

Kouvelas, A., Aboudolas, K., Kosmatopoulos, E.B., and Papageorgiou, M. (2011) *Adaptive performance optimization for large-scale traffic control systems*. IEEE Transactions on Intelligent Transportation Systems, 12 (4). pp. 1434-1445. ISSN 1524-9050

Copyright © 2011 IEEE

A copy can be downloaded for personal non-commercial research or study, without prior permission or charge

Content must not be changed in any way or reproduced in any format or medium without the formal permission of the copyright holder(s)

When referring to this work, full bibliographic details must be given

<http://eprints.gla.ac.uk/81375/>

Deposited on: 11 June 2014

Adaptive Performance Optimization for Large-Scale Traffic Control Systems

Anastasios Kouvelas, Konstantinos Aboudolas, Elias B. Kosmatopoulos, and Markos Papageorgiou, *Fellow, IEEE*

Abstract—In this paper, we study the problem of optimizing (fine-tuning) the design parameters of large-scale traffic control systems that are composed of distinct and mutually interacting modules. This problem usually requires a considerable amount of human effort and time to devote to the successful deployment and operation of traffic control systems due to the lack of an automated well-established systematic approach. We investigate the adaptive fine-tuning algorithm for determining the set of design parameters of two distinct mutually interacting modules of the traffic-responsive urban control (TUC) strategy, i.e., split and cycle, for the large-scale urban road network of the city of Chania, Greece. Simulation results are presented, demonstrating that the network performance in terms of the daily mean speed, which is attained by the proposed adaptive optimization methodology, is significantly better than the original TUC system in the case in which the aforementioned design parameters are manually fine-tuned to virtual perfection by the system operators.

Index Terms—Adaptive control, large-scale systems, parameter optimization, self-optimizing control, traffic control.

I. INTRODUCTION

CURRENTLY, a considerable amount of human effort and time is spent on the calibration of operations of large-scale traffic control systems (LSTCSs). Minor changes in the transport system infrastructure, e.g., installing a new variable message sign in a motorway network, modifying the traffic light signal phasing at an urban junction, and deploying a new bus in a public transport system or a new automated guided vehicle (AGV) in a seaport container terminal, may require the involvement of significant human effort and time to readjust and reprogram the LSTCS decision-making mechanisms. Moreover, the continuous medium- and long-term variations of the overall transport system dynamics (e.g., due to changes of

traffic demand or the number of passengers who use the particular transport system) call for a frequent or even continuous maintenance of LSTCSs, which, if properly done, makes the maintenance of LSTCSs extremely costly. In several cases, the result is that system maintenance is neglected and the system performance deteriorates year after year.

In general, the same process that is required for calibration may also be used in the initial fine-tuning of the control system during its first installation. Both tasks (initial fine-tuning and calibration) are performed (if at all) by experienced personnel due to the lack of an automated and systematic approach; thus, there is no guarantee that the overall fine-tuning or maintenance will successfully end up. In some cases, the LSTCS has never achieved a satisfactory performance in the first place, e.g., in the reported case of the traffic signal control strategy split cycle and offset optimization technique (SCOOT), which is the most popular traffic management system worldwide, in the city of Nijmegen, The Netherlands [1], where the SCOOT application was completely abandoned in the end.

Urban and motorway traffic control systems, LSTCSs for public transport systems, and LSTCSs for railway, airport, and seaport operations are all specific examples of LSTCSs that call for calibration while the system is in operation. In all of these systems, the maintenance involves the recalibration, readjustment, and reprogramming of hundreds of parameters, rules, operational schedules, and decision-making mechanisms, which influence the transport system operations in a highly complex manner. Moreover, while the system is in operation, the use of heuristics, trial-and-error methods, and experience-based techniques involves the risk of poor system performance over a lengthy period of time, which may lead to delays, severe congestion, poor quality of service (QoS), and increased green house gas (GHG) emissions during this period. Finally, note that the involvement of the human factor for the installation, maintenance, and renovation of LSTCSs also involves the risk of unsafe operations: human mistakes due to the lack of expertise and exhaustive working conditions may lead to decisions or actions that put safety at stake.

Recently, we have introduced and analyzed a new family of algorithms, hereafter called adaptive fine-tuning (AFT), that can be used toward the development of a generic, efficient, and systematic approach for the automated fine-tuning of LSTCSs [2]–[5]. The main attributes of these algorithms may be summarized as follows.

- The algorithms are based on adaptive optimization (AO) principles, and as a result, they do not require any *a priori* knowledge or assumption on the traffic system dynamics; moreover, they can be implemented to any type of LSTCS,

Manuscript received October 29, 2010; revised April 6, 2011 and May 24, 2011; accepted May 31, 2011. The Associate Editor for this paper was S. C. Wong.

A. Kouvelas is with the Department of Production and Management Engineering, Technical University of Crete, 73100 Chania, Greece, and also with the Informatics and Telematics Institute, Centre for Research and Technology Hellas, 57001 Thessaloniki, Greece (e-mail: tasos@dssl.tuc.gr).

K. Aboudolas is with the University of California, Berkeley, CA 94720 USA, on leave from the Informatics and Telematics Institute, Centre for Research and Technology Hellas, 57001 Thessaloniki, Greece (e-mail: aboud@iti.gr).

E. B. Kosmatopoulos is with the Department of Electrical and Computer Engineering, Democritus University of Thrace, 67100 Xanthi, Greece (e-mail: kosmatop@ee.duth.gr).

M. Papageorgiou is with the Department of Production and Management Engineering, Technical University of Crete, 73100 Chania, Greece (e-mail: markos@dssl.tuc.gr).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TITS.2011.2159002

regardless of the methodology used for the original design of the LSTCS.

- They are robust with respect to exogenous disturbances, noisy measurements, system interactions, and component failures.
- They are utterly generic, computationally efficient, and straightforward to embed to any type of LSTCS, regardless of its size, level of complexity, and level of decentralization.
- They incorporate powerful learning and estimation mechanisms that render them adaptable to short- and long-term variations of system characteristics such as demand long-term variations and system aging. Moreover, through these learning and estimation mechanisms, they can incorporate the knowledge captured in the past with regard to the dependence of the system performance on the controller parameters and system exogenous inputs.
- Most importantly, they guarantee a safe and efficient fine-tuning procedure, contrary to other popular AO methods that cannot exclude the possibility of poor or even unstable performance during the automatic fine-tuning.

In this paper, we investigate the AFT algorithm [4] to determine the set of design parameters of two distinct and mutually interacting modules of the traffic-responsive urban control (TUC) strategy [6]–[8], i.e., split and cycle, through a simulation-based application to the large-scale urban road network of the city of Chania, Greece, under a number of different scenarios. The comparative evaluation is based on quantitative criteria, e.g., the daily mean speed in the network, as well as specific characteristics of the split and control modules of TUC.

Following a brief review of the related literature in Section II, we present the formulation of the design parameters for the fine-tuning problem in Section III. In Section IV, we describe the employed AFT algorithm. The TUC traffic signal control strategy is briefly presented in Section V. Extensive simulation results of the AFT algorithm application to two distinct mutually interacting modules of TUC to optimize their design parameters for the urban road network of the city of Chania, Greece, are presented in Section VI. Finally, in Section VII, the conclusions of this paper are drawn, and future research directions are suggested.

II. LITERATURE REVIEW

Because there is large-enough literature on designing methodologies for the calibration of LSTCS operations, this brief review is divided into the following two sections: 1) theoretical/simulation-based methods and 2) adaptive or neural/learning methods.

A. Theoretical/Simulation-Based Methods

In the last decades, attempts have been made, particularly LSTCS applications, to develop model-based, i.e., either theoretical- or simulation-based, designs that produce “good” sets of tunable parameters, and although these approaches have helped, in some cases, to reduce the time and effort for installation and maintenance, they did not manage to eliminate or at

least significantly reduce the involvement of the human factor. One example in this class is the implementation of a variable-speed-limit system on the U.K. motorway M42 [9]. Although the initial tunable parameters of the system (which correspond to speed and flow activation/deactivation thresholds) were “optimized” using theoretical tools from the traffic flow theory and extensive simulation experiments, it took more than one year of the calibration of the aforementioned thresholds until the system has reached an acceptable performance, and during this initial deployment phase, the system performance was sometimes worse than the no-control case.

There have also been some attempts to incorporate optimization-based tools into the maintenance procedure; for example, see [10]–[15] and the references therein for an indicative list of references. In these cases, the problem of providing efficient maintenance is formulated as an optimization problem, where the tunable LSTCS parameters are chosen to optimize a performance criterion (e.g., average network speeds in traffic networks, average delays in airborne or seaborne transport systems, the total number of containers that are loaded or unloaded in seaport container terminals, and the average deviation from the operational schedules in public transport systems). However, the optimization of such a performance criterion requires perfect or, at least, very accurate knowledge of the transport system dynamics and the demand. To deal with this problem, optimization-based approaches employ simulation-based or theoretical models to represent the actual system dynamics. Then, assuming that these models quite accurately represent the actual LSTCS operations, different optimization algorithms (e.g., gradient-descent, Gauss–Newton, evolutionary programming, or neural-network-based optimization algorithms) are applied to extract the optimal values of the tunable parameters. However, these approaches have the following two disadvantages: 1) They require extensive and continuous calibration of the simulation/theoretical-based models to optimize their approximation accuracy with respect to the actual transport system operations, and 2) they face the tradeoff between simplicity and accuracy, and in most cases, accuracy has to be sacrificed to avoid the use of extreme computational requirements of simulation or mathematical models that employ a detailed modeling of the LSTCS operations.

B. Adaptive and Neural/Learning Methods

One possible way of bypassing the aforementioned problems is to incorporate adaptive or adaptive-like designs (e.g., neural, fuzzy, or iterative learning methods) to update the parameters of the LSTCS, which render several advantages contrary to the simulation/theoretical-based techniques. AFT belongs to the family of the so-called AO methods, e.g., the simultaneous perturbation stochastic approximation (SPSA) [16], [17]. These methods probably provide the most promising approach for the development of a systematic methodology for automatic, safe, robust, and efficient maintenance and renovation of LSTCSs. The basic functioning procedure for AO methods may be summarized as follows (see Fig. 1).

- At the end of appropriately defined periods (e.g., at the end of each day), the AO algorithm receives the value of

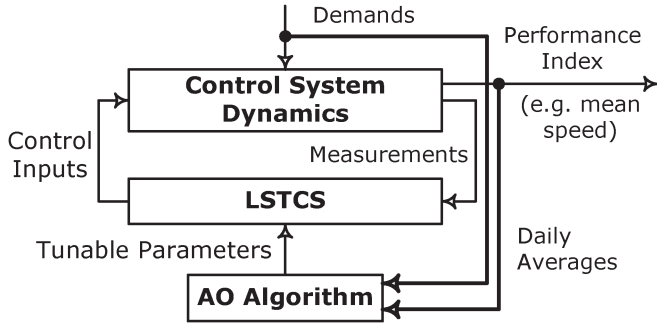


Fig. 1. AO for the automatic calibration of LSTCSs.

the real (measured) performance index (e.g., the average speed over space and time for traffic networks and the total number of containers that are loaded or unloaded for seaport container terminals) and the values of the most significant external factors (e.g., demand). Note that the performance index is an (unknown) function of the external factors and the tunable parameters to be adjusted.

- Using the measured quantities, the AO algorithm calculates new tunable parameter values to be applied at the next period (e.g., the next day) in an attempt to improve the system performance.
- This (iterative) procedure is continued over several periods (e.g., days) until a maximum in performance has been reached. Then, the AO algorithm may remain active for continuous adaptation or be switched off and reactivated at a later stage (e.g., after a few months).

One key idea behind most AO methodologies is to use two different (but interwoven) phases of tunable parameter changes as follows.

- 1) At the *perturbation phase*, the performance of the LSTCS is evaluated at one or more random perturbations of the current set of tunable parameters.
- 2) At the *gradient-descent-like phase*, the current tunable parameter values are modified in a targeted way to increase the performance based on an estimate of the gradient of the LSTCS performance. The gradient can be calculated using the values of the performance index (and the external factors) at the perturbation phase.

The random perturbations are introduced in the perturbation phase for the AO mechanisms to sufficiently explore the overall LSTCS state space (to come up with a suitable decision each time). As shown in several research articles that evaluate AO methods, the introduction of random perturbations is necessary and crucial for the successful operation of the overall scheme (see [2], [3], [16], and [17]). Different researchers reported very encouraging results, by using simulation experiments, on the application of the aforementioned AO methods in the maintenance and renovation of various LSTCSs. Urban signal traffic control [18]–[20], air traffic management [21], vessel traffic management [22], and fleet and transit management [23], [24] are few of the LSTCS maintenance applications where these methods have been applied and evaluated through extensive simulation studies.

Unfortunately, these designs suffer from the following two severe drawbacks.

- 1) Although there are some AO methods that utilize a neural network training phase to capture past knowledge (for example, see [19]), the majority of AO methods do not have any mechanism for incorporating the knowledge captured in the past with regard to the dependence of the LSTCS performance on the tunable parameters and the external factors (demand). In case that such dependence is highly nonlinear and complex, the aforementioned algorithms fail to produce any improvement of the overall LSTCS performance when applied to the baseline control modules under study (see [2] for details).
- 2) Most importantly, the use of random perturbations in the AO algorithms may lead to an unacceptable value of the LSTCS performance; even a small perturbation of a “good” set of tunable parameters may lead to an unacceptable or, even worse, an unstable or catastrophic behavior. Hence, AO methods possess the disadvantage of not guaranteeing efficient and, most importantly, safe performance at the perturbation phase (in the method described in [19], offline simulation experiments are used to predefine acceptable levels of parameter perturbations for a stable traffic behavior).

In a series of papers [2]–[5], we introduced and analyzed a new family of AO algorithms that can overcome the aforementioned limitations (1) and (2). This approach appropriately combines the favorable features of AO algorithms with the features of approximation and adaptive mechanisms to come up with an AO methodology that can rapidly and efficiently optimize systems of arbitrary complexity and scale such as LSTCS and, most importantly, guarantee robust and safe performance while the maintenance operation is on.

III. PROBLEM FORMULATION

Consider a general discrete-time control system where the underlying dynamics are described according to the following nonlinear first-order difference equation:

$$z_{t+1} = F(z_t, u_{i,t}, d_t, t), \quad z_0 = z(0) \quad (1)$$

where z_t , $u_{i,t}$, and d_t are the vectors of system states, control inputs, and exogenous (possibly measurable) signals, respectively, t denotes the discrete-time index, i denotes the controller index, and $F(\cdot)$ is a sufficiently smooth nonlinear vector function. Note that the proposed methodology can be applied to a system, although the function F is unknown. In addition, consider that one or more control laws are applied to the system (1), which are described as follows:

$$u_{i,t} = \varpi_i(\theta_i, z_t) \quad (2)$$

where $\varpi_i(\cdot)$ are known smooth vector functions, and θ_i is the vector of the i th controller tunable parameters. Note that we do not impose any restriction on either the form of (2) or the number of the applied control laws. In addition, the discrete-time index t may be different for each control law i .

The overall system performance is evaluated through the following objective function (performance index):

$$\begin{aligned} J(\theta; z_0, D_T) &= \pi_T(z_T) + \sum_{i=1}^I \sum_{t=0}^{T-1} \pi_{i,t}(z_t, u_{i,t}) \\ &= \pi_T(z_T) + \sum_{i=1}^I \sum_{t=0}^{T-1} \pi_{i,t}(z_t, \varpi(\theta_i, z_t)) \end{aligned} \quad (3)$$

where $\theta = \text{vec}(\theta_1, \theta_2, \dots, \theta_I)$, π_T , and $\pi_{i,t}$ are known nonnegative functions, I is the number of the fine-tuned controllers, T is the finite-time horizon over which the control laws (2) are applied, and $D_T \triangleq [d_0, d_1, \dots, d_{T-1}]$ denotes the time history of the exogenous signals over the optimization horizon T . By defining $x = \text{vec}(z_0, D_T)$, (3) may be rewritten as

$$J(\theta; z_0, D_T) = J(\theta, x). \quad (4)$$

Equation (4) indicates that the system performance is affected by the vector of the tunable parameters θ . The problem at hand is to develop an appropriate iterative algorithm, which will be applied every T and will update the current control system parameters vector θ to achieve better performance but also provide a safe and efficient behavior. This condition means that the algorithm should guarantee a stable sustainable system performance.

In every iteration k of the algorithm (fine-tuning experiment), the following two cases take place.

- The LSTCS performance (1)–(3) is evaluated for $\theta = \theta_k$ through the measurement

$$J_k \equiv J(\theta_k, x_k). \quad (5)$$

- The current controller design parameter vector θ_k is updated so that it converges, as close as possible, to one of the local minima θ^* of the average value of J (with respect to the exogenous random vectors x_k), which is defined according to

$$\frac{\partial}{\partial \theta} E[J(\theta^*, x_k) | \mathcal{G}_k] = 0 \quad (6)$$

where \mathcal{G}_k is an appropriately defined term that refers to the past values of vector θ and the exogenous inputs x .

The requirement of the convergence of θ_k to one of the local minima θ^* is not sufficient in most practical situations. In addition to this requirement, the fine-tuning algorithm should provide *safe* and *efficient* performance during the fine-tuning. More precisely, at each iteration of the fine-tuning algorithm, the performance index measurement should satisfy

$$J_k \leq J_{k-1} + \epsilon_k \quad (7)$$

where ϵ_k is an appropriately defined positive term, whose magnitude is proportional to the magnitude and variance of the exogenous inputs.

The requirement (7) is more than crucial in most practical LSTCS fine-tuning applications, because the violation of such a requirement may cause serious performance and safety problems. For example, in the case of the fine-tuning

of traffic control systems, the violation of requirement (7) may lead to serious problems (e.g., complaints and dangerous driving), which may force the traffic operators to cancel the fine-tuning. Similarly, in the case of the fine-tuning of LSTCSs for mechanical structures, the violation of this requirement may cause the permanent deformation or even the destruction of the structure. Note that standard AO methodologies, e.g., the SPSSA algorithm, cannot guarantee that the requirement (7) holds during the fine-tuning mainly due to the use of random perturbations of the controller parameters.

IV. ADAPTIVE FINE-TUNING ALGORITHM

This section briefly presents the main structure and the performance characteristics of the AFT algorithm.

A. Structure of the AFT Algorithm

The main components of the employed algorithm are summarized as follows.

- An approximator $\hat{J}(\theta, x)$ is used (e.g., a neural network or a polynomial-like approximator) to obtain an approximation of the nonlinear mapping $\hat{J}(\theta, x) = J(\theta, x)$.
- An online adaptive/learning mechanism is employed to “train” the aforementioned approximator. Globally convergent learning algorithms (for example, see [25] and [26]) are required for such a purpose.
- At each algorithm iteration k , several randomly chosen candidate perturbations of vector θ_k are selected, and the effect of each of these perturbations to the LSTCS performance is estimated using the aforementioned approximator. The perturbation that corresponds to the “best” estimate (i.e., the perturbation that leads to the best value for \hat{J}) is picked to be the new tunable parameter values θ_{k+1} , which will be applied at the next period (e.g., the next day).

B. Performance Approximator

As aforementioned, for the approximation of the objective function $J(\theta, x)$, a polynomial-like approximator with L_g regressor terms is used, which takes the form

$$\hat{J}(\theta, x) = \vartheta^\top \phi(\theta, x) \quad (8)$$

where ϑ denotes the matrix of the approximator parameter estimates, and

$$\phi(\theta, x) = [\phi_1(\theta, x), \phi_2(\theta, x), \dots, \phi_{L_g}(\theta, x)]^\top. \quad (9)$$

The nonlinear functions $\phi_i(\theta, x)$ are given by

$$\phi_i(\theta, x) = S^{d_1}(\theta_{m_1}) \cdot \bar{S}^{d_2}(x_{m_2}) \cdot S^{d_3}(\theta_{m_3}), \quad d_i \in \{0, 1\} \quad (10)$$

where d_1, m_i are randomly chosen at each iteration of the AFT algorithm, and $S(\cdot), \bar{S}(\cdot)$ are smooth monotone nonlinear functions. In the neural-networks literature [27], [28], these

TABLE I
VARIABLES USED WITHIN AFT

k	iteration index
ℓ	past performance measurements index
J_ℓ	performance value for the ℓ -th calibration experiment
\hat{J}_ℓ	an estimate of J_ℓ obtained at the ℓ -th iteration
θ_k	the vector of tunable parameters at the k -th calibration experiment
θ_k^*	the “best” set of tunable parameters until the k -th experiment
x_k	the exogenous signals as defined in (III)
\bar{x}_k	an estimate/prediction of the exogenous signals x_k
$\Delta_{k,i}^{(j)}$	zero-mean random sequences (e.g. Gaussian)
$\Delta\theta_k$	the perturbation picked by the algorithm

Remark: The reader is referred to [2] where a procedure is presented for the construction of \bar{x}_k using detector measurements at the network origins.

TABLE II
AFT ALGORITHM DESCRIPTION

- 1) calculate K random perturbations
$$\Delta\theta_{k,i}^{(j)} = \alpha_k \Delta_{k,i}^{(j)} - \theta_{k-1,i} + \theta_{k-1,i}^*, j \in \{1, \dots, K\}$$
- 2) calculate the number of approximator regressor terms
$$L_g^{(k)} = \min \left\{ 2(k-1), \bar{L}_g \right\}$$
- 3) calculate the number of past measurements
$$\ell_k = \max \{k - T_h, 1\}$$
- 4) produce the polynomial-like approximator
$$\phi_\ell^{(k)} = \phi^{(k)}(\theta_\ell, \bar{x}_\ell)$$
- 5) calculate the optimal approximator parameter estimates
$$\vartheta_k \mapsto \arg \min_{\vartheta} \frac{1}{2} \sum_{\ell=\ell_k}^{k-1} \left(J_\ell - \vartheta^\tau \phi_\ell^{(k)} \right)^2$$
- 6) apply the $2K$ random perturbations $\pm \Delta\theta_k^{(j)}$ to the \hat{J}_k

$$\hat{J}_k \left(\pm \Delta\theta_k^{(j)} + \theta_k^*, \bar{x}_k \right) = \vartheta_k^\tau \phi^{(k)} \left(\pm \Delta\theta_k^{(j)} + \theta_k^*, \bar{x}_k \right)$$
- 7) pick the “best” random perturbation (according to the \hat{J}_k)
$$\Delta\theta_k = \arg \max_{\Delta\theta_k^{(\pm j)}} \hat{J}_k \left(\pm \Delta\theta_k^{(j)} + \theta_k^*, \bar{x}_k \right)$$

α_k is a user-defined positive sequence (e.g. constant stepsize $\alpha_k \equiv \alpha \in (0, 1)$)

T_h, L_g, K are user-defined positive integers

$\theta_k^* + \Delta\theta_k$ denotes the vector of tunable parameters picked to be applied at the next experiment $k+1$

functions are usually chosen to be “sigmoidal.” In this application, we choose

$$S(\theta) = \tanh(\lambda_1 \theta + \lambda_2), \quad \bar{S}(x) = \tanh(\lambda_3 x + \lambda_4) \quad (11)$$

where λ_i are nonnegative real numbers that are initially defined by the user. After four or five iterations of the algorithm, the values of λ_i are optimized to minimize $\min \sum_{\ell=1}^{k-1} (J_\ell - \vartheta^\tau \phi_\ell^{(k)})^2$.

C. Algorithm Design and Convergence Properties

Table I presents a description of the design parameters and variables used within the AFT algorithm, whereas Table II presents a mathematical description of the AFT dynamics.

The proposed algorithm assumes that an estimate or prediction \bar{x}_k of the vector x_k is available. In several applications such an assumption is realistic, because the entries of x_k correspond to system states and exogenous inputs that are available for measurement. However, there may be cases where such an assumption is not realistic. In this case, \bar{x}_k can be estimated or predicted using appropriate estimation algorithms.

Note that, similarly to conventional AO algorithms, the proposed algorithm introduces random perturbations to the current control design parameter vector θ . In addition, the use of random perturbations is crucial for the efficiency of the proposed algorithm, because it provides the so-called persistence of excitation (PE) property, which is a sufficient and necessary condition for the neural approximator \hat{J} to efficiently learn the unknown function J . However, due to the use of step 6 (see Table II), the proposed methodology avoids poor performance or instability problems and guarantees safe and efficient performance, because (7) is fulfilled.

Contrary to other applications of neural approximators, where the number of neurons \bar{L}_g should be large enough to guarantee efficient approximation over the whole input set, this is not the case here. In the case of the proposed algorithm, it is sufficient that the approximator has enough regressor terms to come up with an approximation of the unknown function J over a “small neighborhood” around the most recent vector θ_k .

As shown in [2]–[5], using strict mathematical arguments, if the structure of the approximator and its learning mechanism satisfy certain design considerations (that are independent of the particular application), then the aforementioned process guarantees rapid convergence of the overall maintenance procedure to the same performance levels that would have been obtained if efficient nonlinear optimization schemes such as the steepest descent or Gauss–Newton schemes can be applied to the particular problem. Most importantly, the aforementioned procedure guarantees safe, stable, and efficient transient performance, because the system performance during maintenance remains within acceptable levels that can be, in the worst case, similar to the system performance before the maintenance has started.

V. TRAFFIC-RESPONSIVE URBAN CONTROL SIGNAL CONTROL STRATEGY

The TUC signal control strategy (see [6]–[8] for details) is a recently developed efficient TUC strategy whose design principles are based on the feedback control theory as opposed to most of the existing strategies that employ model-based optimization techniques. TUC consists of four distinct interconnected control modules that allow for real-time control of the following traffic measures:

- 1) green times (the split module);
- 2) the cycle time (the cycle module);
- 3) the offset (the green wave along an arterial);
- 4) the provision of public transport priority.

These four control modules are complemented by the fifth data-processing module. All control modules are based on feedback concepts of various types, which leads to TUC’s computational simplicity compared to model-based optimization

approaches, without sacrificing efficiency. In this paper, we will concentrate on the fine-tuning of the design parameters of the following two distinct mutually interacting control modules of TUC: 1) split and 2) cycle. Note that the proposed algorithm can also be applied to the fine-tuning of other control modules.

In the next two sections, we present the control laws that govern the split and cycle modules of TUC.

A. Split Control Module

The split control part of the TUC signal control strategy is derived from a formulation in the format of a linear–quadratic (LQ) control problem that leads to the multivariable regulator

$$g_{t_s} = g^N - Lz_{t_s} \quad (12)$$

where $t_s = 0, 1, 2, \dots$ is the discrete time index, with the sample time period typically equal to the cycle time duration C , g_{t_s} is the control vector (that consists of the green times of all stages in all junctions) that will be applied during the next cycle, g^N is a nominal control vector (that consists of the nominal green times) that corresponds to a prespecified fixed signal plan (the impact of this plan on the resulting control was found to be limited), and z_{t_s} is the state vector (that consists of the vehicle numbers in all network links during the last cycle) that is estimated by the data-processing module of TUC. Finally, L is a constant feedback gain matrix (of appropriate dimensions) that is calculated offline based on a straightforward procedure according to the LQ regulator methodology. The entries of matrix L depend on the network geometry, the turning rates, and the saturation flows. The sensitivity of TUC's performance to moderate variations of these values has been found to be negligible [6], [29]. The aim of (12) is to balance the relative space occupancies $z_i/z_{i,\max}$ in the network links to minimize the risk of queue spillovers, which may lead to a waste of green time and even to gridlocks. To this end, the regulator (12) may apply an inherent gating, i.e., reduce the green time of links that feed a saturating road, although these links are two or more junctions away.

The number of vehicles z_{i,t_s} for link i during the last cycle is estimated through the following equation:

$$z_{i,t_s} = z_{i,\max} f(o_{i,t_s}, l_i) b_i \quad (13)$$

where o_{i,t_s} denotes the measured average time occupancy (usually measured by loop detectors that are located at a certain distance from the stop line) during the last cycle time, $f(\cdot)$ is an empirical function [8], [30] that is constructed from practical investigations, and l_i denotes the distance of the loop detector from the stop line divided by the total link length. Finally, b_i is a nonnegative design parameter for each link i (the so-called “importance factor”), which is introduced such that the z_{i,t_s} -values that result from (13) are multiplied with the corresponding b_i before being used in the multivariable regulator (12). The default values are $b_i = 1$, but experienced system operators may manually select a real value $b_i \in (0, 3]$ to increase or decrease the importance of specific links, i.e., make them look more or less saturated than the measurements actually reflect. These design parameters are critical for the

successful deployment and operation of the TUC signal control strategy and were hence selected for automated fine-tuning by the AFT algorithm.

B. Cycle Control Module

Cycle control is another module that TUC uses to influence traffic conditions. Longer cycle times typically increase the capacity of a junction but, on the other hand, may increase vehicle delays in undersaturated junctions due to longer waiting times during the red phase or, even worse, create queue spillovers. Considering the aforementioned remarks, the objective of the cycle control module is to increase the junctions' capacities as much as necessary to limit the maximum observed saturation level in the network. Within TUC, this objective is effectuated through the application of a simple feedback-based regulator that uses the current saturation level of a prespecified percentage of the network links as a criterion for the increase or decrease of the cycle. The cycle module control law takes the form

$$C_{t_c} = \begin{cases} C^N + K_1(\sigma_{t_c} - \sigma_{N_1}), & \text{if } \sigma_{t_c} \leq \sigma_{cr} \\ C^N - K_2(\sigma_{t_c} - \sigma_{N_2}), & \text{if } \sigma_{t_c} > \sigma_{cr} \end{cases} \quad (14)$$

where $t_c = 0, 1, 2, \dots$ is the discrete-time index of the cycle control, C^N denotes a nominal network cycle time (typically equal to the minimum permissible cycle C_{\min}); σ_{t_c} is a vector that is composed of the mean values of the space occupancies for the prespecified links over the last cycle control period; σ_{N_1} , σ_{N_2} , and $\sigma_{cr} \in [0, 1]$ denote user-defined design parameters; and K_1 and $K_2 > 0$ are networkwide design parameters, the selection of which affects the intensity of the cycle control module reactions and may hence cause the degradation of the overall performance of the TUC strategy if not suitably configured. In other words, high K_1 and K_2 values force the control law to strongly react even for small differences of σ_{t_c} from σ_{N_i} , $i = 1, 2$. For this reason, the design parameters K_1 , K_2 , σ_{N_1} , σ_{N_2} , and σ_{cr} were selected for automated fine-tuning by the AFT algorithm. After the application of (14), the calculated cycle time C_{t_c} is constrained within the range $[C_{\min}, C_{\max}]$, if necessary, to become feasible, where C_{\min} and C_{\max} are the minimum and maximum permissible network cycle times, respectively.

VI. APPLICATION OF THE ADAPTIVE FINE-TUNING ALGORITHM TO THE TRAFFIC-RESPONSIVE URBAN CONTROL SIGNAL CONTROL STRATEGY

To evaluate the efficiency of the aforementioned AFT algorithm to the problem of optimizing the design parameters of the split and cycle control modules of TUC, extensive simulation experiments have been conducted. The performance of AFT is compared to the base case (the non-AFT case), where the aforementioned design parameters were manually fine-tuned to virtual perfection by the system operators for the original TUC system [30].

A. Network and Simulation Setup

For the simulation experiments of the proposed approach, the road network of the city center of Chania, Greece, was

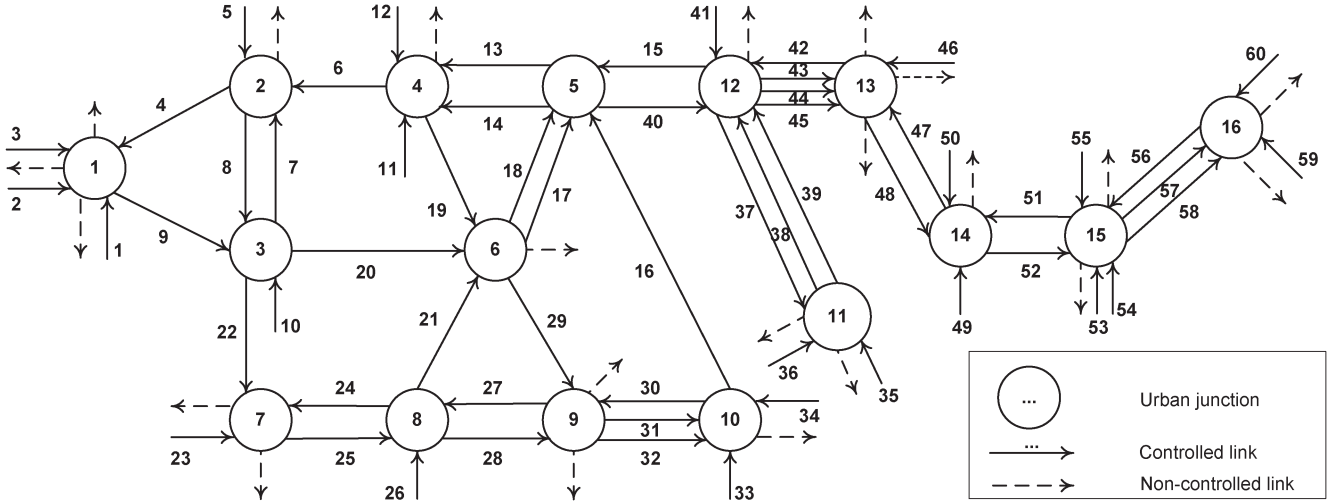


Fig. 2. Chania urban road network.

considered. The model of the network (see Fig. 2) consists of 16 signalized junctions (nodes) and 60 links (arrows). Each network link corresponds to a particular junction phase. Typical loop-detector locations within the Chania urban network links are either around the middle of the link or some 40 m upstream the stop line. Note that severe congestion problems occur in the actual Chania network, which sometimes leads to grid-lock situations.

The commercial microscopic simulator AIMSUN (version 6.0.1) [31] was employed as a simulation tool. The simulation step for the microscopic simulation model of the urban road network of the city center of Chania, Greece, was set to 0.5 s. The traffic network characteristics (e.g., saturation flows and turning rates) and the fixed plan g^N in (12) used in AIMSUN and TUC were suggested by the system operators of the traffic control center (TCC) of the city (details are omitted due to space constraints). Note that the fixed plan g^N is one of the six fixed predefined network signal plans used by the TCC. For the application of the TUC strategy, the following typical design values were used.

- $t_s = C$.
- $t_c = 600$ s.
- $C_{\min} = 60$ s.
- $C_{\max} = 120$ s.
- $C^N = C_{\min}$.

In addition, for the implementation of the AFT algorithm, the following design values were used.

- $T_h = 90$.
- $\bar{L}_g = 150$.
- $K = 20$.
- $\alpha_k = \alpha = 0.1$.

Initial values were set to λ_i according to $\lambda_1 = 100$, $\lambda_3 = 0.1$, and $\lambda_2 = \lambda_4 = 0$. In reference to the notation used in Section III, we have $\theta = \text{vec}(\theta_1, \theta_2)$, where $\theta_1 = (b_1, b_2, \dots, b_{60})$ and $\theta_2 = (K_1, K_2, \sigma_{N_1}, \sigma_{N_2}, \sigma_{cr})$ are the design parameters of the split and cycle control modules of TUC, respectively. The initial values for these parameters were chosen to correspond to values that are usually chosen during the initial

field implementation of the TUC system. More precisely, the parameters θ were initialized according to $\theta_1 = 1$ and $\theta_2 = (240, 300, 0.15, 0.6, 0.4)$. Finally, to assess the overall system performance, the criterion $J \equiv ms$ [see (3)] was set to the actual daily network mean speed.

To investigate the performance of the AFT algorithm under different traffic conditions, two basic traffic demand scenarios (time history of vehicles that enter the network in the network origins during the day) were designed based on actual measurements. The simulation horizon of each scenario is 4 h. Scenario 1 comprises medium demand in all network origins, whereas scenario 2 comprises high demand, and the network faces serious congestion for some 2 h, with some link queues spilling back into upstream links. For simplicity, we assume that a demand scenario with a time horizon of 4 h corresponds to a day. Each day (iteration of the AFT algorithm), a randomly perturbed 5% width version of the basic demand scenarios is produced, and the assessment criterion is gathered from the AIMSUN simulator. Then, the design parameters of TUC strategy are updated by the AFT algorithm according to the calculated assessment criterion.

The overall closed-loop scheme consists of the following two main control loops: 1) the inner loop and 2) the outer loop. The inner loop is used by the TUC strategy to produce the traffic signal settings. More specifically, at each cycle C , AIMSUN delivers the (emulated) occupancy measurements at the locations where detectors are placed (as in real conditions). These measurements are used by the control modules of the TUC strategy to produce the traffic signal settings (split and cycle). These signal settings are then forwarded to the microsimulator for application through the application programming interface (API) programming module of AIMSUN. The outer loop is used by the AFT algorithm to update the design parameters of the TUC strategy. More specifically, at each day, AIMSUN delivers the mean speed for the whole urban road network. The mean speed is used by the AFT algorithm to produce the design parameters of the split and cycle control modules of the TUC strategy (the vector $\theta = \text{vec}(\theta_1, \theta_2)$). These design parameters are then forwarded to the TUC strategy for application, and so forth.

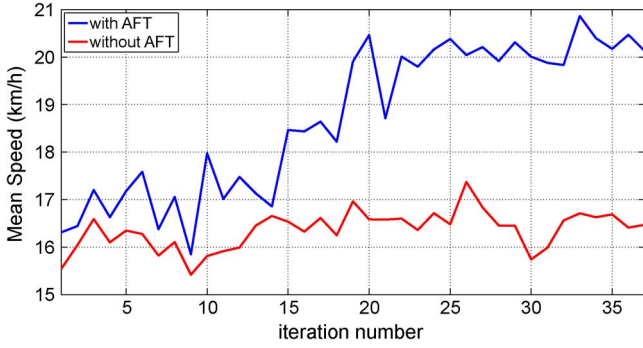


Fig. 3. Mean speed trajectory for demand scenario 1.

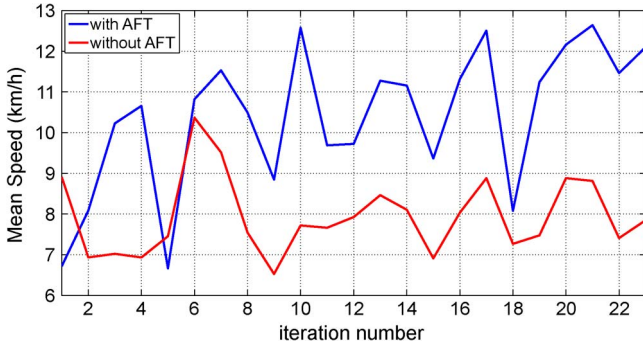


Fig. 4. Mean speed trajectory for demand scenario 2.

B. Global Results

Figs. 3 and 4 compare the networkwide mean speed of the original TUC system (dashed line) versus the TUC system combined with the AFT algorithm (solid line) delivered for scenarios 1 and 2, respectively. It is shown that the application of the AFT algorithm to the TUC signal control strategy leads to better performance than the original TUC for both medium- and high-traffic-demand scenarios. More precisely, the AFT algorithm approaches optimal overall system performance within a few days (iteration number in Figs. 3 and 4) by efficiently fine-tuning the design parameters for both the split and cycle control modules while avoiding decreasing the daily mean speed lower than the initial point. The trajectory of the system performance (mean speed) persistently increases in both scenarios until it converges to a local maximum value. Note that, in scenario 2, the high sensitivity of the system performance to the $\pm 5\%$ daily random perturbations leads to slight improvements for some days/iterations (see Fig. 4); however, the system performance is still improved in terms of average values.

Table III displays the mean speed, the standard deviation of the mean speed due to the $\pm 5\%$ daily random perturbations of the demand scenarios (for the original TUC system), and the mean speed after the convergence of the AFT algorithm, in averages of the number of iterations (days), for each scenario. It is shown that the use of the AFT algorithm leads to an average improvement of the system performance of some 14% for demand scenario 1 and some 31% for the higher demand scenario 2 (see Table III, column 3). Note that the AFT algorithm learns the traffic system dynamics during the first iterations (days) and then converges to a local optimal solution (i.e., after day 20 for

scenario 1 and day 10 for scenario 2; see Figs. 3 and 4). Thus, if the learning period is excluded (see the last two columns in Table III), the improvement increases to 23% and 40% for scenarios 1 and 2, respectively.

Table IV displays two more assessment criteria that are gathered from the microscopic simulator AIMSUN for both simulation scenarios. The average delay time (DT) per kilometer traveled (in seconds per kilometer) and the average number of stops (NS) per vehicle and per kilometer traveled are illustrated. The table also presents the improvement of these criteria due to the use of the AFT algorithm (20.86% and 16.21% for scenario 1 and 37.72% and 21.59% for scenario 2).

Figs. 5 and 6 display the importance factors of the network links according to the optimal solution of the AFT algorithm for scenarios 1 and 2, respectively. The green color indicates low importance ($b_i \leq 0.7$), the black color indicates medium importance ($0.7 < b_i < 1.3$), and the red color indicates high importance ($b_i \geq 1.3$). It is shown that the AFT algorithm increases the weight of the importance factors for network links along the main entrance to and exit from the city center (junctions 16, 15, 14, 13, 12, 5, and 4), whereas for other links, which are not very crucial for the overall system performance, the corresponding weights are decreased. Although there are links with the same color for both scenarios, there are other links that are green in scenario 1 and red in scenario 2, and *vice versa*. The AFT algorithm converges to a local optimal solution that fine-tunes the importance factors and optimizes the traffic control system performance, which also depends on the special characteristics of each demand scenario.

The cycle control parameters are also fine-tuned, and the obtained trajectories are displayed in Figs. 7–10 for both demand scenarios. It is shown that, again, the AFT algorithm converges to an optimal solution (in average values) for $\theta_2 = (K_1, K_2, \sigma_{N_1}, \sigma_{N_2}, \sigma_{cr})$, which depends on the special characteristics of each demand scenario. More specifically, the design parameters K_1, K_2 , which reflect the slope (positive and negative, respectively) of cycle time in (14), converge in average values of 220, 310, 270, and 320 for scenarios 1 and 2, respectively. Finally, the critical saturation level σ_{cr} is around 0.4 for both scenarios, whereas the average fine-tuned values for $\sigma_{N_1}, \sigma_{N_2}$ are lower for scenario 1 (around 0.05 and 0.6, respectively) and higher for scenario 2 (around 0.12 and 0.7, respectively).

The average computational time for every iteration of the AFT algorithm is less than 10 s, which means that the implementation of the algorithm in a real-time large-scale application will be feasible, regardless of the type of the operating traffic control system. Finally, note that the AFT algorithm can also be utilized as an offline network optimization tool for calculating optimum sets of design parameters for LSTCSs of any type, because its system dynamics and controls (1) and (2) and related performance criterion (4) incorporate all the necessary network characteristics.

C. Detailed Results

In this section, we report on some selected results, focusing on the city's main shopping district (see Fig. 2, junction 5).

TABLE III
COMPARISON OF THE AVERAGE MEAN SPEED (IN MILLISECONDS) WITH AND WITHOUT THE APPLICATION OF THE AFT ALGORITHM TO THE TUC STRATEGY

scenario	average ms (km/h)		average ms comparison	ms deviation (km/h)		after the convergence of AFT	
	without AFT	with AFT		without AFT	average ms (km/h)	improvement	
1	16.39	18.72	14.22%	0.39	20.10	22.64%	
2	7.94	10.41	31.11%	0.94	11.09	39.67%	

TABLE IV
COMPARISON OF THE AVERAGE DT AND AVERAGE NS WITH AND WITHOUT THE AFT ALGORITHM

scenario	average DT per km traveled (s/km)		DT improvement	average NS per km traveled		NS improvement
	without AFT	with AFT		without AFT	with AFT	
1	158.34	125.31	20.86%	4.01	3.36	16.21%
2	424.49	264.37	37.72%	6.53	5.12	21.59%

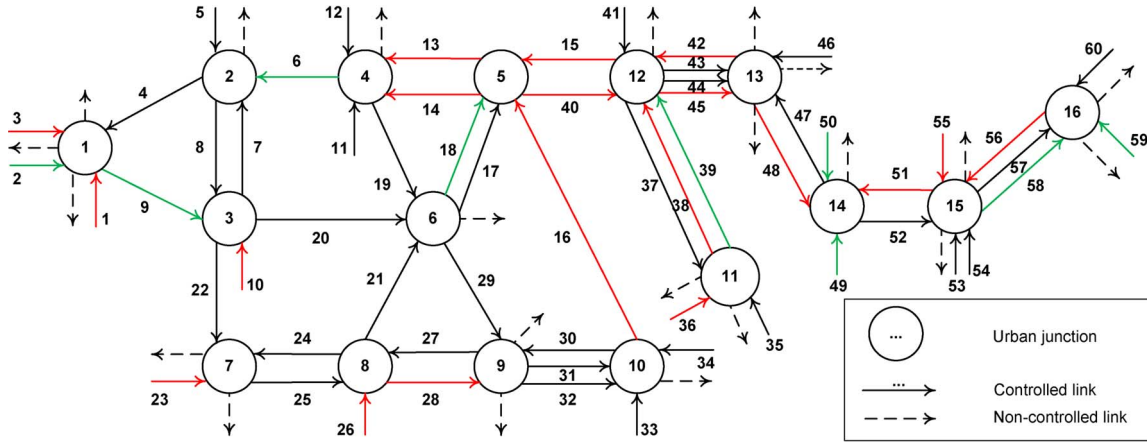


Fig. 5. Network links importance (green for low, black for medium, and red for high) to the split module of TUC according to the AFT algorithm for scenario 1.

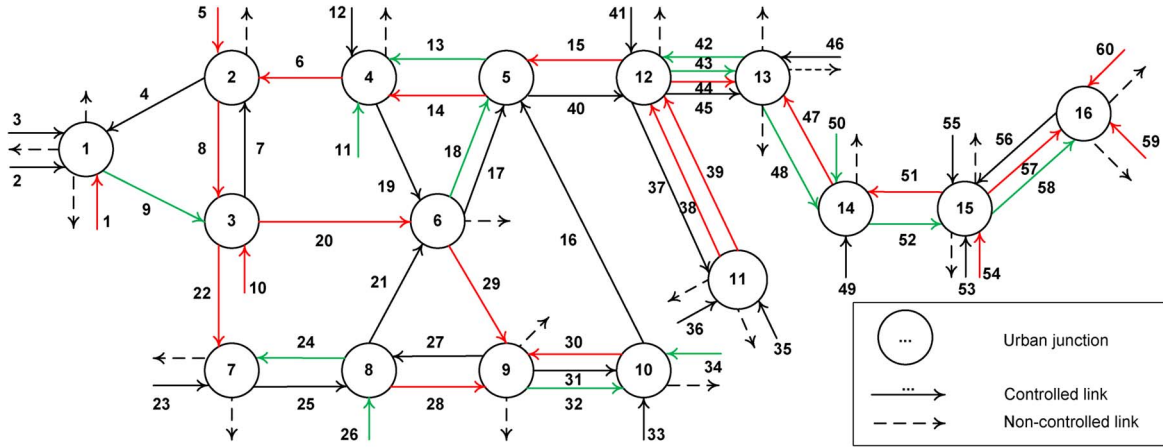


Fig. 6. Network links importance (green for low, black for medium, and red for high) to the split module of TUC according to the AFT algorithm for scenario 2.

With regard to the split control module of the TUC strategy, Fig. 11 compares the time evolution of the design parameters b_{15} and b_{18} under the use of the AFT algorithm, with some optimized values for these parameters. The optimized values come from a manual fine-tuning procedure, previously performed by human experts in a field evaluation of the TUC strategy in the Chania network [30]. The manual tuning of b_{15} and b_{18} led to the optimized weights 1.8 and 0.6, respectively. This link weighting is quite reasonable, because link 15 is a crucial link in the main arterial of the city center, contrary to link 18, which

does not carry substantial traffic loads. The AFT algorithm starts from the initial weights $b_{15} = b_{18} = 1$ and, by iteratively optimizing their values, converges to weights that are shown to be close to the roughly optimized values. Fig. 12 displays the aforementioned design parameters for demand scenario 2. The weights again converge close to the optimized values, although they are slightly different from scenario 1 due to different traffic conditions. What is clear in both figures is that link 15 is more important than link 18 for the network mean speed. Note that this case holds for several other network links that

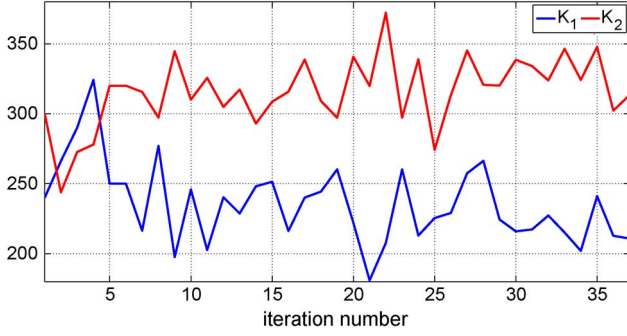


Fig. 7. Trajectories of the cycle parameters K_1 and K_2 (scenario 1).

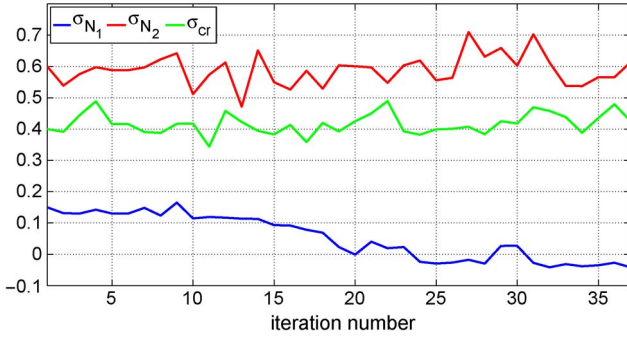


Fig. 8. Trajectories of the cycle parameters σ_{N_1} , σ_{N_2} , and σ_{cr} (scenario 1).

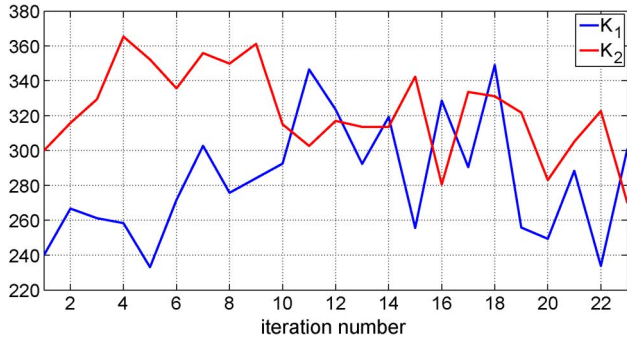


Fig. 9. Trajectories of the cycle parameters K_1 and K_2 (scenario 2).

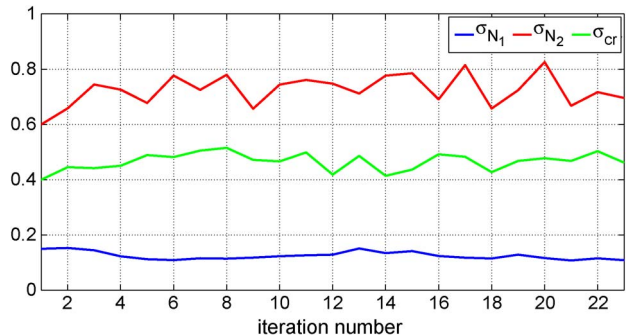


Fig. 10. Trajectories of the cycle parameters σ_{N_1} , σ_{N_2} , and σ_{cr} (scenario 2).

are not shown here (see Figs. 5 and 6 for a general view). This case demonstrates that the proposed algorithm is a feasible and viable solution for the automated parameter fine-tuning of such systems.

In the following discussion, we illustrate the impact of the AFT algorithm to the cycle control module of the TUC strategy.

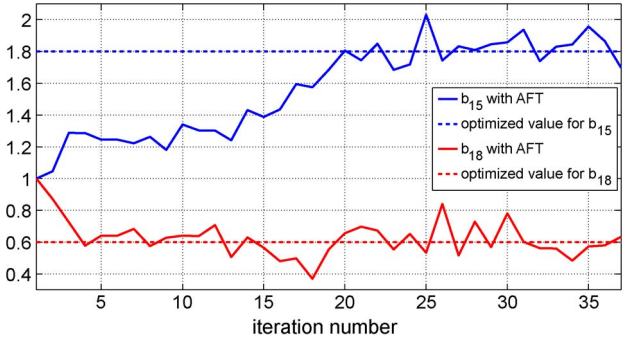


Fig. 11. Trajectories of the split parameters b_{15} and b_{18} and the corresponding optimized values (scenario 1).

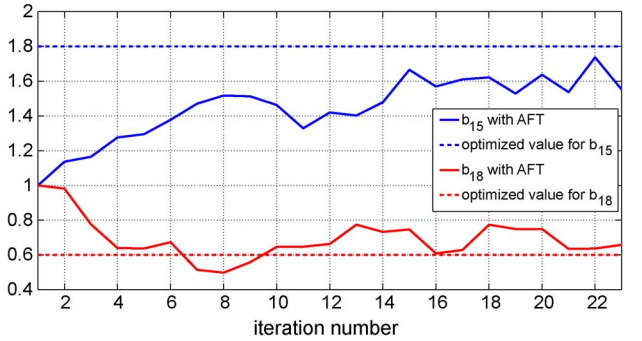


Fig. 12. Trajectories of the split parameters b_{15} and b_{18} and the corresponding optimized values (scenario 2).

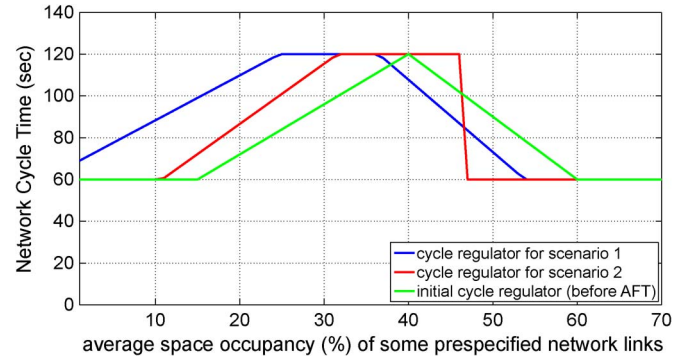


Fig. 13. Trajectories of the initial cycle regulator and the cycle regulators after the convergence of AFT.

The AFT algorithm, by changing the design parameters $\theta_2 = (K_1, K_2, \sigma_{N_1}, \sigma_{N_2}, \sigma_{cr})$, changes the cycle control feedback regulator (14) and succeeds in notably improving the mean speed of the simulated network. Fig. 13 displays different values of σ_{t_c} , the initial cycle control regulator, and the fine-tuned cycle control regulator after the convergence of AFT algorithm for scenarios 1 and 2, respectively. Recall that $\sigma_{t_c} \in [0, 1]$ is the average space occupancy for some prespecified percentage of network links over the last cycle control period. Here, we depict the three different regulators for σ_{t_c} from 0 to 0.7.

The initial cycle control regulator [see (14)] consists of one monotonically increasing function for undersaturated traffic conditions ($\sigma_{cr} < 0.4$) and one monotonically decreasing function for saturated traffic conditions ($\sigma_{cr} > 0.4$). The maximum cycle $C_{max} = 120$ s is applied for the critical occupancy $\sigma_{cr} = 0.4$. After the convergence of the AFT algorithm, a

cycle regulator with the following three regimes is obtained for both simulated scenarios: 1) one regime with increasing cycle periods for undersaturated traffic conditions; 2) one regime with maximum cycle periods $C_{\max} = 120$ s (for $0.25 \leq \sigma_{cr} \leq 0.4$ and $0.32 \leq \sigma_{cr} \leq 0.4$ for scenarios 1 and 2, respectively) for traffic conditions that require us to increase the capacity of the network; and 3) one regime with decreasing cycle periods for saturated traffic conditions (see Fig. 13). For the third regime, the network faces severe congestion problems due to queue spillovers and partial gridlocks that lead to a strong performance deterioration. Again, there are slight differences between the fine-tuned cycle control regulators of scenarios 1 and 2, which depend on the different traffic characteristics.

In general, the derived trapezoidal shape (see Fig. 13) of the cycle control regulators over the saturation level of the network outperforms the initial cycle control regulator, as shown by the overall system performance. Eventually, the following three traffic regimes may be identified: 1) undersaturated traffic conditions; 2) critical traffic conditions; and 3) saturated traffic conditions. This way, the cycle control module of TUC applies appropriate cycle times for each case, i.e., smaller cycle times for regimes 1 and 3 and the maximum cycle time for regime 2, to maximize the network's capacity. Note that regime 2 occurs for different values of σ_{cr} for demand scenarios 1 and 2, which means that the real-time implementation of the AFT algorithm can be vital for the overall system performance.

VII. CONCLUSION

This paper has investigated the efficiency of the AFT algorithm for the problem of optimizing the design parameters of LSTCSs that are composed of distinct mutually interacting modules. This AO methodology aims at replacing the conventional manually based optimization with a fully automated procedure. Extensive simulation experiments have been conducted for the signal control problem of the large-scale traffic network of the city of Chania, Greece, where the design parameters of two distinct mutually interacting modules of the TUC strategy were fine-tuned by the AFT algorithm. The simulation results and the comparison to the base case, where the aforementioned design parameters of the TUC system were manually fine-tuned to virtual perfection by the system operators, demonstrate the algorithm's efficiency and feasibility.

The design parameters of the split and cycle control modules of TUC have been considered for fine-tuning. It was demonstrated that the application of the AFT algorithm to the TUC signal control strategy leads to better network performance (in terms of the daily mean speed) compared with the original TUC system. This case underlines the superiority of the fully automated optimization procedure, pursued by the AFT algorithm, even in the case that the design parameters are already manually fine-tuned by field experts.

With regard to the design parameters of the split control module, it was shown that the AFT algorithm increases the weight of the importance factors for network links along the main entrance to and exit from the city center, whereas for other links, which are not very crucial, the corresponding weights are decreased. Furthermore, it was shown that the AFT algorithm

converges to a set of quite-reasonable design parameters that are close to the roughly optimized values provided by the system operators for the original TUC system. Finally, with regard to the design parameters of the cycle control module, it was shown that the AFT algorithm leads to cycle control regulators with a trapezoidal shape of three traffic regimes, which outperform the initial cycle control regulator.

Future work will deal with the application of the AFT algorithm to all control modules of the TUC strategy (i.e., split, cycle, and offset) and with investigations with regard to the following two approaches: 1) the online generation of the gain sequences α_k and 2) the approximation of the objective function $J(\theta, x)$ through support vector machine (SVM) regressors, which can possibly increase the efficiency of the algorithm.

REFERENCES

- [1] F. Middelham, H. Taale, J. Dibbitts, and W. Fransen, "The assessment of the SCOOT system in Nijmegen," in *Proc. IEEE Conf. Pub.*, 1996, vol. 1996, no. 422, pp. 66–70.
- [2] E. B. Kosmatopoulos, M. Papageorgiou, A. Vakouli, and A. Kouvelas, "Adaptive fine-tuning of nonlinear control systems with application to the urban traffic control strategy TUC," *IEEE Trans. Control Syst. Technol.*, vol. 15, no. 6, pp. 991–1002, Nov. 2007.
- [3] E. B. Kosmatopoulos, "An adaptive optimization scheme with satisfactory transient performance," *Automatica*, vol. 45, no. 3, pp. 716–723, Mar. 2009.
- [4] E. B. Kosmatopoulos and A. Kouvelas, "Large-scale nonlinear control system fine-tuning through learning," *IEEE Trans. Neural Netw.*, vol. 20, no. 6, pp. 1009–1023, Jun. 2009.
- [5] E. B. Kosmatopoulos, "Adaptive control design based on adaptive optimization principles," *IEEE Trans. Autom. Control*, vol. 53, no. 11, pp. 2680–2685, Dec. 2008.
- [6] C. Diakaki, "Integrated control of traffic flow in corridor road networks," Ph.D. dissertation, Tech. Univ. Crete, Chania, Greece, 1999.
- [7] C. Diakaki, M. Papageorgiou, and K. Aboudolas, "A multivariable regulator approach to traffic-responsive networkwide signal control," *Control Eng. Pract.*, vol. 10, no. 2, pp. 183–195, Feb. 2002.
- [8] C. Diakaki, V. Dinopoulou, K. Aboudolas, M. Papageorgiou, E. Ben-Shabat, E. Seider, and A. Leibov, "Extensions and new applications of the traffic-responsive urban control strategy: Coordinated signal control for urban networks," *Transp. Res. Rec.*, no. 1856, pp. 202–211, 2003.
- [9] K. McCabe and A. Riley, "A flexible approach to motorway control," *Atkins Tech. J.* [Online], vol. 3, pp. 37–42. Available: <http://www.atkinsglobal.com/sectors-and-services/sectors/roads/technical-journals/technical-journal-3>
- [10] P. Ioannou, Ed., *Intelligent Freight Transportation*. Boca Raton, FL: CRC, 2008.
- [11] L. Li, N. Tang, X. Mu, and F. Shi, "Implementation of traffic lights control based on Petri nets," in *Proc. Intell. Transp. Syst.*, 2003, vol. 2, pp. 1087–1090.
- [12] J. Sanchez, M. Galan, and E. Rubio, "Applying a traffic lights evolutionary optimization technique to a real case: Las Ramblas area in Santa Cruz de Tenerife," *IEEE Trans. Evol. Comput.*, vol. 12, no. 1, pp. 25–40, Feb. 2008.
- [13] H. Julia, M. Dessouky, P. Ioannou, and A. Chassiakos, "Container movement by trucks in metropolitan networks: Modeling and optimization," *Transp. Res.*, vol. 41E, no. 3, pp. 235–259, May 2005.
- [14] D. Giralda, M. Rodriguez, F. Pernas, J. Higuera, D. Ortega, and M. Zarzuela, "Intelligent system for dynamic transport fleet management," in *Proc. 10th IEEE Conf. Emerging Technol. Factory Autom.*, 2005, vol. 1, pp. 773–776.
- [15] P.-C. B. Liu, M. Hansen, and A. Mukherjee, "Scenario-based air traffic flow management: From theory to practice," *Transp. Res.*, vol. 42B, no. 7/8, pp. 685–702, Aug. 2008.
- [16] J. Spall, "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation," *IEEE Trans. Autom. Control*, vol. 37, no. 3, pp. 332–341, Mar. 1992.
- [17] SPSSA Algorithm. [Online]. Available: <http://www.jhuapl.edu/SPSSA/>
- [18] D. Chin and R. Smith, "A traffic simulation for mid-Manhattan with model-free adaptive signal control," in *Proc. Summer Comput. Simul. Conf.*, 1994, pp. 296–301.

- [19] J. Spall and D. Chin, "Traffic-responsive signal timing for systemwide traffic control," *Transp. Res.*, vol. 5C, no. 3/4, pp. 153–163, Aug.–Oct. 1997.
- [20] M. C. Choy, D. Srinivasan, and R. Cheu, "Neural networks for continuous online learning and control," *IEEE Trans. Neural Netw.*, vol. 17, no. 6, pp. 1511–1531, Nov. 2006.
- [21] N. Kleinman, S. Hill, and V. Ilenda, "SPSA/SIMMOD optimization of air traffic delay cost," in *Proc. Amer. Control Conf.*, 1997, vol. 2, pp. 1121–1125.
- [22] R. Burnett, "Application of stochastic optimization to collision avoidance," in *Proc. Amer. Control Conf.*, 2004, vol. 3, pp. 2789–2794.
- [23] S. Hill and M. Fu, "Transfer optimization via simultaneous perturbation stochastic approximation," in *Proc. 27th Winter Simul. Conf.*, 1995, pp. 242–249.
- [24] M. Koch, D. Chin, and R. Smith, "Networkwide approach to optimal signal light timing for integrated transit vehicle and traffic operations," in *Proc. 7th Nat. Conf. Light Rail Transit*, 1997, vol. 2, pp. 126–131.
- [25] E. Kosmatopoulos, M. Polycarpou, M. Christodoulou, and P. Ioannou, "High-order neural network structures for identification of dynamical systems," *IEEE Trans. Neural Netw.*, vol. 6, no. 2, pp. 422–431, Mar. 1995.
- [26] E. Kosmatopoulos, M. Christodoulou, and P. Ioannou, "Dynamical neural networks that ensure exponential identification error convergence," *Neural Netw.*, vol. 10, no. 2, pp. 299–314, Mar. 1997.
- [27] V. Maiorov and R. Meir, "Approximation bounds for smooth functions in $C(\mathbb{R}^d)$ by neural and mixture networks," *IEEE Trans. Neural Netw.*, vol. 9, no. 5, pp. 969–978, Sep. 1998.
- [28] G.-B. Huang, L. Chen, and C.-K. Siew, "Universal approximation using incremental constructive feedforward networks with random hidden nodes," *IEEE Trans. Neural Netw.*, vol. 17, no. 4, pp. 879–892, Jul. 2006.
- [29] K. Aboudolas, M. Papageorgiou, A. Kouvelas, and E. Kosmatopoulos, "A rolling-horizon quadratic-programming approach to the signal control problem in large-scale congested urban road networks," *Transp. Res.*, vol. 18C, no. 5, pp. 680–694, Oct. 2010.
- [30] E. Kosmatopoulos, M. Papageorgiou, C. Bielefeldt, V. Dinopoulou, R. Morris, J. Mueck, A. Richards, and F. Weichenmeier, "International comparative field evaluation of a traffic-responsive signal control strategy in three cities," *Transp. Res.*, vol. 40A, no. 5, pp. 399–413, Jun. 2006.
- [31] *AIMSUN Users Manual 6*, Transp. Simul. Syst. (TSS), Barcelona, Spain, 2008.

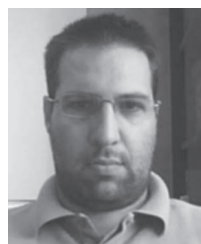


Anastasios Kouvelas was born in Athens, Greece, in 1981. He received the Diploma, M.Eng., and Ph.D. degrees from the Technical University of Crete, Chania, Greece, in 2004, 2006, and 2011, respectively.

From 2003 to 2011, he was a Research and Teaching Assistant with the Dynamic Systems and Simulation Laboratory, Technical University of Crete. In 2009, he was a Short-Term Visiting Scholar with the Center for Advanced Transportation Technologies, Department of Electrical Engineering, University of

Southern California, Los Angeles. He is currently an Adjunct Lecturer with the Department of Production and Management Engineering, Technical University of Crete, and a Research Associate with the Informatics and Telematics Institute, Centre for Research and Technology Hellas, Thessaloniki, Greece. He is a coauthor of 15 journal articles and conference proceedings, and he has participated in various research projects funded by the European Union. His research interests include the modeling, simulation, and control of urban and motorway traffic networks, learning techniques, neural networks, stochastic approximation, adaptive optimization, and intelligent transportation systems.

Dr. Kouvelas is a Member of the Technical Chamber of Greece.



Konstantinos Aboudolas received the Dipl.Eng. degree in production engineering and management and the M.Sc. and Ph.D. degrees in operations research from the Technical University of Crete, Chania, Greece, in 1999, 2003, and 2009, respectively.

From 1999 to 2009, he was a Research and Teaching Assistant with the Technical University of Crete. In 2010, he was an Adjunct Lecturer with the Democritus University of Thrace, Xanthi, Greece. Since 2010, he has been a Research Associate with the Centre for Research and Technology Hellas, Thessaloniki, Greece. He is currently a Visiting Scholar with the University

of California, Berkeley. He is a coauthor of 25 journal articles and conference proceedings and has participated in various European and National research projects. His research interests include traffic flow modeling, simulation, and control for traffic networks; distributed estimation and control for networks of mobile sensors; and large-scale self-tunable and reconfigurable control systems design.



Elias B. Kosmatopoulos received the Diploma, M.Sc., and Ph.D. degrees from the Technical University of Crete (TUC), Chania, Greece, in 1990, 1992, and 1995, respectively.

He was a Faculty Member with the Department of Production Engineering and Management, TUC, a Research Assistant Professor with the Department of Electrical Engineering, University of Southern California, Los Angeles, and a Postdoctoral Fellow with the Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC, Canada.

He is currently an Associate Professor with the Department of Electrical and Computer Engineering, Democritus University of Thrace, Xanthi, Greece. He is currently involved in several research projects funded by the European Union. Among his theoretical contributions, the most important are the analysis of approximation, stability, and learning capabilities of recurrent high-order neural networks and the development and analysis of a switching adaptive controller for unknown dynamical systems. He is a coauthor of more than 35 journal papers. His research interests include neural networks, adaptive optimization and control, and intelligent transportation systems.



Markos Papageorgiou (F'99) was born in Thessaloniki, Greece, in 1953. He received the Dipl.Eng. and Dr. Ing. (with honors) degrees in electrical engineering from the Technical University of Munich, Munich, Germany, in 1976 and 1981, respectively.

From 1976 to 1982, he was a Research and Teaching Assistant with the Department of Control Engineering, Technical University of Munich. He was a Free Associate with Dorsch Consult, Munich, from 1982 to 1988 and with the Institute National de Recherche sur les Transports et leur Sécurité (INRETS), Arcueil, France, from 1986 to 1988. From 1988 to 1994, he was a Professor of automation with the Technical University of Munich. Since 1994, he has been a Professor with the Department of Production and Management Engineering, Technical University of Crete, Chania, Greece. He was a Visiting Professor with the Politecnico di Milano, Milan, Italy, in 1982; the Ecole Nationale des Ponts et Chaussées, Paris, France, from 1985 to 1987; and the Massachusetts Institute of Technology, Cambridge, in 1997 and 2000. He was a Visiting Scholar with the University of Minnesota, Minneapolis, in 1991 and 1993; the University of Southern California, Los Angeles, in 1993; and the University of California, Berkeley, in 1993, 1997, and 2001. He is the author of the books *Applications of Automatic Control Concepts to Traffic Flow Modeling and Control* (Springer, 1983) and *Optimierung* (Oldenbourg, 1991; 1996), the Editor of the *Concise Encyclopedia of Traffic and Transportation Systems* (Pergamon, 1991), and a coauthor of *Optimal Real-Time Control of Sewer Networks* (Springer, 2005). He is also the author or a coauthor of more than 350 technical papers. He is the Editor-in-Chief of *Transportation Research Part C* and was an Associate Editor for several journals. His research interests include automatic control and optimization theory and their applications to traffic and transportation systems, water systems, and further areas.

Dr. Papageorgiou was an Associate Editor for the IEEE Control Systems Society Conference Editorial Board and the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS. He served the International Federation of Automatic Control Technical Committee on Transportation Systems as the Chair from 1999 to 2005 and the Vice-Chair from 1994 to 1999. He received the German Academic Exchange Service Scholarship from 1971 to 1976, the 1983 Eugen-Hartmann Award from the Union of German Engineers, and a 1997 Fulbright Lecturing/Research Award. He received the Outstanding Research Award from the IEEE Intelligent Transportation Systems Society in 2007 and the Transition to Practice Award from the IEEE Control Systems Society in 2010.