

$$h_2(x, y; q) = 0 \tag{12}$$

$$\|x - x_k\|_2 \leq \Delta_k \tag{13}$$

$$y \geq 0 \tag{14}$$

$$x \geq x_L, \tag{15}$$

where x_k is the current iterate, Δ_k is the current trust region radius, α_k and β_k are the current metamodel parameters, and h_2 denotes the queueing model. Equation (12) represents the queueing model, which consists of a system of nonlinear equations. These equations are given explicitly in Osorio and Bierlaire (2009b) (Equations (1)-(6),(9) in that paper). The endogenous variables of the queueing model are subject to positivity constraints (Equation (14)). The analytical form of T is also detailed in Osorio and Bierlaire (2009b). Constraint (13) is the TR constraint. It uses the Euclidean norm (Conn et al., 2009a). Thus the TR subproblem consists of a nonlinear objective function subject to nonlinear and linear equalities, a nonlinear inequality and bound constraints. This problem is solved with the Matlab routine for constrained nonlinear problems, *fmincon*, which resorts to a sequential quadratic programming method (Coleman and Li, 1996; Coleman and Li, 1994).

4.3 Signal plan features

Sampling. The model improvement step of the algorithm attempts to diversify the set of sampled points by drawing points uniformly from the feasible space. A feasible signal plan is defined by Equations (8) and (9) (or equivalently Equations (11) and (15)). We draw uniformly from this space, using the code of Stafford (2006). Given this signal plan, we solve the network model (Equation (12)) following the procedure described in Osorio and Bierlaire (2009a).

Explanatory/independent variables. The polynomial component of the metamodel, ϕ, is a quadratic polynomial in the decision variables x, which are the phase variables of the different intersections. For a given intersection the phase variables are linked through the linear Equation (8). To reduce the correlation between the explanatory variables of the metamodel, we exclude one phase per intersection. Thus for a set of i intersections and p phases, the polynomial is a function of $p - i$ phase variables, and has a total of $2(p - i) + 1$ coefficients.

5 Empirical analysis

In this section, we evaluate the performance of the proposed method on two Lausanne city subnetworks. Firstly, we consider a simplified demand distribution. Secondly, we analyze the performance of the method given the demand of the city of Lausanne for the evening peak hour, and control the plans of a larger set of intersections.

To refer to the metamodel or to its components we use the notation of Equation (2). In both sections, we compare the performance of the proposed metamodel, m, to that of two other metamodels:

- a quadratic polynomial with diagonal second derivative matrix, (i.e. the metamodel consists of ϕ),
- the queueing model (i.e. the metamodel consists of T). This is the procedure proposed in Osorio and Bierlaire (2008). Namely, this procedure uses the same algorithm as the one used to solve the TR subproblem.

5.1 Lausanne subnetwork with simplified demand distribution

We consider the Lausanne road network with a simplified demand distribution. We control a set of two adjacent signalized intersections. Demand arises at the nine centroids nearest to these two intersections. The simulation setup considers a 20 minute scenario, preceded by a 15 minute warm-up time.

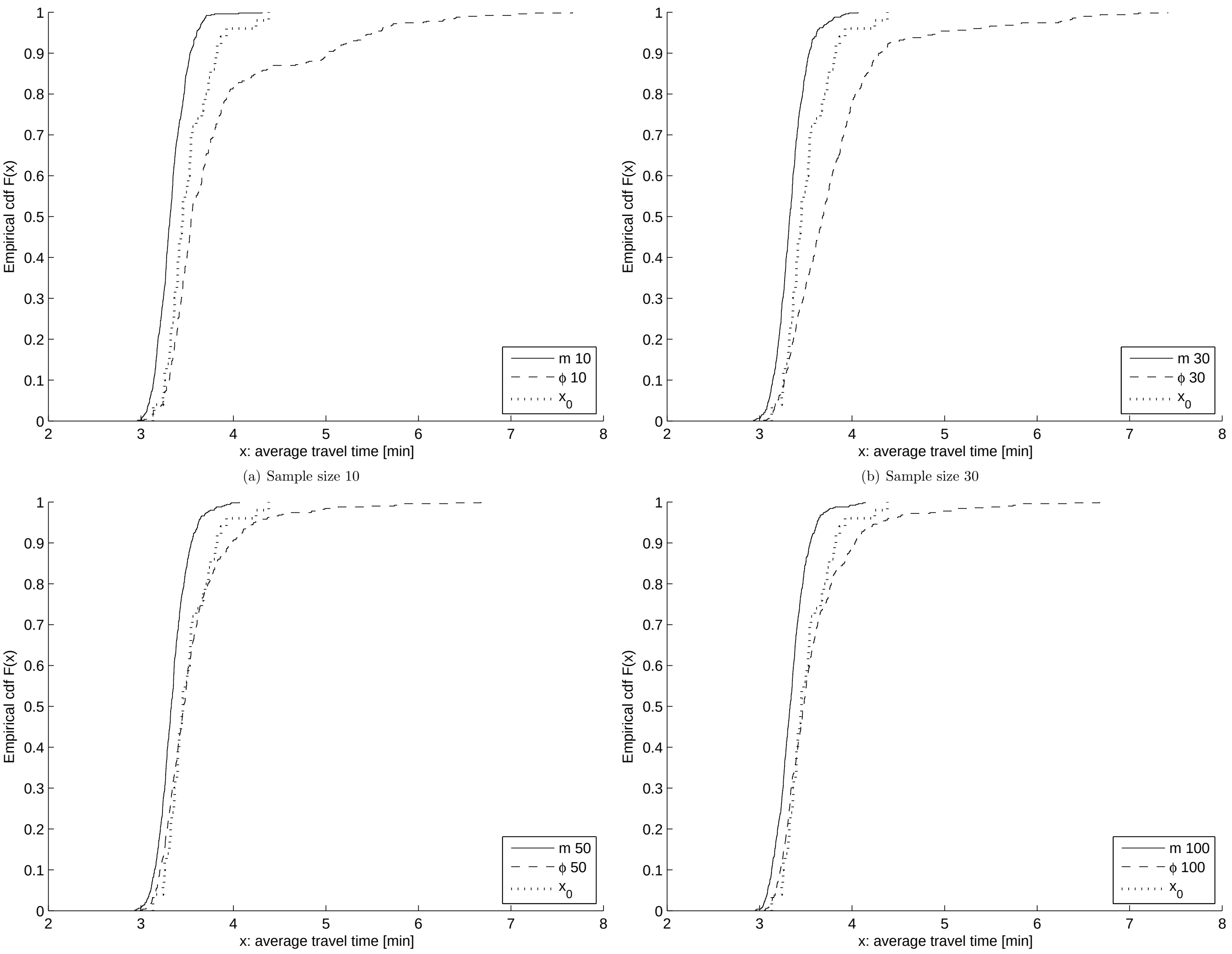
A total of 13 phases are considered variable (i.e. the dimension of the decision vector is 13). This leads to a polynomial with 23 coefficients. The queueing model considers 12 roads that are connected to either of these two intersections. These roads are modeled as a set of 21 queues. The corresponding TR subproblem consists of 131 endogenous variables with their corresponding lower bound constraints, 84 nonlinear and 36 linear equalities.

Firstly, we consider a tight computational budget, which is defined as a maximum number of simulation runs that can be carried out. The computational budget is set to 150 runs. We consider the performance of the proposed metamodel m and of the polynomial ϕ. For both metamodels, we run the algorithm 10 times. We then compare the performance of the two methods for increasing sample sizes. Initially, no simulated observations are available, i.e. we start off with an empty sample. We initialize all runs with a uniformly drawn signal plan generated with the method of Stafford (2006).

To compare the performance of both methods for a given sample size, we consider the 10 signal plans derived and evaluate their performance by running 50 replications of the simulation model. All simulations are preceded by a 15 minute warm-up period. We then compare the empirical cumulative distribution function (cdf) of the average travel times over all 10 signal plans and 50 replications, i.e. each cdf consists of a set of 500 observations.

Figure 2 considers a set of four plots. Each plot displays the initial plan, and the plans derived by both methods for a given sample size. Each plot considers a different sample size. Plots 2(a)-2(d) consider, respectively, sample sizes 10, 30, 50 and 100. The plans m, ϕ and x_0 denote, respectively, the plans derived by the proposed metamodel, the polynomial metamodel and the initial random plan. The numbers denote the corresponding sample sizes, e.g. the cdf labeled "ϕ 10" corresponds to the signal plans proposed by the polynomial with a sample of size 10.

For sample sizes 10 and 30 (plots 2(a) and 2(b)), the proposed method leads to signals plans with improved average travel times when compared to both the intial plan and the plans proposed by the polynomial. Furthermore, the latter plans do not provide improvement when compared to the initial plan.



(a) Sample size 10

(b) Sample size 30

(c) Sample size 50

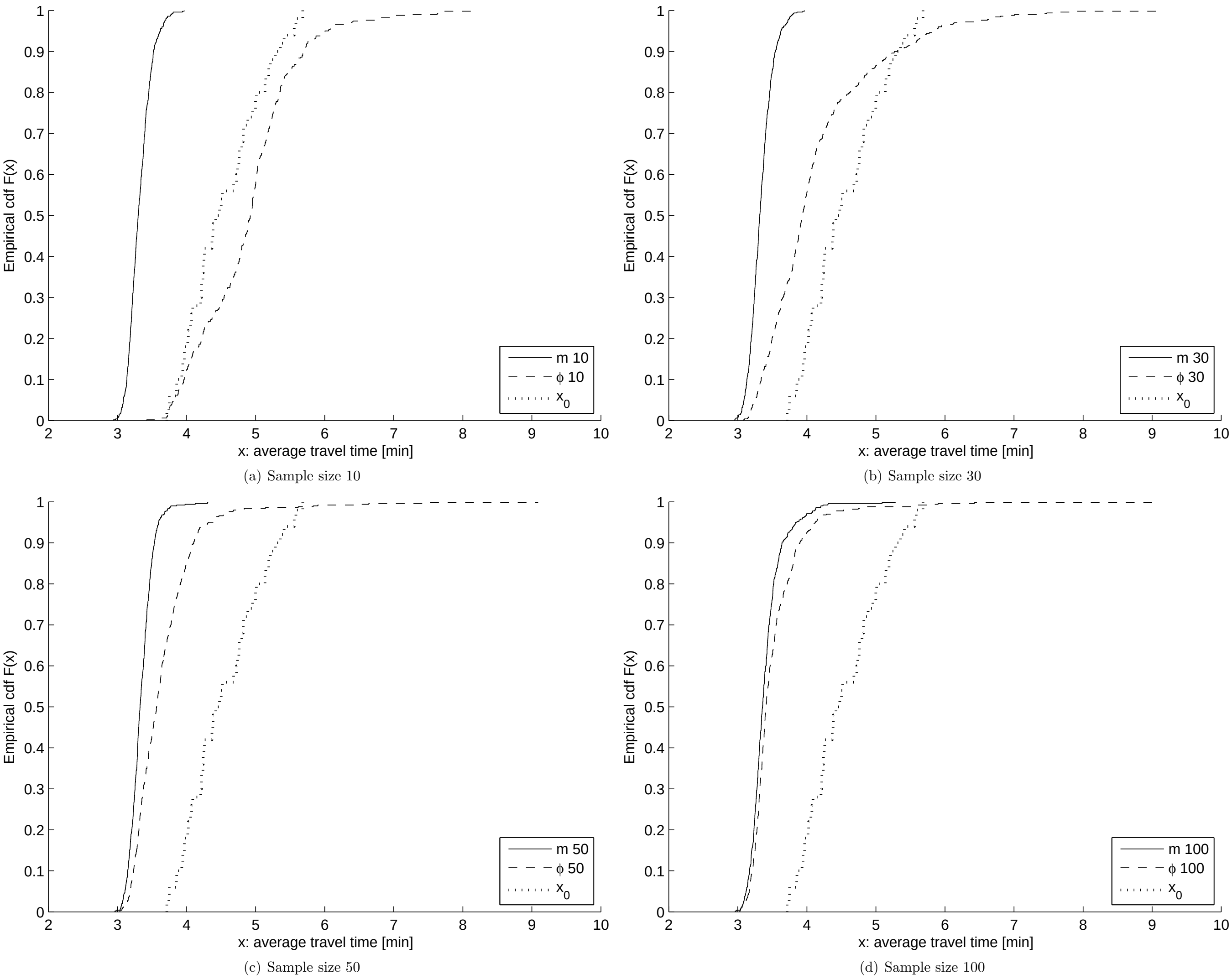
(d) Sample size 100

Figure 2: Empirical cdf's of the average travel times considering an initial random signal plan

For sample size 50 and 100 (plots 2(c) and 2(d)), the proposed method leads to signals plans with improved average travel times when compared to both the intial plan and the plans proposed by the polynomial. The polynomial method leads to signal plans with similar performance compared to the intial plan.

Figure 3 considers the same experiments with a different intial plan. In this case, the conclusions for sample size 10 (plot 3(a)) remain the same as before. As of sample size 30 (plot 3(b)), the polynomial identifies signal plans with improved performance compared to the initial plan. The proposed method outperforms both the polynomial and the initial plans. This is also the case for sample size 50 (plot 3(c)). For a sample of size 100 (plot 3(d)) the plans derived by the polynomial and the proposed method have similar performance. These four plots also illustrate how the performance of the plans derived by the polynomial improves with the sample size.

In all these cases with tight computational budgets, the proposed method leads to signal plans with very similar performance and improved performance compared to the plans derived by the polynomial and to the initial plan.



(a) Sample size 10

(b) Sample size 30

(c) Sample size 50

(d) Sample size 100

Figure 3: Empirical cdf's of the average travel times considering an initial random signal plan

Secondly, we allow for a larger computational budget. We allow for a total of 750 simulation runs. We run each method once. As before, we initialize both methods with a uniformly drawn initial signal plan and start off with an empty sample.

Figure 4 considers the signal plans derived by the polynomial method and the proposed method at sample sizes $10, 20, 30$ and 40. For each signal plan, the figure displays the empirical cdf of the average travel times over the 50 replications. The plans m, ϕ and x_0 denote, respectively, the plans derived by the proposed metamodel, the polynomial metamodel and the initial random plan. The signal plan derived by the proposed method is the same at sample sizes $10, 20, 30$ and 40. At a sample size of 10 both m and ϕ lead to improved average travel times, when compared to the initial plan. As the sample size increases, the polynomial leads to plans with improved performance. At sample size 40, its performance is similar to that of the signal plan proposed by m.

Figure 5 considers the signal plans proposed at sample sizes $40, 50, 100, 250, 500$ and 750. For each signal plan, the cdf of the average travel times over the 50 replications are displayed. This figure shows that the performance of the plans is similar for sample sizes larger than 40.

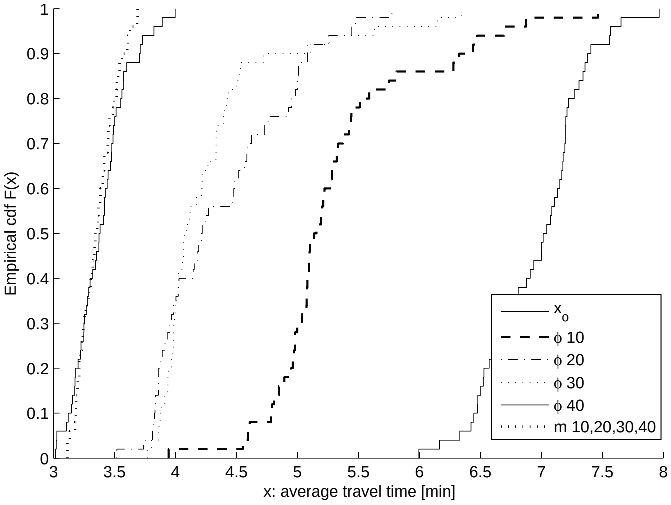


Figure 4: Empirical cumulative distribution functions of the average travel times considering an initial random signal plan and evaluating the performance as the sample size increases from 10 to 40.

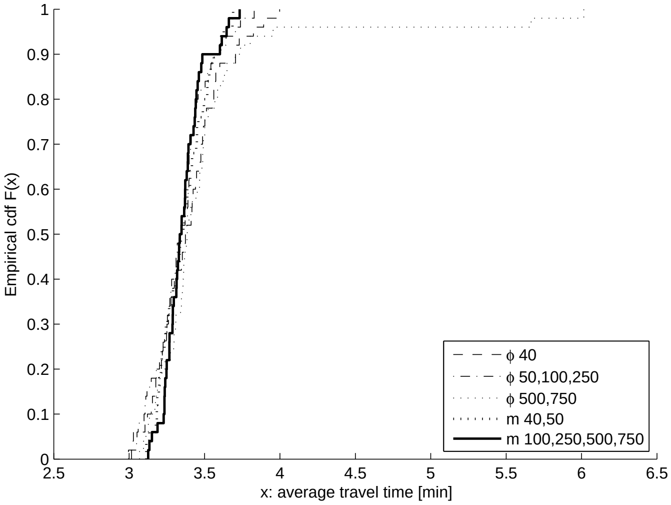


Figure 5: Empirical cumulative distribution functions of the average travel times considering an initial random signal plan and evaluating the performance as the sample size increases from 40 to 750.

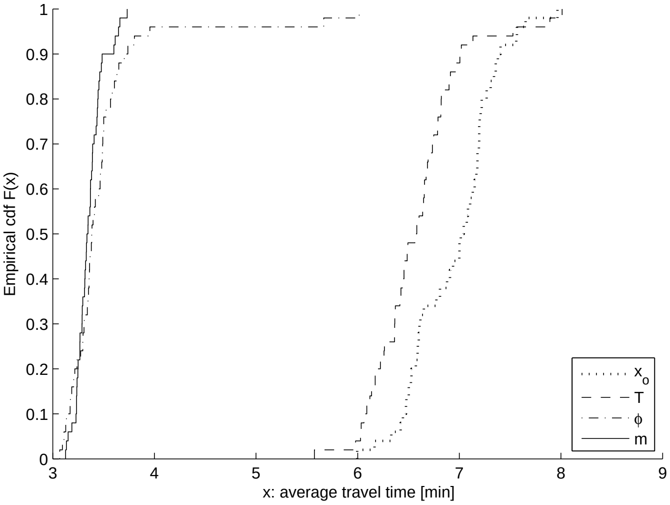


Figure 6: Empirical cumulative distribution functions of the average travel times considering an initial random signal plan and evaluating the performance at sample size 750.

Figure 6 considers the signal plans proposed by m and ϕ at sample size 750, as well as the initial signal plan and the signal plan proposed by the queueing model T. It displays for each method the cdf of the average travel times. All three methods, m, ϕ and T, lead to improved performance compared to the random signal plan. The methods that use simulated observations throughout the optimization process, m and ϕ, lead to improved signal plan performance when compared to the queueing method T.

We consider the full sample (750 observations) and test whether the metamodel parameters of the proposed model are significantly different from zero. To do so we perform a t-test. The null hypothesis assumes that the parameters are equal to zero, whereas the alternative hypothesis assumes they differ. We set the confidence level of the test to 0.05. The corresponding critical value is 1.96. Recall that there are 23 parameters. Nine are significantly different from zero. These 9 parameters concern 5 linear terms, 2 quadratic terms, the queueing model parameter (α) and the intercept (β_1). This indicates that the proposed metamodel indeed captures information about the relationship between the observed travel times and the phase variables.

The results of this section indicate that for small to moderate sample sizes (compared to the dimension of the decision vector) the structural information provided by the queueing model leads to well-performing signal plans. As the sample size increases the polynomial metamodel improves its accuracy, and achieves with a moderate sample size similar performance to that of the proposed metamodel. By comparing the proposed model to the queueing model, the results indicate that the simulated observations indeed improve the accuracy of the model, leading to signal plans that reduce the average travel times.

5.2 Lausanne subnetwork with evening peak hour demand

We evaluate the performance of the proposed method by considering a subnetwork of the Lausanne city center. The subnetwork is presented in Osorio and Bierlaire (2008). The considered scenario consists of the evening peak period (17h-18h). The simulation outputs used both to fit the metamodel

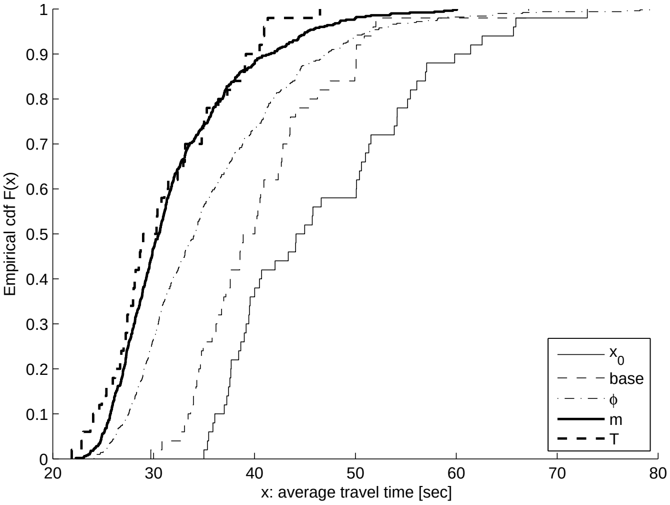


Figure 7: Empirical cumulative distribution functions of the average travel times considering an initial random signal plans and running 10 instances of each method. Each instance consists of 150 simulation runs.

and to evaluate the performance of the derived signal plans are the subnetwork average travel times.

The queueing model of this subnetwork consists of 102 queues. The TR subproblem consists of 621 endogenous variables with their corresponding lower bound constraints, 408 nonlinear equality constraints, 171 linear equality constraints and 1 nonlinear inequality constraint.

Note that this problem is considered a large-scale problem for existing unconstrained DF methods, not to mention the added complexity of the nonlinear constraints. In particular, the problem has 51 decision variables. Thus if one were to resort to a classical interpolation-based quadratic polynomial surrogate, 1378 function evaluations would be necessary to fit the full polynomial. This is because for a problem with n decision variables $(n + 1)(n + 2)/2$ suitably sampled points (i.e. well poised (Conn et al., 2000; Conn et al., 2009b)) are necessary to fit the full quadratic.

We consider a tight computational budget, which is defined as a maximum number of simulation runs that can be carried out, and no initial observation available. The computational budget is set to 150 runs. For a given initial signal plan, we run the corresponding algorithm 10 times, deriving 10 signal plans. We then evaluate the performance of each of these signal plans by running 50 replications of the simulation model. All simulations are preceded by a 15 minute warm-up period. To compare the methods, we consider the empirical cdf of the average travel times over all 10 signal plans and 50 replications, i.e. each cdf consists of a set of 500 observations.

Firstly, we consider the performance of these methods given a uniformly drawn initial signal plan, which we generate with the method of Stafford (2006). The plot of Figure 7 considers a random initial plan and presents the cdf's of the average travel times. The plans m, T, ϕ and x_0 denote, respectively, the plans derived by the proposed metamodel, the queueing model, the polynomial metamodel and the initial random plan. The plan denoted by *base plan* is an existing signal plan for the city of Lausanne.

Figure 7 indicates that all methods have an improved performance when compared to both the base and the initial plans. Both the proposed metamodel and the queueing model derive signal plans with improved performance compared to the polynomial.

	Average [sec]	Standard deviation
m	32.16	6.55
ϕ	35.96	8.57
T	31.08	5.76
x_0	46.8	9.50

Table 1: Travel time statistics for the different signal plans, obtained based on 500 observations. The methods have been initialized considering the initial plan x_0.

	m	ϕ	x_0
ϕ	7.88		
x_0	28.38	18.96	
T	-2.76	-10.57	-31.66
degrees of freedom: 998			
confidence level: 0.05			
hypothesized mean difference: 0			

Table 2: T-statistics assuming equal variance. Each statistic is calculated based on the difference between the corresponding row and column methods. This scenario considers a random initial plan.

The travel time statistics for the different methods are displayed in Table 1. To test whether the difference in the average travel times is significant we perform a t-test. We assume that the observed average travel times arise from a normal distribution with common but unknown variance. The null hypothesis assumes that the expected travel time is the same for both methods, whereas the alternative hypothesis assumes that they differ. The t-statistics are given in Table 2. Each statistic is calculated based on the difference between the row-wise method and the column-wise method. The critical value is 1.96. This table indicates that all differences are significant. That is, the queueing approach leads to signal plans with the best performance, followed by the proposed approach, the polynomial and the initial plan.

Secondly, we use the existing signal plan for the city of Lausanne (the base plan) as the initial plan. Once again, we run each method 10 times, and allow for 150 simulations each time. Figure 8 gives the cdf's of the different methods. Here the proposed method and the polynomial lead to signal plans with similar performance and improved performance compared to the base plan. The queueing model leads to a signal plan with the best performance in terms of the distribution of the average travel times.

The corresponding travel time statistics are displayed in Table 3. We perform t-test's to evaluate whether the difference in the average travel times is significant. The t-statistics are given in Table 4. Each statistic is calculated based on the difference between the row-wise method and the column-wise method. The critical value is 1.96. Once again, all differences are significant and we have the same ranking between the methods, that is: the queueing approach, the proposed metamodel, the polynomial and the base plan.

We now consider a scenario with a higher computational budget. We allow for 1000 simulation runs and consider a random initial point. In this case, we run the algorithm once. We then evaluate

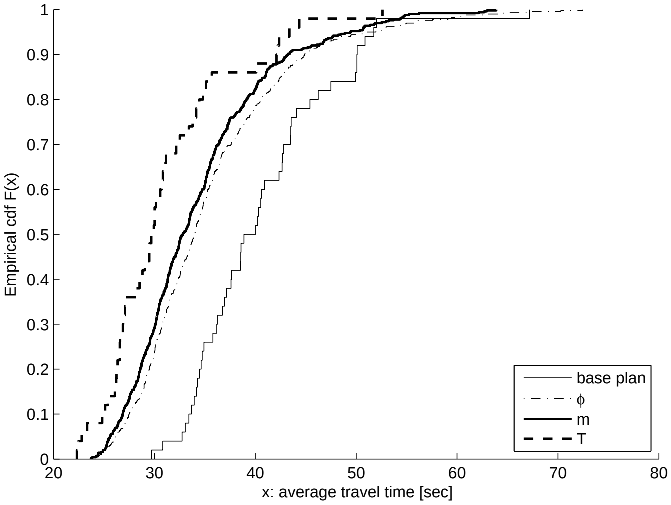


Figure 8: Empirical cumulative distribution functions of the average travel times considering the base plan as the initial point and running 10 instances of each method. Each instance consists of 150 simulation runs.

	Average [sec]	Standard deviation
m	34.35	7.09
ϕ	35.5	7.78
T	31	6.29
base	40.65	6.95

Table 3: Travel time statistics for the different signal plans, obtained based on 500 observations. The methods have been initialized considering the base plan as the initial plan.

	m	ϕ	*base*
ϕ	2.44		
base	14.18	11.04	
T	-7.91	-10.06	-23.01
degrees of freedom: 998			
confidence level: 0.05			
hypothesized mean difference: 0			

Table 4: T-statistics assuming equal variance. Each statistic is calculated based on the difference between the corresponding row and column methods. This scenario considers the base plan as the initial plan.

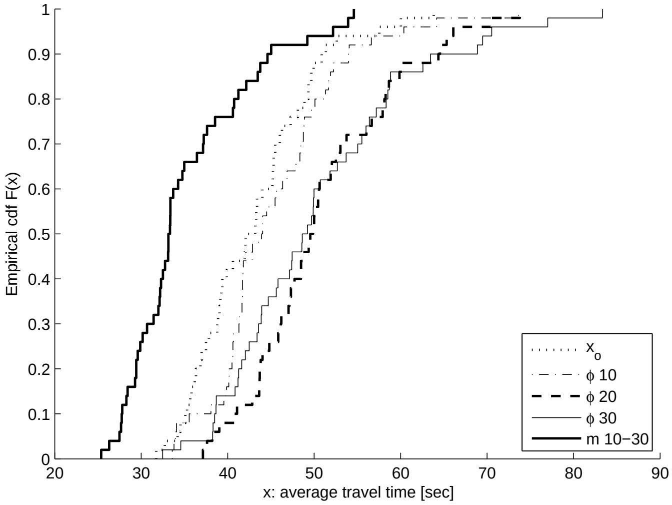


Figure 9: Empirical cumulative distribution functions of the average travel times considering a random signal plan as the initial point. The signal plans displayed are those derived at sample sizes 10, 20 and 30.

the performance of the derived plans by running 50 replications of the simulation model.

Figure 9 presents the cdf's of the average travel times across the 50 replications, considering the initial plan, and the plans derived by both the proposed and the polynomial method at sample sizes $10, 20$ and 30. The plan proposed by the polynomial method at sample size 10 has similar performance compared to the initial plan, whereas the plans at sample sizes 20 and 30 perform less well than the initial plan. The proposed method leads to the same plan for all three sample sizes. This plan has improved performance when compared to the initial plan and to the plans derived by the polynomial method.

Figure 10 presents the cdf's of the average travel times across the 50 replications, considering the initial plan, and the plans derived by both the proposed and the polynomial method at sample sizes 40 and 50. The signal plans derived by the polynomial do not provide improvement. The proposed method leads to the same plan for both sample sizes. This plan improves the distribution of average travel times compared to both the polynomial and the initial plan. Figure 11 considers sample sizes $250, 500$ and 750. With a sample of size 250 the polynomial leads to reduced travel times compared to the proposed method. For the other sample sizes the signal plans of both methods have similar performance.

Figure 12 considers the plans proposed at sample size 1000, and compares them to the initial plan, as well as to the plan proposed by the queueing method. All methods have an improved performance when compared to the initial plan. The plans derived by the proposed and the polynomial methods have similar performance, while the queueing model performs best.

The travel time statistics are displayed in Table 5. We perform t-test's to evaluate whether the difference in the average travel times is significant. The t-statistics are given in Table 6. Each statistic is calculated based on the difference between the row-wise method and the column-wise method. The critical value is 1.98. This table indicates that all three methods lead to improved performance compared to the initial plan. The difference in performance between the three methods is not significant.

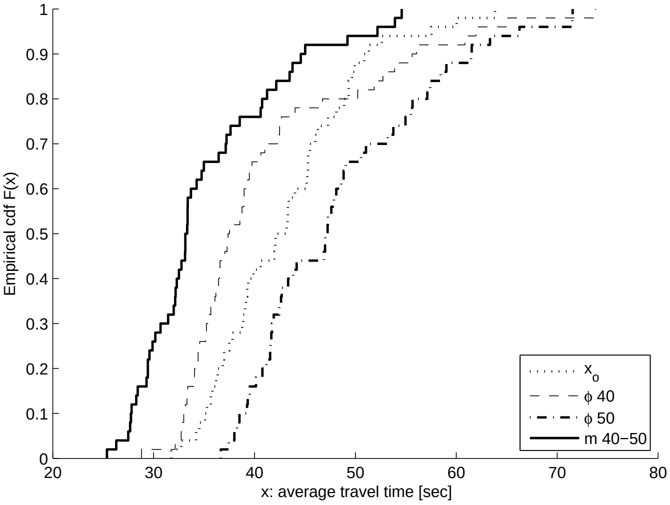


Figure 10: Empirical cumulative distribution functions of the average travel times considering a random signal plan as the initial point. The signal plans displayed are those derived at sample sizes 40 and 50.

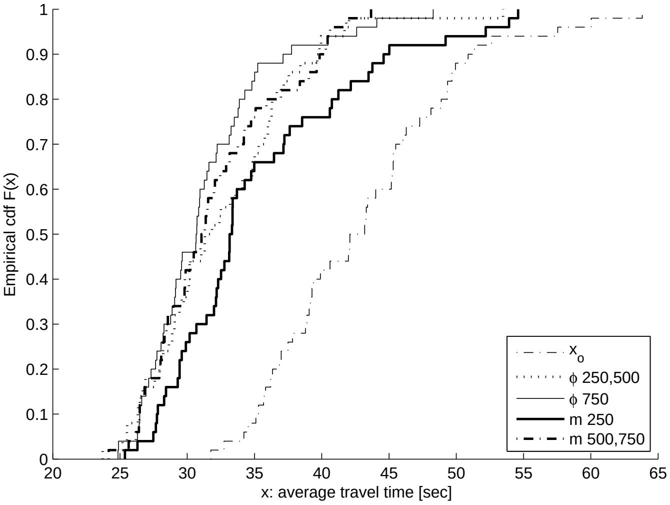


Figure 11: Empirical cumulative distribution functions of the average travel times considering a random signal plan as the initial point. The signal plans displayed are those derived at sample sizes 250, 500 and 750.

	Average [sec]	Standard deviation
m	32.01	4.9
ϕ	32.47	6.47
T	30.84	6.42
x_0	43.06	7

Table 5: Travel time statistics for the different signal plans, obtained based on 50 replications. The methods have been initialized considering a uniformly drawn random initial plan.

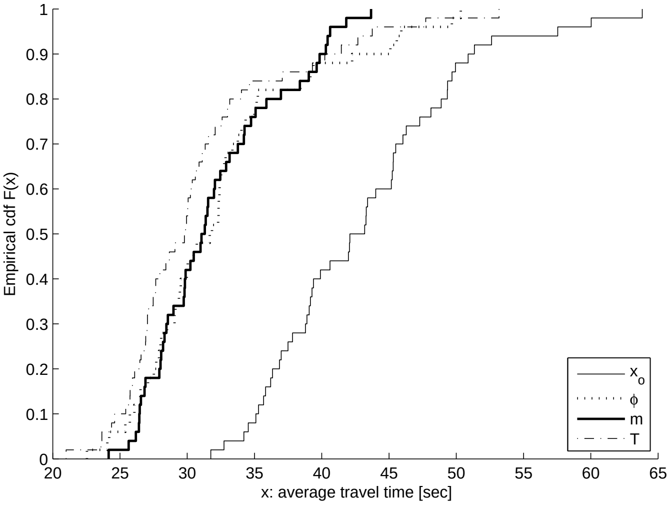


Figure 12: Empirical cumulative distribution functions of the average travel times considering a random signal plan as the initial point. The signal plans displayed are those obtained with a sample of size 1000.

	m	ϕ	x_0
ϕ	0.39		
x_0	9.14	7.86	
T	-1.04	-1.27	-9.1
degrees of freedom: 98			
confidence level: 0.05			
hypothesized mean difference: 0			

Table 6: T-statistics assuming equal variance. Each statistic is calculated based on the difference between the corresponding row and column methods. This scenario considers a random plan as the initial point.

We consider this sample of 1000 observations and test whether the metamodel parameters of the proposed model are significantly different from zero. We perform the corresponding t-tests as described in Section 5.1. In this case there are 86 model parameters. Seven are significantly different from zero. These 7 parameters concern 3 linear terms, 2 quadratic terms, the queueing model parameter (α) and the intercept (β_1). This indicates that the proposed metamodel indeed captures information regarding the relationship between the observed travel times and the decision variables.

The results of this section indicate that for small sized samples (compared to the dimension of the decision vector) limiting the metamodel of this framework to a quadratic polynomial fails to provide well-performing trial points, whereas providing structural information analytically via the queueing model allows for improvement. For small sized samples the queueing approach has the best performance, followed by the proposed method, the polynomial and the initial plans. We also considered one instance with a larger sample. Here all three methods yield similar performance and improved performance compared to the initial plan.

6 Conclusions

This paper presents a simulation-based optimization framework for the management of congested networks. It proposes a metamodel that combines information from a traffic simulation tool and an analytical network model. It integrates this metamodel within a derivative-free trust region optimization algorithm.

Firstly, the performance of this framework is evaluated by solving a fixed-time signal control problem for two intersections of the Lausanne city center, and a simplified demand distribution. For small to moderate sized samples (small compared to the dimension of the decision vector), the metamodel provides improvement when compared to the polynomial method. For larger samples, the proposed metamodel leads to signal plans that perform similarly to those of the polynomial. For larger samples, both methods yield reduced travel times when compared to the plan proposed by the queueing model and to the initial plan.

Secondly, we consider the demand of the Lausanne city network for the evening peak period (17-18h). A larger set of intersections is considered endogenous. The performance is evaluated considering as initial plans: random plans and also an existing plan for the city of Lausanne.

In this case we ran two types of experiments. Firstly, we assumed a tight computational budget (approximately three times the dimension of the decision vector) and ran 10 instances of the algorithm. We initialized the algorithm both with a random plan and with an existing plan for the city of Lausanne. In both cases, the queueing approach leads to the signal plans with best performance, followed by the proposed metamodel, the polynomial and the initial plans.

We then allowed for a large computational budget and initialized with a random initial plan. We analyzed the performance of the signal plans with increasing sample sizes. For small samples, the proposed method yields well-performing plans, unlike the polynomial. For large samples, the proposed and the polynomial methods derive signal plans with similar performance. Their performance is also similar to that of the queueing approach.

Efficiently tackling unconstrained high dimensional problems (e.g. more than 100 variables) is one of the main limitations of existing derivative-free methods, not to mention the added complexity of constrained problems. The generic metamodels used in these algorithms, e.g. quadratic polynomials, require a moderate to large sample to initially fit the metamodel of interest. By combining these generic metamodels with application-specific models that analytically capture the structure of the underlying problem, these algorithms can be used to tackle high dimensional problems under tight computational budgets. This added structure overcomes the need for a substantial initial sample, and provides meaningful trial points since the very first iterations.

For the Lausanne subnetwork scenario and a tight computational budget, it is the queueing model that leads to the best performance. Nonetheless, it is of interest to preserve the functional or generic component of the metamodel, since it ensures the asymptotic convergence of the algorithm. More research is needed to investigate whether other functional forms (e.g. splines, radial basis functions), or other interaction terms can improve the accuracy of the metamodel. We will also investigate the sensitivity of the method to the numerous algorithmic parameters.

Given the good performance of the queueing model, a natural extension would be to use the simulated observations to improve the accuracy of the exogenous queueing model parameters. This can be done by iteratively calibrating these parameters as the observations are collected. In particular, we expect the calibration of the exogenous parameters that depend on the decision vector, such as the turning proportions, to further improve the models accuracy.

References

Alexandrov, N. M., Dennis, J. E., Lewis, R. M. and Torczon, V. (1998). A trust region framework for managing the use of approximation models in optimization, *Structural Optimization* **15**: 16–23.

Alexandrov, N. M. and Lewis, R. M. (2001). An overview of first-order model management for engineering optimization, *Optimization and Engineering* **2**: 413–430.

Alexandrov, N. M., Lewis, R. M., Gumbert, C. R., Green, L. L. and Newman, P. A. (1999). Optimization with variable-fidelity models applied to wing design, *Technical Report CR-1999-209826*, NASA Langley Research Center, Hampoton, VA, USA.

Alexandrov, N. M., Lewis, R. M., Gumbert, C. R., Green, L. L. and Newman, P. A. (2001). Approximation and model management in aerodynamic optimization with variable-fidelity models, *Journal of Aircraft* **38**(6): 1093–1101.

Alexandrov, N. M., Nielsen, E. J., Lewis, R. M. and Anderson, W. K. (2000). First-order model management with variable-fidelity physics applied to multi-element airfoil optimization, *Proceedings of the 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Long Beach, CA, USA.

Atkeson, C. G., Moore, A. W. and Schaal, S. (1997). Locally weighted learning, *Artificial Intelligence Review* **11**: 11–73.

Bandler, J. W., Cheng, Q., Dakroury, A., Mohamed, A., Bakr, M., Madsen, K. and Søndergaard, J. (2004). Space mapping: The state of the art, *IEEE Transactions on Microwave Theory and Techniques* **52**(1): 337–360.

Bandler, J. W., Koziel, S. and Madsen, K. (2006). Space mapping for engineering optimization, *SIAG/Otimization Views-and-News* **17**(1): 19–26.

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

Booker, A. J., Dennis, J. E., Frank, P. D., Serafini, D. B., Torczon, V. and Trosset, M. W. (1999). A rigorous framework for optimization of expensive functions by surrogates, *Structural Optimization* **17**: 1–13.

Carter, R. G. (1986). *Multi-model algorithms for optimization*, PhD thesis, Rice University.

Coleman, T. F. and Li, Y. (1994). On the convergence of reflective newton methods for large-scale nonlinear minimization subject to bounds, *Mathematical Programming* **67**(2): 189–224.

Coleman, T. F. and Li, Y. (1996). An interior, trust region approach for nonlinear minimization subject to bounds, *SIAM Journal on Optimization* **6**: 418–445.

Conn, A. R., Gould, N. I. M. and Toint, P. L. (2000). *Trust-region methods*, MPS/SIAM Series on Optimization, Society for Industrial and Applied Mathematics and Mathematical Programming Society, Philadelphia, PA, USA.

Conn, A. R., Scheinberg, K. and Toint, P. L. (1998). A derivative free optimization algorithm in practice, *Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, St. Louis, MO, USA.

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

Conn, A. R., Scheinberg, K. and Vicente, L. N. (2009b). *Introduction to derivative-free optimization*, MPS/SIAM Series on Optimization, Society for Industrial and Applied Mathematics and Mathematical Programming Society, Philadelphia, PA, USA.

Dumont, A. G. and Bert, E. (2006). Simulation de l'agglomération Lausannoise SIMLO, *Technical report*, Laboratoire des voies de circulation, ENAC, Ecole Polytechnique Fédérale de Lausanne.

Fu, M. C. (2006). Gradient estimation, *in* S. G. Henderson and B. L. Nelson (eds), *Handbooks in operations research and management science: Simulation*, Vol. 13, Elsevier, Amsterdam, chapter 19, pp. 576–616.

Fu, M. C., Glover, F. W. and April, J. (2005). Simulation optimization: a review, new developments, and applications, *in* M. E. Kuhl, N. M. Steiger, F. B. Armstrong and J. A. Joines (eds), *Proceedings of the 2005 Winter Simulation Conference*, Piscataway, New Jersey, USA, pp. 83–95.

Moré, J. and Wild, S. (2009). Benchmarking derivative-free optimization algorithms, *SIAM Journal on Optimization* **20**(1): 172–191.

Oeuvray, R. and Bierlaire, M. (2009). Boosters: a derivative-free algorithm based on radial basis functions, *International Journal of Modelling and Simulation* **29**(1): 26–36.

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

Osorio, C. and Bierlaire, M. (2008). Network performance optimization using a queueing model, *Proceedings of the European Transport Conference (ETC)*, Noordwijkerhout, The Netherlands.

Osorio, C. and Bierlaire, M. (2009a). An analytic finite capacity queueing network model capturing the propagation of congestion and blocking, *European Journal Of Operational Research* **196**(3): 996–1007.

Osorio, C. and Bierlaire, M. (2009b). A surrogate model for traffic optimization of congested networks: an analytic queueing network approach, *Technical Report 090825*, Transport and Mobility Laboratory, ENAC, Ecole Polytechnique Fédérale de Lausanne.

Powell, M. J. D. (2003). On trust region methods for unconstrained minimization without derivatives, *Mathematical Programming* **97**(3): 605–623.

Serafini, D. B. (1998). *A Framework for Managing Models in Nonlinear Optimization of Computationally Expensive Functions*, PhD thesis, Rice University.

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

Stafford, R. (2006). *The Theory Behind the 'randfixedsum' Function.* http://www.mathworks.com/matlabcentral/fileexchange/9700.

The Mathworks, I. (2008). *Optimization Toolbox Version 4. User's Guide Matlab*, Natick, MA, USA.

TSS (2008). *AIMSUN NG and AIMSUN Micro Version 5.1*, Transport Simulation Systems.

VSS (1992). *Norme Suisse SN 640837 Installations de feux de circulation; temps transitoires et temps minimaux*, Union des professionnels suisses de la route, VSS, Zurich.

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**: 3197–3219.