


Adaptive performance optimization for large-scale traffic control systems

Conference Paper**Author(s):**

Kouvelas, Anastasios ; Kosmatopoulos, Elias B.; Papageorgiou, Markos; Aboudolas, Konstantinos

Publication date:

2009

Permanent link:

<https://doi.org/10.3929/ethz-b-000276687>

Rights / license:

In Copyright - Non-Commercial Use Permitted

Originally published in:

IFAC Proceedings Volumes 42(15), <https://doi.org/10.3182/20090902-3-us-2007.0037>

Adaptive Performance Optimization for Large-Scale Traffic Control Systems

A. Kouvelas, E. Kosmatopoulos, M. Papageorgiou, K. Aboudolas

*Dynamic Systems and Simulation Laboratory,
Technical University of Crete,
GR-73100 Chania, Greece (Tel.: +30-28210-37306;
e-mail: {tasos,kosmatop,markos,aboud}@dssl.tuc.gr).*

Abstract: The majority of practical Large-Scale Traffic Control Systems (LSTCSs) requires the optimization (fine-tuning) of their design parameters. A tremendous amount of human effort and time is spent for optimization of the overall LSTCS, which is usually performed by experienced personnel in the lack of an automated – well established – systematic approach. This paper, investigates the efficiency of the Adaptive Fine-Tuning algorithm, when applied for automated fine-tuning of an urban traffic LSTCS with mutually-interacting control modules, each one with its distinct design parameters. The approach of AFT is based on a recently proposed Adaptive Optimization (AO) methodology that is aiming at replacing the manually-based optimization by a fully-automated procedure and is proven – using rigorous mathematical arguments – to provide with safe and reliable, efficient and rapid optimization of general LSTCSs. Simulations results demonstrate the efficiency of the proposed approach when applied to the simultaneous fine-tuning of two mutually-interacting LSTCS control modules.

Keywords: Modeling, Control and Optimization of Transportation Systems; Intelligent Transportation Systems.

1. INTRODUCTION

Currently, a tremendous amount of human effort and time is spent for 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, 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 in order to re-adjust and re-program the LSTCS decision making mechanisms. Moreover, the continuous medium- and long-term variations of the overall transport system dynamics (due to e.g. changes of traffic demand or number of passengers using the particular transport system) call for a frequent or even continuous maintenance of LSTCSs, which – if done properly – makes the maintenance of LSTCSs extremely costly. In many cases, the result is that system maintenance is neglected and the system performance deteriorates year after year.

Moreover, the same processes that are required for calibration must also be used in the initial fine-tuning of the system during its first installation. Both tasks (initial fine-tuning and calibration) are performed (if at all) by experienced personnel in the lack of an automated and systematic approach; thus there is no guarantee that the overall fine-tuning and/or maintenance procedure will end-up successfully. In some cases, the LSTCS has never achieved a satisfactory performance in the first place, as for instance in the reported case of the urban signal man-

agement strategy SCOOT (the most popular urban signal management strategy worldwide) in the city of Nijmegen in The Netherlands [5], where the SCOOT application was abandoned completely in the end.

Urban and motorway traffic control systems, LSTCSs for public transport systems and LSTCSs for large-scale 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 procedure involves the re-calibration, re-adjustment and re-programming of hundreds of parameters, rules, operational schedules, decision-making mechanisms, etc., which influence the transport system operations in a highly complex manner. Moreover, the use of heuristic, trial-and-error, experience-based techniques, while the system is in operation, involves the risk of poor system performance over a lengthy period of time, which may lead to poor-quality-service problems, delays, severe congestion and increased Green House Gas (GHG) emissions during this period. It is finally worth noting 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 lack of expertise, exhaustive working conditions, etc., may lead to decisions/actions that put safety at stake.

Recently, we have introduced and analyzed a new family of algorithms – called hereafter Adaptive Fine-Tuning (AFT) – that can be used towards the development of a generic, efficient and systematic approach for the automated fine-tuning of LSTCS [1]–[4]. The main attributes of these algorithms may be summarized as follows:

- They 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 regardless of the methodology used for the original design of the LSTCS.
- They are robust with respect to exogenous disturbances, noisy measurements, system interactions, component failures, etc.
- 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 which render them adaptable to short-term and long-term variations of system characteristics such as demand long-term variations, system aging etc. Moreover, through these learning and estimation mechanisms, they are capable of incorporating the knowledge captured in the past regarding 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 process.

This paper investigates the efficiency and real-time feasibility of one of the AFT algorithms [3], when applied to the problem of calibrating the parameters of an urban traffic LSTCS via simulation experiments. The aforementioned design parameters derive from two distinct control modules operating concurrently within the traffic-responsive urban control strategy TUC [24], [6].

2. LSTCS OPTIMIZATION (FINE-TUNING): BACKGROUND

2.1 Theoretical/Simulation-based methods

The last decades, attempts have been made in particular LSTCS applications to develop model-based, i.e., either theoretical-based or simulation-based designs that produce “good” sets of tunable parameters, and although they have helped in some cases to reduce time and effort for installation and maintenance, they did not manage to eliminate, or at least reduce significantly, the involvement of the human factor. One example in this class is the implementation of a Variable Speed Limit system on the UK motorway M42 [25]. Despite the fact that the initial tunable parameters of the system (which correspond to speed and flow activation/deactivation thresholds) were “optimized” using theoretical tools from traffic flow theory and extensive simulation experiments, it took more than a year of calibration of the aforementioned thresholds until the system reached an acceptable performance and, during this initial deployment phase, the system performance was sometimes worse than in the no-control case.

There have also been some attempts to incorporate optimization-based tools within the maintenance procedure, see e.g. [7]–[12] and the references therein for an indicative list of references. In these cases, the problem of

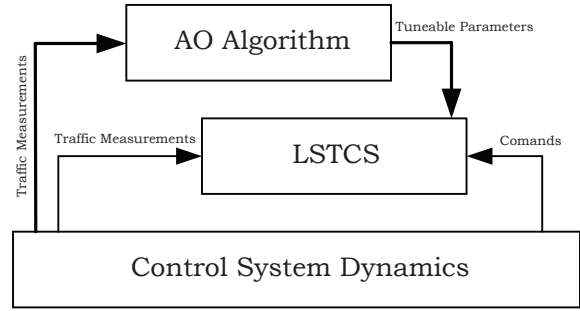


Fig. 1. AO for automatic calibration of LSTCSs.

providing efficient maintenance is formulated as an optimization problem, where the tunable LSTCS parameters are chosen so as to optimize a performance criterion (e.g. average network speeds in traffic networks, average delays in airborne or seaborne transport systems, total number of containers loaded/unloaded in seaport container terminals, average deviation from the operational schedules in public transport systems). However, optimization of such a performance criterion requires perfect or, at least, very accurate knowledge of the transport system dynamics as well as the demand. To deal with this problem, optimization-based approaches employ simulation-based or theoretical models for representing the actual system dynamics. Then, based on the assumption that these models represent quite accurately the actual LSTCS operations, different optimization-based algorithms (e.g. gradient-descent, Gauss-Newton, evolutionary programming or neural network-based optimization algorithms) are applied to the simulation/theoretical-based models in order to extract the optimal values of the tunable parameters. However, these approaches (a) require extensive and continuous calibration of the simulation/theoretical-based models, so as to optimize their approximation accuracy with respect to the actual transport system operations, and (b) face the tradeoff between simplicity and accuracy; in most cases accuracy has to be sacrificed in order to avoid the use of extreme computational requirements of simulation or mathematical models that employ detailed modeling of the LSTCS operations.

2.2 Adaptive and Neural/Learning methods

One possible way to by-pass the above-mentioned problems is to incorporate adaptive or adaptive-like designs (such as neural, fuzzy, iterative learning, etc., methods) for updating the parameters of the LSTCS, which render many advantages contrary to the simulation/theoretical-based techniques. AFT belongs to the family of the so-called AO methods, such as the Simultaneous Perturbation Stochastic Approximation (SPSA) [13]–[15]. These methods provide probably 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 (Fig. 1):

- At the end of appropriately defined periods (e.g. at the end of each day), the AO algorithm receives the value of the real (measured) performance index (e.g. average speed over space and time for traffic networks, total number of containers loaded/unloaded for seaport

container terminals) as well as 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 many periods (e.g. days) until a maximum in performance is reached; then, the AO algorithm may remain active for continuous adaptation or be switched off and re-activated at a later stage (e.g. after few months).

A key idea behind most AO methodologies is to use two different (but inter-woven) 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, so as to increase performance, based on an estimate of the gradient of the LSTCS performance. The gradient is calculated using the values of the performance index (and of the external factors) at the perturbation phase.

The random choice of the perturbations introduced in the perturbation phase is in order for the AO mechanisms to sufficiently explore the overall LSTCS decision space (so as to be able to come up with suitable decision each time). As it was shown in several research articles evaluating AO methods, the introduction of random perturbations is necessary and crucial for the successful operation of the overall scheme ([1], [2], [13], [15]). Different researchers reported very encouraging results - by using simulation experiments - on the application of the aforementioned AO methods in maintenance and renovation of various LSTCS: Urban signal traffic control [16]–[18], air traffic management [19], vessel traffic management [20] and fleet and transit management [21], [22] are few of the LSTCS maintenance applications where these methods have been applied and evaluated using simulation-based experiments.

Unfortunately, these designs suffer from two severe drawbacks:

- (1) The majority of AO methods do not have any mechanism to incorporate the knowledge captured in the past, regarding the dependence of the LSTCS performance on the tunable parameters and the external factors (demand); in case where such a dependence is highly non-linear and complex, the aforementioned algorithms fail to produce any improvement of the overall LSTCS performance.
- (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, unstable or catastrophic behaviour. Hence, AO methods possess the disadvantage of not guaranteeing efficient and, most

importantly, safe performance at the perturbation phase.

In a series of papers [1]–[4], we introduced and analyzed a new family of AO algorithms which are capable of overcoming the limitations (1)–(2) above. This approach appropriately combines the nice features of AO algorithms with those of approximation and adaptive mechanisms in order to come up with an adaptive optimization methodology capable of rapidly and efficiently optimizing systems of arbitrary complexity and scale such as LSTCS and, most importantly, guaranteeing robust and safe performance while the maintenance operation is on.

3. 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 $z_t, u_{i,t}, d_t$ 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 even if the function F is unknown. Consider also, 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 neither on the form of the equation (2), nor on the number of the applied control laws. Also, 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 $\theta = \text{vec}(\theta_1, \theta_2, \dots, \theta_I)$, $\pi_T, \pi_{i,t}$ are known nonnegative functions, I is the number of the fine-tuned controllers, T 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 $x = \text{vec}(z_0, D_T)$, equation (3) may be rewritten as

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

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

4. THE AFT ALGORITHM

The main components of the proposed algorithm are summarized as follows:

- An approximator $\hat{J}(\theta, x)$ is used (e.g., a neural network or a polynomial-like approximator) in order to obtain an approximation of the non-linear mapping $\hat{J}(\theta, x) = J(\theta, x)$.
- An on-line adaptive/learning mechanism is employed for “training” the above approximator. Globally convergent learning algorithms (see e.g., [30], [31]) are required for such a purpose.
- At each algorithm iteration k , many randomly chosen candidate perturbations of vector θ_k are selected and the effect of each of these perturbations to the LSTCS performance is estimated by using the approximator mentioned above; the perturbation that corresponds to the “best” estimate (i.e., the one that leads to the best value for \hat{J}) is picked to be the new tunable parameter values θ_{k+1} , to be applied at the next period (e.g. the next day).

4.1 The formation of the performance approximator

As already mentioned, 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

$$\hat{J}(\theta, x) = \vartheta^\tau \phi(\theta, x) \quad (5)$$

where ϑ denotes the matrix of the approximator parameter estimates and

$$\phi(\theta, x) = [\phi_1(\theta, x), \phi_2(\theta, x), \dots, \phi_{L_g}(\theta, x)]^\tau. \quad (6)$$

The non-linear functions $\phi_i(\theta, x)$ are given by

$$\phi_i(\theta, x) = S^{d_1}(\theta_{m_1}) \cdot \bar{S}^{d_2}(x_{m_2}) \cdot S^{d_3}(\theta_{m_3}), \quad d_i \in \{0, 1\} \quad (7)$$

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 [26], [27] these functions are usually chosen to be “sigmoidal” functions. In this application we choose

$$S(\theta) = \tanh(\lambda_1 \theta + \lambda_2), \quad \bar{S}(x) = \tanh(\lambda_3 x + \lambda_4) \quad (8)$$

where λ_i are nonnegative real numbers initially defined by the user; after 4–5 iterations of the algorithm the values of λ_i are optimized so as to minimize $\min \sum_{\ell=1}^{k-1} (J_\ell - \vartheta^\tau \phi_\ell^{(k)})^2$.

4.2 The structure of the AFT algorithm

This section presents the details and the performance characteristics of the AFT algorithm. Table 1 presents a brief description of the design parameters and variables used within the AFT algorithm, while Table 2 presents a brief mathematical description of the AFT dynamics. Since the purpose of this paper is not to present the AFT algorithm in each and every detail, we present only a brief description of the algorithm while focusing on application results; for more details on AFT the reader is referred to [1]–[4].

As shown in [1]–[4] 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 above described process guarantees rapid convergence of the overall maintenance procedure to the same performance levels that would have been obtained if efficient non-linear optimization schemes such as the steepest descent or Gauss-Newton schemes could be applied to the particular problem. Most importantly, the above-mentioned procedure guarantees safe, stable and efficient transient performance in the sense that the system performance during maintenance remains within acceptable levels that can be, in the worst case, similar to the system performance before maintenance started.

Table 1. Variables used within the 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 (3)
\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 [1] where a procedure is presented for the construction of \bar{x}_k (defined in Table 1) using detector measurements at the network origins.

Table 2. 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	$L_g^{(k)} = \min \{2(k-1), \bar{L}_g\}$
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	$\vartheta_k \mapsto \arg \min_{\vartheta} \frac{1}{2} \sum_{\ell=\ell_k}^{k-1} (J_\ell - \vartheta^\tau \phi_\ell^{(k)})^2$
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	
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 $k + 1$	

5. APPLICATION TO THE LSTCS TUC

5.1 Brief Introduction to TUC

TUC [24], [6] is a recently developed, efficient real-time urban traffic control strategy, whose design principles are based on feedback control theory as opposed to most of the existing strategies employing model-based optimization techniques. TUC consists of four distinct interconnected control modules that allow for real-time control of the following: (a) green times (split); (b) cycle time; (c) offset (green wave along an arterial), as well as (d) provision of public transport priority. These four control modules are complemented by a fifth data processing module. All control modules are based on feedback concepts of various types, which leads to TUC's computational simplicity as compared to model-based optimization approaches, without actually sacrificing efficiency. In this paper, we will concentrate on the fine-tuning of the TUC split and cycle control module parameters. Note that, the proposed algorithm can be applied to the fine-tuning of the other control modules as well.

We next briefly describe the control laws applied by the split and cycle control modules. The split is a network-wide control module, i.e., all available measurements are used to calculate the green time of each stage via the multivariable regulator

$$g_{t_s} = g^N - Lz_{t_s} \quad (9)$$

where $t_s = 0, 1, 2, \dots$ is the discrete time index with sample time period typically equal to the cycle time duration C ; the vector g_{t_s} includes the green times of all stages in all junctions to be applied during the next cycle; g^N is a prespecified vector of fixed green times (fixed plan) whose impact on the resulting control was found to be limited; the vector z_{t_s} comprises the numbers of vehicles in each network link during the last cycle, estimated by the data processing module; L is the control matrix (with dimensions number-of-stages/number-of-links) which results from an off-line applied software code based on the Linear-Quadratic regulator design procedure. The traffic data required to calculate L are: saturation flows of links; average turning rates at junctions; maximum numbers of vehicles $z_{i,\max}$ in links. The aim of (9) is to balance the relative space occupancies $z_i/z_{i,\max}$ in the network links, so as 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 (9) may apply an inherent gating, i.e., reduce the green time of links that feed a saturating road, even if these links are two or more junctions away.

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

$$z_{i,t_s} = z_{i,\max} f(o_{i,t_s}, l_i) b_i \quad (10)$$

where o_{i,t_s} denotes the measured average time-occupancy (measured usually by loop-detectors located at a certain distance from the stop light) during the last cycle time; $f(\cdot)$ is an empirical function [6], [23] constructed from practical investigations; l_i denotes the distance of the loop-detector from the stop line divided by the total link length; finally, the nonnegative variable b_i is used to denote the influence of the link i to TUCs control decisions. These parameters are as many as the network links and are the ones fine-tuned by the AFT.

Cycle control is another module 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 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 as criterion for the increase or decrease of the cycle, the current saturation level of a prespecified percentage of the network links. The cycle module control law takes the form:

$$C_{t_c} = \begin{cases} C^N + K_1(\sigma_{t_c} - \sigma_{N_1}) & \text{if } \sigma_{t_c} \leq \sigma_{cr} \\ C^N - K_2(\sigma_{t_c} - \sigma_{N_2}) & \text{if } \sigma_{t_c} > \sigma_{cr} \end{cases} \quad (11)$$

where $t_c = 0, 1, 2, \dots$ is the discrete time index of the cycle control; C^N denotes a nominal network cycle time; the vector σ_{t_c} comprises the mean values of the space occupancies, for the prespecified links, over the last cycle control period; $\sigma_{N_1}, \sigma_{N_2}, \sigma_{cr} \in [0, 1]$ denote user-defined design parameters; $K_1, K_2 > 0$ are network-wide control parameters, the selection of which affects the intensity of the control reactions. In other words, high K_1, K_2 values force the control law to react strongly even for small differences of σ_{t_c} from σ_{N_i} . After the application of (11), 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. The variant interpretable fine-tuned parameters of the cycle control module are: $K_1, K_2, \sigma_{N_1}, \sigma_{N_2}, \sigma_{cr}$.

5.2 Network and simulation set-up

For the simulation experiments of the proposed approach the road network of the city centre of Chania, Greece, is considered. The model of the network (Fig. 2) consists of 16 signalized junctions (nodes) and 60 links (arrows), which receive green light independently. 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 of the stop line. It is noteworthy that severe congestion problems occur in the actual Chania network, which sometimes leads to grid-lock situations.

The microscopic traffic simulator AIMSUN (v. 6.0.1) [28], [29] was used for the simulation experiments. The simulation step was set to 0.5 s. The traffic network characteristics (saturation flows, turning rates) and the fixed plan g^N in (9) used in AIMSUN and in TUC were suggested by the operators of the Traffic Control Centre (TCC) of the city. Note that, the fixed plan g^N is one of the six field-applied fixed plans used by the TCC. Two basic traffic demand scenarios (time-history of vehicles entering the network in the network origins during the day) were designed based on actual measurements, in order to investigate the behaviour of the methodology under different traffic conditions. The simulation horizon for each scenario is 4 hours. Scenario 1 comprises medium demand in all network origins, while scenario 2 comprises high demand and the network faces serious congestion for some 2 hours with some link queues

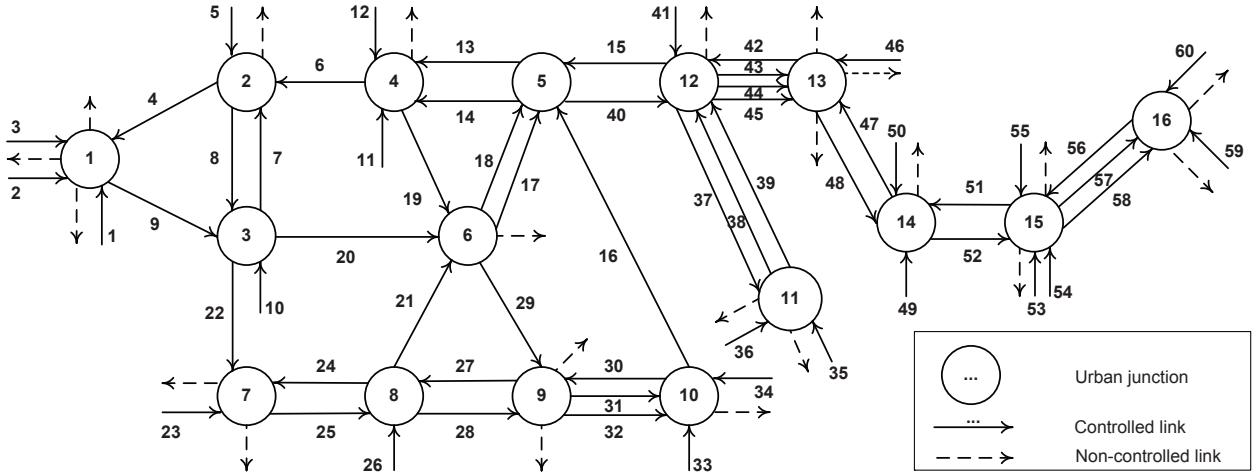


Fig. 2. The Chania urban road network.

spilling back into upstream links. Each day – iteration of the AFT – a randomly perturbed 5%-width version of the basic demand scenarios was used. Note that a 5%-width random perturbation makes small difference in the whole network’s traffic behaviour for the first scenario, but may cause intense system oscillations to demand scenario 2.

The TUC parameters used for the simulation experiments were $t_s = C, t_c = 600s, C_{min} = 60s, C_{max} = 120s, C^N = C_{min}$. Also, for the implementation of AFT the following parameter values were used for the experiments: $T_h = 90, \bar{L}_g = 150, K = 20, \alpha_k = \alpha = 0.1$ and initial values to λ_i according to $\lambda_1 = 100, \lambda_3 = 0.1, \lambda_2 = \lambda_4 = 0$. The assessment criterion $J \equiv ms$ was set the actual daily network mean speed, as calculated by the simulator AIM-SUN. Finally, the initial values for the tunable parameters were chosen so as to correspond to values that usually are chosen during the initial implementation of TUC in practical applications. More precisely, the parameters θ were initialized according to $b_i = 1\forall i, K_1 = 240, K_2 = 300, \sigma_{N_1} = 0.15, \sigma_{N_2} = 0.6, \sigma_{cr} = 0.4$.

5.3 Simulation Results

The AFT exhibited an impressive performance when applied to both medium and high traffic demand scenarios as shown in Fig. 3 and 4. More precisely, AFT achieves to optimize the overall system performance within few days, by efficiently fine-tuning both split and cycle control modules’ parameters, while avoiding decreasing the daily mean speed lower than the initial point. The trajectory of the system performance (mean speed) is persistently increasing in both scenarios until it converges to a local maximum value (Fig. 3). This corollary is not clearly represented in Fig. 4, due to the high sensitivity of the system performance to the random changes of the demand scenario, but still in terms of average values the system performance is improved by the use of AFT.

Table 3 displays the average mean speed of the experiments for each scenario, without the use of AFT, with the implementation of AFT, as well as the standard deviation of the mean speed for every simulated scenario. The comparison of the simulation results indicates that AFT achieves an average improvement to the system perfor-

mance of some 14% for demand scenario 1 and some 31% for the higher demand scenario 2.

Fig. 5–10 exhibit the trajectories of the system tunable parameters for both simulated scenarios. AFT increases the value of the impact factors b_i (split control parameters) for the important links of the network, while for others – not so crucial for the overall system performance – this weight is decreased (Fig. 5 and 8). Finally, the cycle control parameters are also fine-tuned according to the special characteristics of each scenario (Fig. 6–7 and 9–10).

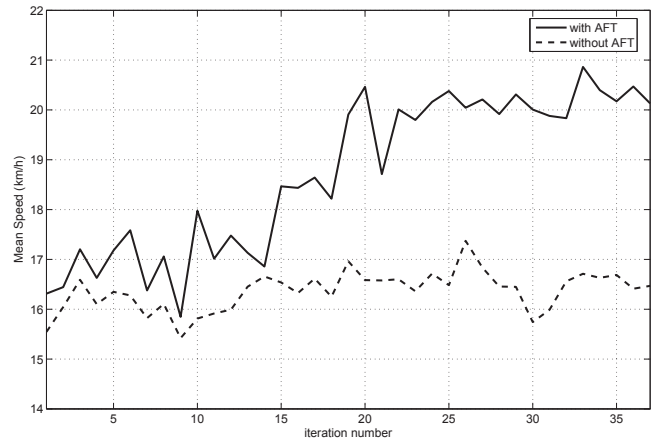


Fig. 3. Mean speed trajectory for the demand scenario 1.

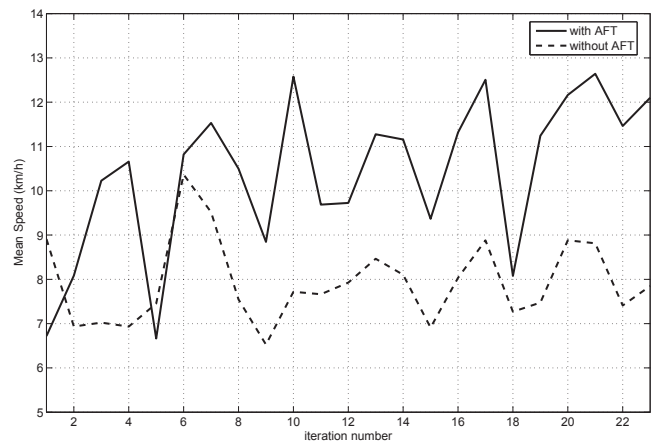


Fig. 4. Mean speed trajectory for the demand scenario 2.

Table 3. Comparison of the average mean speed (ms).

scenario	average ms (km/h)		ms deviation (km/h)	comparison average ms
	without AFT	with AFT	without AFT	
1	16.39	18.72	0.39	14.22%
2	7.94	10.41	0.94	31.11%

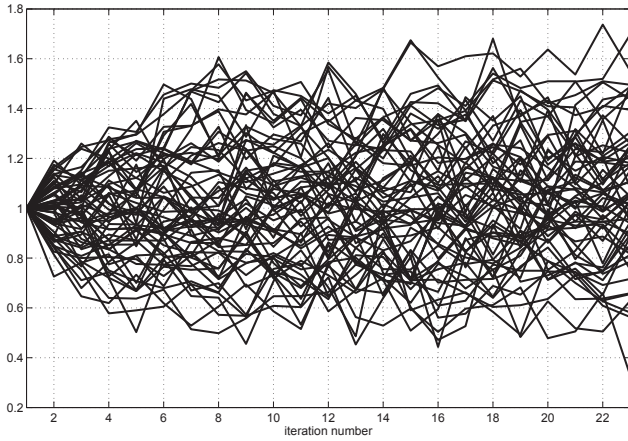


Fig. 5. Trajectories of the split control parameters (scenario 1).

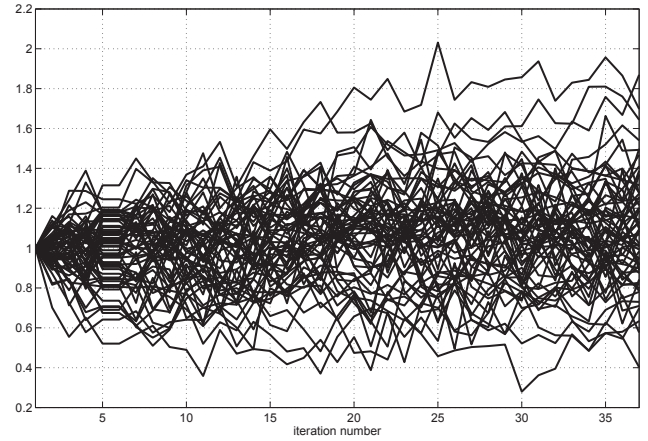


Fig. 8. Trajectories of the split control parameters (scenario 2).

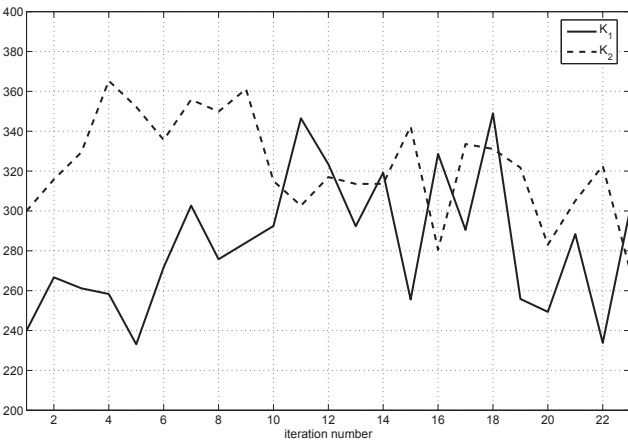


Fig. 6. Trajectories of the cycle parameters K_1 , K_2 (scenario 1).

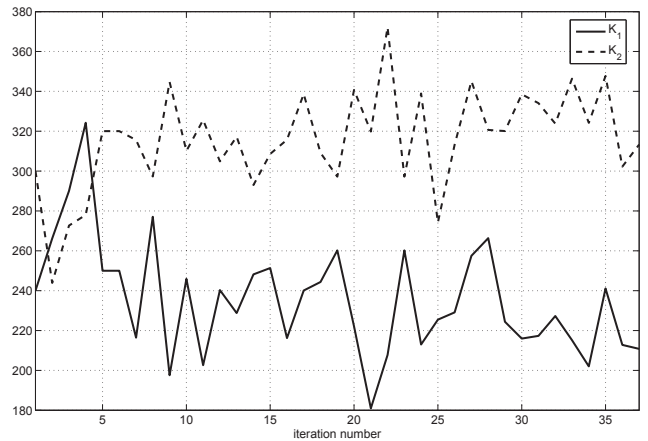


Fig. 9. Trajectories of the cycle parameters K_1 , K_2 (scenario 2).

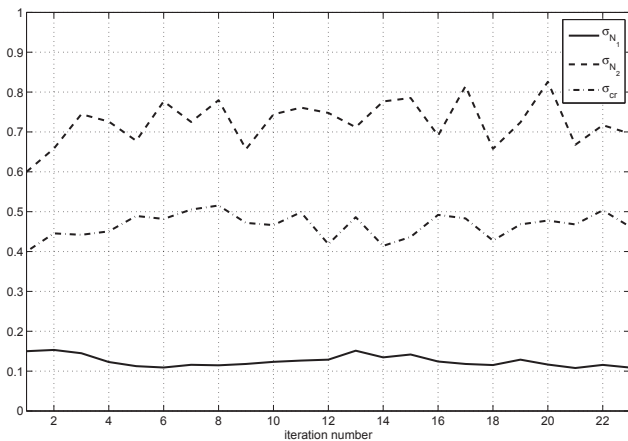


Fig. 7. Trajectories of the cycle parameters σ_{N_1} , σ_{N_1} , σ_{cr} (scenario 1).

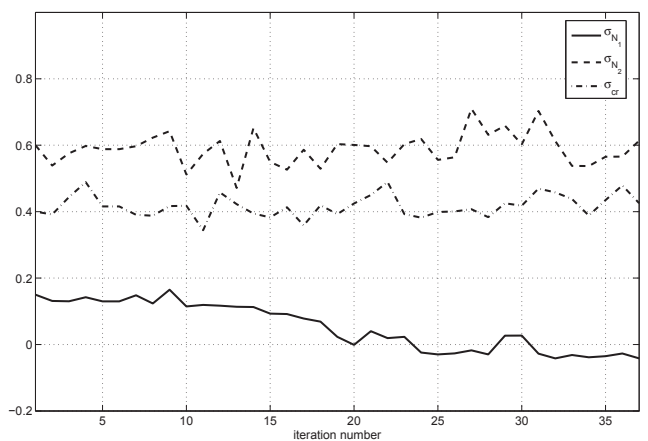


Fig. 10. Trajectories of the cycle parameters σ_{N_1} , σ_{N_1} , σ_{cr} (scenario 2).

The average computational time for every iteration of the algorithm is about 20 min, which means that the implementation of AFT in a real-time large-scale application

would be feasible, regardless of the type of the operating Traffic Control System. At the end of each day AFT collects all the necessary data and calculates the new tunable

parameters to be applied next day aiming to improve the Traffic Control System performance.

6. CONCLUSIONS AND FUTURE WORK

Preliminary simulations results using the complicated traffic network of the city of Chania, establish the comparative efficiency of the AFT algorithm. Results indicate that AFT can improve the overall system performance, when used to fine-tune the parameters of the mutually-interacting split and cycle control modules of the urban control strategy TUC. Future work will deal with experiments of AFT with all TUC control parameters (split, cycle, offset), as well as with investigations on on-line generation of the gain sequences α_k , which could possibly increase the efficient performance of the algorithm.

REFERENCES

- [1] E. B. Kosmatopoulos, M. Papageorgiou, A. Vakouli and A. Kouvelas, "Adaptive fine-tuning of non-linear control systems with application to the urban traffic control strategy TUC", *IEEE Transactions on Control Systems Technology*, vol. 15, no. 6, pp. 991-1002, 2007.
- [2] E. B. Kosmatopoulos, "An adaptive optimization scheme with satisfactory transient performance", *Automatica*, in press.
- [3] E. B. Kosmatopoulos and A. Kouvelas, "Large-Scale NCS Fine-Tuning through Learning", *IEEE Transactions on Neural Networks*, in press.
- [4] E. B. Kosmatopoulos, "Adaptive control design based on adaptive optimization principles", *IEEE Transactions on Automatic Control*, vol. 53, no. 11, pp. 2680-2685, 2008.
- [5] F. Middelham and H. Taale, "SCOOT compared", *3rd ITS World Congress*, Orlando, FL, U.S.A., 1996, (available only electronically); see also, F. Middelham, *et al.*, "Assessment of the SCOOT system in Nijmegen", *IEEE Conference Publication*, vol. 422, pp. 66-70, 1996.
- [6] C. Diakaki, V. Dinopoulou, K. Aboudolas, M. Papageorgiou, E. Ben-Shabat, E. Seider, and A. Leibov, "Extensions and new applications of the traffic signal control strategy TUC", *Transportation Research Record*, no. 1856, pp. 202-216, 2003.
- [7] P. Ioannou, (ed), "Intelligent Freight Transportation", CRC press, 2008.
- [8] L. Li, N. Tang, X. Mu, F. Shi, "Implementation of traffic lights control based on Petri nets", *IEEE Transactions on Intelligent Transportation Systems*, vol. 3, pp. 1749-1752, 2003.
- [9] J. Sánchez, M. Galn, E. Rubio, "Applying a traffic lights evolutionary optimization technique to a real case: Las Ramblas" area in Santa Cruz de Tenerife", *IEEE Transactions on Evolutionary Computation*, vol. 12, pp. 25-39, 2008.
- [10] H. Julia, M. Dessouky, P. Ioannou, A. Chassiakos, "Container movement by trucks in metropolitan networks: Modeling and optimisation", *Transportation Research Part E*, vol. 41, no. 3, pp. 235-259, 2005.
- [11] D.B. Giralda, M.A. Rodriguez, F.J.D. Pernas, J.F.D. Higuera, D.G. Ortega, M.M. Zarzuela, "Intelligent system for dynamic transport fleet management", *IEEE Symposium on Emerging Technologies and Factory Automation, ETFA 1 2 VOLS*, art. no. 1612603, pp. 773-776, 2005.
- [12] P.-c.B. Liu, M. Hansen, A. Mukherjee, "Scenario-based air traffic flow management: From theory to practice", *Transportation Research Part B*, 2008.
- [13] J. C. Spall, "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation", *IEEE Transactions on Automatic Control*, vol. 37, pp. 332-341, 1992.
- [14] J.C. Spall, "Adaptive stochastic approximation by the simultaneous perturbation method", *IEEE Transactions on Automatic Control*, vol. 45, pp. 1839-1853, 2000.
- [15] <http://www.jhuapl.edu/SPSA/>.
- [16] D.C. Chin and R.H. Smith, "A Traffic Simulation for Mid-Manhattan with Model-Free Adaptive Signal Control", *Proceedings of the Summer Computer Simulation Conference*, pp. 296-301, 1994.
- [17] J.C. Spall and D.C. Chin, "Traffic-Responsive Signal Timing for System-Wide Traffic Control", *Transportation Research Part C*, vol. 5, pp. 153-163, 1997.
- [18] M. C. Choy, D. Srinivasan and R. L. Cheu, "Neural networks for continuous online learning and control", *IEEE Transactions on Neural Networks*, vol. 17, no. 6, Pages 1511-1531, 2006.
- [19] N. L. Kleinman, S. D. Hill and V. A. Irenda, "SPSA/ SIMMOD Optimization of Air Traffic Delay Cost", *Proceedings of the American Control Conference*, 4-6 June 1997, Albuquerque, NM, pp. 1121-1125, 1997.
- [20] R. Burnett, "Application of Stochastic Optimization to Collision Avoidance", *Proceedings of the American Control Conference*, 29 June-2 July 2004, Boston, MA, pp. 2789-2794, 2004.
- [21] S.D. Hill and M.C. Fu, "Transfer Optimization via Simultaneous Perturbation Stochastic Approximation", *Proceedings of 27th the Winter Simulation Conference*, pp. 242-249, Arlington, Virginia, 1995.
- [22] M. I. Koch, D. C. Chin and R. H. Smith, "Network-Wide Approach to Optimal Signal Light Timing for Integrated Transit Vehicle and Traffic Operations", *Proceedings of the 7th National Conference on Light Rail Transit*, vol. 2, National Academy of Sciences Press, pp. 126-131, 1997.
- [23] E. B. Kosmatopoulos, M. Papageorgiou, V. Dinopoulou, Ch. Bielefeldt, R. Morris, J. Mueck, A. Richards, F. Weichenmeier, "International comparative field evaluation of a traffic signal control strategy in three cities", *Transportation Research Part A*, vol. 40, no. 5, pp. 399-413, 2006.
- [24] C. Diakaki, M. Papageorgiou, K. Aboudolas, "A multivariable regulator approach to traffic-responsive network-wide signal control", *Control Engineering Practice*, 10, pp. 1831-1835, 2002.
- [25] B. Harbord, J. White, K. McCabe, A. Riley, and S. Tarry, "A flexible approach to motorway control", ATKINS technical paper, http://www.atkinstransportsystems.com/controlled_motorways.html.
- [26] V. Maiorov and R. S. Meir "Approximation bounds for smooth functions in $C(\mathcal{R}^d)$ by neural networks and mixture networks", *IEEE Transactions on Neural Networks*, vol. 9, no. 5, pp. 969-978, Sept. 1998.
- [27] G.-B. Huang, L. Chen and C. K. Sew, "Universal approximation using incremental constructive feedforward networks with random hidden nodes", *IEEE Transaction on Neural Networks*, vol. 17, pp. 879-892, 2006.
- [28] TSS-Transport Simulation Systems, "AIMSUN microscopic traffic simulator: a tool for the analysis and assessment of ITS systems", *TSS-Transport Simulation Systems/HCC Simulation Meeting*.
- [29] <http://www.aimsun.com/site/>.
- [30] E.B. Kosmatopoulos, M.M. Polycarpou, M.A. Christodoulou, and P.A. Ioannou, "High-order neural network structures for identification of dynamical systems", *IEEE Transactions on Neural Networks*, vol. 6, no. 2, pp. 422-431, 1995.
- [31] E.B. Kosmatopoulos, M.A. Christodoulou, and P.A. Ioannou, "Dynamical neural networks that ensure exponential error convergence", *Neural Networks*, vol. 10, no. 2, pp. 299-314, 1997.