A Switching-Based Adaptive Dynamic Programming Method to Optimal Traffic Signaling

Liu, Di; Yu, Wenwu; Baldi, Simone; Cao, Jinde; Huang, Wei

Published in:
IEEE Transactions on Systems, Man, and Cybernetics: Systems

DOI:
10.1109/TSMC.2019.2930138

Publication date:
2020

A Switching-Based Adaptive Dynamic Programming Method to Optimal Traffic Signaling

Di Liu, Wenwu Yu, *Senior Member, IEEE*, Simone Baldi, Jinde Cao, *Fellow, IEEE*, and Wei Huang

***Abstract*—The work presented in this paper concerns a switching-based control formulation for multi-intersection and multiphase traffic light systems. A macroscopic traffic flow modeling approach is first presented, which is instrumental to the development of a model-based and switching-based optimization method for traffic signal operation, in the framework of adaptive dynamic programming (ADP). The main advantage of the switching-based formulation is its capability to determine both "when"' to switch and "which" mode to switch on without the need to use the cycle-based average flow approximation typical of state-of-the-art formulations. In addition, the framework can handle different cycle times across intersections without the need for synchronization constraints and, moreover, minimum dwell-time constraints can be directly enforced to comply with minimum green/red times in each phase. The simulation experiments on a multi-intersection and multiphase traffic light systems are presented to show the effectiveness of the method.**

***Index Terms*—Adaptive dynamic programming (ADP), dwell-time switching, model-based and switching-based optimization, traffic flow model, traffic signal operation.**

Manuscript received November 23, 2018; accepted July 17, 2019. Date of publication August 6, 2019; date of current version October 15, 2020. This work was supported in part by the Fundamental Research Funds for the Central Universities (RECON-STRUCT) under Grant 4007019109, in part by the Special Guiding Funds for Double First-Class under Grant 4007019201, in part by the National Natural Science Foundation of China under Grant 61673107 and Grant 61833005, in part by the National Ten Thousand Talent Program for Young Top-Notch Talents under Grant W2070082, in part by the General Joint Fund of the Equipment Advance Research Program of Ministry of Education under Grant 6141A020223, and in part by the Jiangsu Provincial Key Laboratory of Networked Collective Intelligence under Grant BM2017002. This paper was recommended by Associate Editor J. Lu. *(Corresponding author: Wenwu Yu.)*

D. Liu is with the School of Cyber Science and Engineering, Southeast University, Nanjing 210096, China (e-mail: liud923@126.com).

W. Yu is with the School of Cyber Science and Engineering, Southeast University, Nanjing 210096, China, and also with the School of Mathematics, Southeast University, Nanjing 210096, China (e-mail: wwyu@seu.edu.cn).

S. Baldi is with the School of Mathematics, Southeast University, Nanjing 210096, China, and also with the Delft Center for Systems and Control, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: s.baldi@tudelft.nl).

J. Cao is with the Jiangsu Provincial Key Laboratory of Networked Collective Intelligence, Southeast University, Nanjing 210096, China, and also with the School of Mathematics, Southeast University, Nanjing 210096, China (e-mail: jdcao@seu.edu.cn).

W. Huang is with the Intelligent Transportation System Research Center, Southeast University, Nanjing 210096, China (e-mail: hhhwei@126.com).

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/TSMC.2019.2930138

I. Introduction

TRAFFIC congestion has become a serious problem on the agenda of many public/private stakeholders, due to the constantly increasing urban traffic volumes, and to the lack of space and public funds to construct new transportation infrastructure. These problems are coupled with the complexity of understanding, modeling, and controlling the dynamics of traffic networks [1]–[3]. In fact, as an indispensable part of any traffic control department, traffic signal operations play an important role in the effective functioning of the urban traffic. A significant traffic engineering challenge is to find more intelligent traffic signaling methods to make transportation more efficient [3]–[5].

Due to their complexity, there is still no common agreement on the best description for the dynamics of traffic networks [6]–[9]. Recent research showed that we can distinguish at least two main families in this area. The first family is the *microscopic* simulation-based approach, which uses historical traffic data to build a vehicle-based simulation environment of the traffic network. Then, in combination with artificial intelligence learning methods, one can forecast the future states and design optimal traffic signal policies [10]–[16]. For example, researchers have proposed to control traffic lights in real time by means of a reinforcement learning [12]–[14]. Li *et al.* used deep neural networks to learn the Q-function from the sampled traffic state/control inputs and the corresponding traffic system performance output. Then, based on the deep neural networks, they found appropriate signal timing policies [13]. Liang *et al.* [14] proposed a deep reinforcement learning model to decide the traffic signals' duration based on the collected data from different sensors and vehicular networks. Reinforcement learning was also proposed for decision making of intelligent vehicles [17]–[19]. The curse of dimensionality is the main problem of microscopic frameworks: in fact, because the model describes dynamics at the vehicle level, the state easily becomes extremely large, making optimization prohibitive. Most of the aforementioned works involve only a single intersection, while extension to multiple intersections seems in general prohibitive. Some methods have been proposed in the literature for tackling such dimensionality issues. Tahifa *et al.* [20] utilized the multiagent framework to model a traffic network and demonstrated the effectiveness of cooperative swarm Q-learning for traffic signal control. Multiagent theory alleviates the curse of dimensionality by breaking the optimization

into subproblems, but convergence guarantees for multiagent reinforcement learning can be provided only under strong assumptions. Prashanth and Bhatnagar [21] developed a Q-learning-based reinforcement learning algorithm with function approximation. Function approximation alleviates the curse of dimensionality but poses the problem of feature selection. Again, the convergence guarantees for reinforcement learning with function approximation is not easy to get.

The second family of methods to describe the dynamics of traffic systems is the *macroscopic* model-based approach, which can capture the aggregate dynamics of traffic flow. In other words, while microscopic models describe what happens at the single-vehicle level (or sometimes at a single-cell level), macroscopic models capture average characteristics of the traffic flow. Therefore, the macroscopic approach can intrinsically reduce the curse of dimensionality, at the expense of less detailed modeling. In recent years, several macroscopic traffic models have been proposed to describe the dynamics of urban traffic networks [22]–[25]. Widely adopted models include the Store-and-forward model [26], [27], the BLX-model [28], and the S-model [29]–[31]. Based on such models, a number of model-based optimization control strategies have been studied [26], [29]–[32]. The common feature of the Store-and-forward, BLX, and S-models is to take the cycle time as the sampling interval and to average the vehicle flow across one cycle time [26], [30], [31] (we remind that a cycle is the time period in which the set of signal phases is complete). In other words, instead of describing what happens for each vehicle at a certain time step, one gets a description of the average of the vehicle flow across one cycle time. By doing this, the curse of dimensionality is certainly reduced, but extra structural restrictions must be imposed on the network: most notable macroscopic frameworks assume that the cycle time homogeneous in the network, and they treat the control variable (green time) as a continuous function.

These structural restrictions are often unrealistic and create two problems. The first problem is that additional constraints must be taken when the cycle time in the network is not homogeneous, leading to nonconvex optimization problems. For example, in the Store-and-forward model, one should "rescale and project" the optimal solution to a linear-quadratic problem, in such a way that minimum green/red times or nonhomogeneous cycle time can be handled [26]; in the S-model, nonhomogeneous cycle time leads to considering the synchronization constraints among different intersections. Clearly, the constraints give rise to some feasibility problems that might be difficult to analyze. The second problem is that cycle-based sampling time cannot capture what happens in between a cycle time. In fact, the cycle-based sampling time gives a rough (average) approximation of the actual traffic dynamics, which should exhibit a switching behavior (change of regime) between the green and the red phases of the intersections [26]. In view of the aforementioned issues, an open problem in macroscopic traffic modeling and control seems to be how to overcome the structural restrictions typical of the state-of-the-art: a promising framework in this direction seems to be the so-called *switched systems* framework. In other words, a traffic signal network can be seen as a giant switching system composed of many different traffic light phases, where at each switching instant only one phase is active: the cycle-based average dynamics are in general just a rough approximation of the true switching dynamics (see [33], where the differences between the average and switching dynamics at low and high frequency are discussed). When the switching occurs at low frequency, which is the case of traffic lights, the average dynamics can be quite far from the switched dynamics. This is also recognized (implicitly) by the Store-and-forward, BLX, and S-model, which distinguish among the under-saturated and saturated case depending on whether the queue can be served within one cycle time or not. On the other hand, by explicitly taking into account the switching among different phases, one can obtain traffic dynamics which are closer to reality, and leverage on recently developed optimization approaches for switched systems [34]–[38].

In this paper, we propose a novel model-based and switching-based frameworks for traffic signal operation: the framework does not use cycle-based sampling time and allows us to determine both which phase to switch on and when to switch it on. In addition, minimum dwell-time constraints can be easily imposed, to comply with a minimum green/red time in each phase. The main contributions of this paper can be summarized as follows.

1) We propose a switching-based model to describe multi-intersections and multiphases traffic light systems. Based on this model, appropriate adaptive dynamic programming (ADP) methods are used to seek the optimal traffic light policy. To the best of our knowledge, it is the first time that such a switching-based ADP method is proposed for optimal traffic signal operation.
2) Some advantages over the Store-and-forward, BLX, and S-models arise, that is, no need to average the dynamics over one cycle time. The sampling time can be selected by the designer to the desired accuracy. In addition, we can more directly impose minimum green/red time in terms of minimum dwell-time constraints, without resorting to constraining the continuous solution as in the Store-and-forward model. Finally, we can directly handle different cycle times at different intersections without the need to impose synchronization constraints as in the S-model.
3) We make use of the structure of the system to define new ADP heuristics (in the form of the piecewise smooth neural network approximators) that can take into account some structural and nonlinear characteristics of the problem. The effectiveness of the method is presented via simulations on a benchmark traffic network.

The rest of this paper is organized as follows. Section II proposes the macroscopic urban traffic model and gives the problem formulation; Section III presents the optimization framework; Section IV provides the traffic network benchmark and some simulations; and Section V is the conclusion.

II. Proposed Macroscopic Urban Traffic Model

In this section, we present the proposed urban traffic model. The model is a macroscopic flow-based model. As

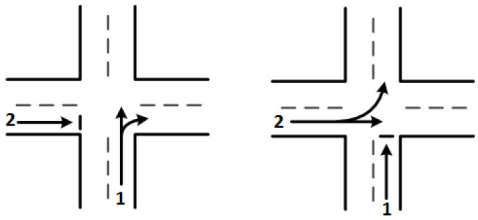

Fig. 1. Phases for an illustrative single intersection.

compared with the BLX model [28], the Store-and-forward model [26], [27], and the S-model [29]–[31], where the flow is averaged over one cycle time, in our case, we can average the flow dynamics with finer precision than one cycle time. In fact, a sampling time T, typically much smaller than the cycle time, can be selected by the designer for the desired accuracy. Then, the flow can be approximated over T.

In the following, the notation x_{k+1} will be used to indicate $x(k+1)$ and the notation x_k will be used to indicate $x(k)$. Let us use a simple single-intersection example to describe the modeling approach. With reference to Fig. 1, let us define x_i to be the queue length [in veh] at link i. Because the example under consideration has two links and two phases, it is not difficult to see that the dynamics can be represented as follows.

Phase 1:

$$\begin{cases} x_{1_{k+1}} = x_{1_k} + \big(\alpha_1^{\text{in}} - \big(\beta_{1,r} + \beta_{1,s}\big)\mu_1\big)T \\ x_{2_{k+1}} = x_{2_k} + \alpha_2^{\text{in}} T. \end{cases} \tag{1}$$

Phase 2:

$$\begin{cases} x_{1_{k+1}} = x_{1_k} + \alpha_1^{\text{in}} T \\ x_{2_{k+1}} = x_{2_k} + \big(\alpha_2^{\text{in}} - \big(\beta_{2,l} + \beta_{2,s}\big)\mu_2\big)T. \end{cases} \tag{2}$$

The dynamics are discrete time, where $k+1$ indicates the time after T seconds; $\beta_{i,r}$, $\beta_{i,s}$, and $\beta_{i,l}$ indicate the turning rates [in %] at link i, i.e., the vehicles going right, straight, and left, respectively; α_i^{in} indicates the inflow rate [in veh/s] at origins; and μ_i indicates the outflow rate [in veh/s] of each link. The equations above simply indicate that the number of vehicles in a link facing the red light can only increase due to the inflow rate, whereas for a link facing the green light there is also an outflow to the downstream links.

Because the number of vehicles cannot go below zero, let us give the following notation.

Phase 1:

$$\begin{cases} x_{1_{k+1}} = \mathbb{P}\big[x_{1_k} + \big(\alpha_1^{\text{in}} - \big(\beta_{1,r} + \beta_{1,s}\big)\mu_1\big)T\big] \\ x_{2_{k+1}} = x_{2_k} + \alpha_2^{\text{in}} T. \end{cases} \tag{3}$$

Phase 2:

$$\begin{cases} x_{1_{k+1}} = x_{1_k} + \alpha_1^{\text{in}} T \\ x_{2_{k+1}} = \mathbb{P}\big[x_{2_k} + \big(\alpha_2^{\text{in}} - \big(\beta_{2,l} + \beta_{2,s}\big)\mu_2\big)T\big] \end{cases} \tag{4}$$

to indicate the projection operator $\mathbb{P}$ that constrains the number of vehicles to be greater or equal to zero

$$x_{i_{k+1}} = \mathbb{P}\big[g\big(x_{i_k}\big)\big] = \begin{cases} 0 & \text{if } \ g\big(x_{i_k}\big) < 0 \\ g\big(x_{i_k}\big) & \text{otherwise.} \end{cases} \tag{5}$$

Note that the projection operator is not necessary when a link faces a red light.

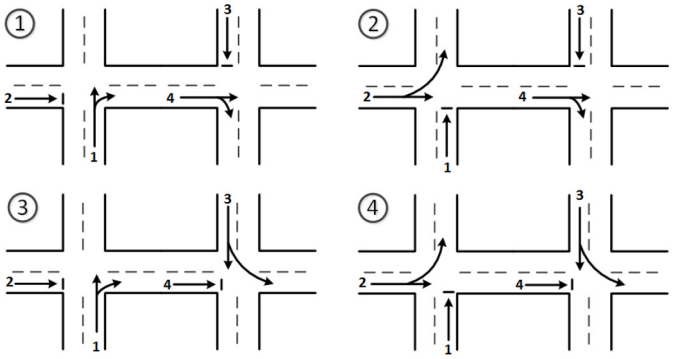

Fig. 2. Phases for an illustrative double intersection.

The dynamics (3) and (4) can be easily extended to multiple intersections: to illustrate how to distribute this idea in case of multiple intersections, let us consider a double-intersection example, connected as in Fig. 2.

Phase 1:

$$\begin{cases} x_{1_{k+1}} = \mathbb{P}\big[x_{1_k} + \big(\alpha_1^{\text{in}} - \big(\beta_{1,r} + \beta_{1,s}\big)\mu_1\big)T\big] \\ x_{2_{k+1}} = x_{2_k} + \alpha_2^{\text{in}} T \\ x_{3_{k+1}} = x_{3_k} + \alpha_3^{\text{in}} T \\ x_{4_{k+1}} = \mathbb{P}\big[x_{4_k} + \big(\beta_{1,r}\mu_1 - \big(\beta_{4,r} + \beta_{4,s}\big)\mu_4\big)T\big]. \end{cases} \tag{6}$$

Phase 2:

$$\begin{cases} x_{1_{k+1}} = x_{1_k} + \alpha_1^{\text{in}} T \\ x_{2_{k+1}} = \mathbb{P}\big[x_{2_k} + \big(\alpha_2^{\text{in}} - \big(\beta_{2,l} + \beta_{2,s}\big)\mu_2\big)T\big] \\ x_{3_{k+1}} = x_{3_k} + \alpha_3^{\text{in}} T \\ x_{4_{k+1}} = \mathbb{P}\big[x_{4_k} + \big(\beta_{2,s}\mu_2 - \big(\beta_{4,r} + \beta_{4,s}\big)\mu_4\big)T\big]. \end{cases} \tag{7}$$

Phase 3:

$$\begin{cases} x_{1_{k+1}} = \mathbb{P}\big[x_{1_k} + \big(\alpha_1^{\text{in}} - \big(\beta_{1,r} + \beta_{1,s}\big)\mu_1\big)T\big] \\ x_{2_{k+1}} = x_{2_k} + \alpha_2^{\text{in}} T \\ x_{3_{k+1}} = \mathbb{P}\big[x_{3_k} + \big(\alpha_3^{\text{in}} - \big(\beta_{3,l} + \beta_{3,s}\big)\mu_3\big)T\big] \\ x_{4_{k+1}} = x_{4_k} + \beta_{1,r}\mu_1 T. \end{cases} \tag{8}$$

Phase 4:

$$\begin{cases} x_{1_{k+1}} = x_{1_k} + \alpha_1^{\text{in}} T \\ x_{2_{k+1}} = \mathbb{P}\big[x_{2_k} + \big(\alpha_2^{\text{in}} - \big(\beta_{2,l} + \beta_{2,s}\big)\mu_2\big)T\big] \\ x_{3_{k+1}} = \mathbb{P}\big[x_{3_k} + \big(\alpha_3^{\text{in}} - \big(\beta_{3,l} + \beta_{3,s}\big)\mu_3\big)T\big] \\ x_{4_{k+1}} = x_{4_k} + \beta_{2,s}\mu_2 T. \end{cases} \tag{9}$$

The main difference as compared to the single-intersection case is that link 4 takes as inflow the vehicles coming from link 2 (going straight during phases 2 and 4) or link 1 (turning right during phases 1 and 3). The turning rates, and inflow and outflow rates have a similar meaning as in the previous single intersection case. It is clear that by connecting appropriately different links, one can extend this modeling methodology to networks of arbitrary topology. In this section, all roads have been taken as one-way roads, which is consistent with the typical Manhattan-like regular networks often considered in [39]. Clearly, two-way roads can be considered after adding more states in the system.

At this point, it is worth comparing the proposed model with the most popular flow-based models adapted in the state-of-the-art, namely, the BLX model [28], the Store-and-forward

model [26], [27], and the S-model [29]–[31]. In such models, each link can be in one of these two conditions.

1) *Saturated:* The link has a continuous outflow of vehicles which is equal to its maximum (saturated) capacity.
2) *Unsaturated:* The link can serve all the cars in queue at the link. The corresponding flow is less than the saturated flow.

In the BLX model, the Store-and-forward model, and the S-model, the above two conditions must be clearly distinguished because the flow is averaged over one cycle time. Therefore, during one cycle time, one can serve all vehicles (unsaturated condition) or continuously have vehicles to serve (saturated condition): in our case, because the sampling time T is smaller than the cycle time, the distinction about the two conditions is made by simply imposing $x_{i_k} \geq 0$ for all k (via the projection operator $\mathbb{P}$). In other words, if for a certain link at a certain time k, there are no more vehicles to serve, yet the traffic light is green, one can impose $x_{i_{k+1}} = x_{i_k}$. To reveal other features of the proposed modeling framework, let us now embed the phase dynamics in a so-called switched system framework.

A. Problem Formulation

It is now possible to embed the phase dynamics previously described section in a discrete-time switched system with M autonomous subsystems [37], [38]

$$x_{k+1} = f_\nu(x_k), k \in \mathbb{Z}_+, \nu \in \mathbb{I}, x_k \in \mathbb{R}^n \tag{10}$$

where $f_\nu : \mathbb{R}^n \to \mathbb{R}^n$ is a continuous vector-valued function where each entry represents the dynamics of a set of links during a certain phase ν. Every phase is represented by a subsystem of the switched system, and the subsystems are indexed by $\mathbb{I} = \{1, 2, \ldots, M\}$. The non-negative integer n denotes the dimension of the state vector $x_k = [x_{1_k} \ldots x_{n_k}]'$, i.e., the number of links (the prime symbol denotes the transpose of a vector). The subscript ν in $f_\nu(\cdot)$ denotes the active subsystem: specifically, only one subsystem is active at time k, which is denoted as ν_k. Let us now formulate an optimal switching problem as one of the minimizing cost functions

$$J = \psi(x_N) + \sum_{i=k}^{k+N-1} \gamma^{i-k} r(x_i) \tag{11}$$

with a horizon of N steps, and with $0 < \gamma \leq 1$ being a discount factor. The function $\psi : \mathbb{R}^n \to \mathbb{R}_+$ is the final cost, while $r(x_i)$ is known as the utility function (or running cost).

Because of the underlying traffic signal control problem, we are interested in considering constrained switching signals. In view of the minimum green time requirements typical of traffic lights, let us define a minimum dwell time as $D \in \mathbb{Z}_+$, and we impose a minimum dwell time constraint on the switching. This implies that the current subsystem (the current phase) has to stay active at least a minimum number of time steps before being able to switch to another subsystem (the next phase).

We are now ready to define the control objective.

Problem: For the switched system (10), the objective is to find a feedback switching policy $\nu(\cdot)$, which can minimize the cost function (11) under the constraint of the minimum dwell time D.

Remark 1: It is worth remarking that, in this paper, both the subsystem sequence and the number of switching are free. This means that the solution of the problem will tell us "when" to switch and "to which mode" to switch on, in such a way that the cost function (11) is minimized. This provides a clear advantage as compared with the BLX model, the Store-and-forward model, and the S-model, in case, the cycle time is different across the intersection.

1) In the BLX and Store-and-forward models, it is typically assumed that the cycle time is the same for all intersections. In general, it is not easy to handle different cycle times: at most, in the Store-and-forward model, it is possible to embed double cycling.
2) In the S-model, additional synchronization constraints must be taken into account when the cycle times differ for the intersections.
3) By embedding the phase dynamics in a switched system, one can easily handle cycle times differing up to multiples of the sampling time T. This is because each mode contains information about the status (green/red) of each phase. However, it must be said that a disadvantage of this strategy is that the number of subsystems will increase exponentially with the number of intersections.

The following section will describe the methodology adopted for the solution of the traffic light problem.

III. Optimization Framework

A. Adaptive Dynamic Programming Approach

This section relies on the tools in [37] and [40], with some ad-hoc modifications that will be clarified later. Initially, let us forget about the dwell-time constraints, in such a way to simplify the presentation. It is well known from ADP that minimizing (11) can be recast as the problem of selecting the optimal policy that minimizes the cost-to-go/value function

$$V^*(x_k) = \min_\nu \left[\psi(x_N) + \sum_{i=k}^{N-1} \gamma^{i-k} r(x_i) \right]. \tag{12}$$

The value function (12) is a function of $\tau := N - k$, i.e., the number of time steps before the end of the horizon N, and of the current state x_k. Then, the optimal switching policy at time k on state x_k is given by

$$\nu^*(x_k) = \arg\min_\nu \left[\psi(x_N) + \sum_{i=k}^{N-1} \gamma^{i-k} r(x_i) \right]. \tag{13}$$

The optimal switching (13) is state-feedback because it depends on the state of the system. However, it is worth noting that switching should also obey the minimum dwell-time constraint that was imposed. Therefore, both the elapsed time d_k of the current subsystem and already active mode ν_{k-1} should play an important role in determining $\nu^*(x_k)$.

1) The optimal $\nu^*(x_k)$ depends on the elapsed time of the current subsystem d_k, because if the minimum dwell time is more than the elapsed time, no switching should be allowed.
2) The optimal $\nu^*(x_k)$ depends on the already active subsystem/mode ν_{k-1} because if ν_{k-1} is actually equal to ν_k^*, no switching will be needed.

According to above arguments, an augmented state of the system (10) can be defined as $w_k := [x_k', d_k, \nu_{k-1}]' \in \Omega := \mathbb{R}^n \times \mathbb{D} \times \mathbb{I}$, where the range of variation of the elapsed time d_k is denoted as $\mathbb{D} := \{1, 2, \ldots, D\}$. Note that $d_k \geq D$ is equivalent to $d_k = D$, i.e., the range of variation of d_k can be represented by a saturation function. Then, the dynamics of w_k is given by

$$w_{k+1} = F_\nu(w_k) := \begin{bmatrix} f_\nu(x_k) \\ \text{sat}\big(I_{\nu_{k-1}}(\nu) d_k + 1\big) \\ \nu \end{bmatrix} \quad \forall w_k = \big[x_k', d_k, \nu_{k-1}\big]' \in \Omega \tag{14}$$

where $I_\nu(\bar{\nu})$ is an indicator function, i.e., $I_\nu(\bar{\nu}) = 1$, if $\nu = \bar{\nu}$ and $I_\nu(\bar{\nu}) = 0$, if $\nu \neq \bar{\nu}$. sat$(\cdot)$ denotes the saturation function for the elapsed time, i.e., when $0 \leq d \leq D$, sat$(d) = d$, and when $d \geq D$, sat$(d) = D$. Summarizing:

1) the term x_{k+1} can be calculated from $f_\nu(x_k)$;
2) $d_{k+1} = \text{sat}(d_k + 1)$ when $\nu = \nu_{k-1}$, and $d_{k+1} = 1$ when $\nu \neq \nu_{k-1}$;
3) the last term of the function $F_\nu(\cdot)$ implies that if ν at time k is chosen, then the active subsystem/mode at the next time step will be ν.

By denoting the value function as $V_\tau^* : \Omega \to \mathbb{R}_+$ (where $\tau := N - k$ is the time before the end of the horizon), one obtains from (11)

$$V_0^*(w_N) = \psi(x_N) \ \forall w_N = \big[x_N', d_N, \nu_{N-1}\big]' \in \Omega \tag{15}$$

and

$$V_{\tau+1}^*(w_k) = r(x_k) + \gamma V_\tau^*\Big(F_{\nu_k^*}(w_k)\Big) \tag{16}$$

$\forall \tau \in T := \{0, 1, \ldots, N-1\}$ $\forall w_k = [x_k', d_k, \nu_{k-1}]' \in \Omega$, where ν_k^* denotes the optimal active subsystem at time k.

Let $M(w_k)$ denote the set of subsystems eligible to be active, given the current state of w_k. Note that $M(w_k)$ depends on w_k because if $d_k < D$, then $M(w_k) = \{\nu_{k-1}\}$ (only one element in the set as the system is not allowed to switch to another subsystem). According to the Bellman optimality principle, we can get

$$V_{\tau+1}^*(w_k) = \min_{\nu \in M(w_k)} \big[r(x_k) + \gamma V_\tau^*(F_\nu(w_k))\big] \tag{17}$$

$\forall \tau \in T$ $\forall w_k \in \Omega$. After obtaining the optimal value function, the optimal switching policy can be obtained by

$$\begin{aligned} \nu_k^*(w_k) &= \arg\min_{\nu \in M(w_k)} \big[r(x_k) + \gamma V_{N-k-1}^*(F_\nu(w_k))\big] \\ &\approx \arg\min_{\nu \in M(w_k)} \big[r(x_k) + \gamma V_N^*(F_\nu(w_k))\big] \end{aligned} \tag{18}$$

whose calculation can be done in real time. The second in (18) is an approximation, for N large enough, given by the fact the discount factor γ can guarantee convergence of the value function. This is necessary because in a traffic system the state (number of vehicles) will never converge to zero for the whole network: therefore, without a discount factor and for $N \to \infty$, the value function would not be finite. Note that in the original formulation [37], a nondiscounted formulation is considered. The next section will propose an algorithm to learn an approximation of the desired value function V_τ^*.

B. Value Function Approximation for Switching Problem

It is well known from the dynamic programming that the desired (approximate) value function (15) and (17) should be derived backward in time, i.e., from $\tau = N$ to $\tau = 0$. For the purpose of the approximating value function in switching problem, and motivated by the development in the HDP literature for switching problems [37], [38], [41], it is proposed to utilize a critic NN to learn the optimal time-dependent value at each time step.

Denote the approximation value function, which is known as critic, as

$$W_{\tau,\nu,d}' \phi(x_k) \approx V_\tau^*(w_k) \tag{19}$$

where $W_{\tau,\nu,d} \in \mathbb{R}^m$ is the unknown optimal weights at time step τ, for the active mode ν, and the elapsed time d, and $\phi(x_k)$ is the basis function of the critic NN, which is a polynomial function composed of the states x_k.

Remark 2: The proposed critic networks turn out be *multiple parametrized critic networks*, i.e., they depend not only on the horizon τ as the actual value function but also on the active mode ν and on the elapsed time d. Taking the approximated value function dependent on the horizon τ, on the active mode ν, and on the elapsed time d certainly increases the number of weights to be trained, but it has the clear advantage that the basis function $\phi(x_k)$ can be taken as dependent on x_k instead of the full state w_k. This means that in order to obtain the weights, i.e., evaluate the approximation over many different samples chosen from Ω, the number of features necessary to train the NN are sensibly reduced (no polynomials over ν and d are necessary in the regressor of the NN [37]).

Let us now denote the state samples with $x^{[j]}$, $j = 1, 2, \ldots, p$, where p is a large positive integer: by exploiting the least squares method one can get

$$W_{\tau,\nu,d} = \arg\min_{W \in \mathbb{R}^m} \sum_{j=1}^{p} \big(W' \phi\big(x^{[j]}\big) - V_\tau^*\big(w^{[j]}\big)\big)^2 \quad \tau = 0, 1, \ldots, N \tag{20}$$

where $V_\tau^*(w^{[j]})$ is approximated using $W_{\tau+1,\nu,d}$ ($w^{[j]}$ represents the fact that the value function is evaluated for the samples $x^{[j]}$, for the active subsystem/mode ν and for the elapsed time d, i.e., $w^{[j]} = [x^{[j]\prime}, d, \nu]'$); we will use $W_{\tau+1,\nu,d}$ to calculate each $W_{\tau,\nu,d}$, i.e., by backward recursions (15) and (17). The starting point of such recursions is W_0 obtained from (15) and (20).

The algorithm used to train the neural network is summarized in Algorithm 1. The algorithm includes two stages: the first stage is offline (steps 1–6). To tune the parameters of the function approximator, this stage involves solving $(N+1)DM$ least squares problems through steps 2 and 4, i.e., this stage is the most expensive stage in terms of computation. In addition, the memory requirement need to store $(N+1)DM$ sets of critic NN weights (i.e., $W_{\tau,\nu,d}$ $\forall \tau = 0, 1, \ldots, N$ $\forall \nu = 1, \ldots, M$ $\forall d = 1, \ldots, D$).

The second stage is online for feedback policy calculation (for online control in real time). The computational cost of this stage is much lower since it just needs to evaluate no more than M scalar-valued functions. Finally, it is worth noting that

Algorithm 1 Switched-Based ADP Based on Multiple Parameterized Critic Networks

1: **Initialization:** Given the state-space system (14) and the cost (11), grid the state space in p points or randomly select p different state samples $x^{[j]}$, $j = 1, 2, \cdots p$, with p being a large positive integer.

2: *Training final network:* Train the network weights W_0 (using least squares) such that

$$W'_{0,\nu,d}\phi(x^{[j]}) = \psi(x^{[j]}) \approx V_0^*(w^{[j]})$$

$\forall j \in \{1, 2, \cdots, p\}$, and with $w^{[j]} = [x^{[j]\prime}, d, \nu]'$ (the approximation at this step is the same for any ν and d).

3: **Offline phase:** Set $\tau = 0$

4: *Approximate optimality principle:* For any ν and d, denote $w^{[j]} = [x^{[j]\prime}, d, \nu]'$. Calculate $V^*_{\tau+1}(w^{[j]})$ by using

$$V^*_{\tau+1}(w^{[j]}) \approx \min_{\nu \in M(w^{[j]})} \left[r(x^{[j]}) + \gamma W'_{\tau,\nu,d}\phi(F_\nu(w^{[j]}))\right]$$

$\forall j \in \{1, 2, \cdots, p\}$, and where $W'_{\tau,\nu,d}\phi(F_\nu(w^{[j]}))$ is the approximation of $V^*_\tau(F_\nu(w^{[j]}))$ based on the weights $W_{\tau,\nu,d}$ calculated at the previous step.

5: *Training backward network:* For each ν and d, use least-squares method to calculate weight $W_{\tau+1,\nu,d}$

$$W_{\tau+1,\nu,d} = \arg\min_{W \in \mathbb{R}^m} \sum_{j=1}^{p} \left[W'\phi(x^{[j]}) - V^*_{\tau+1}(w^{[j]})\right]^2,$$

(the approximation at this step will be different for each ν and d due to the different $V^*_{\tau+1}(w^{[j]})$ associated with each ν and d).

6: Set $\tau = \tau + 1$. Go back to step 4 until $\tau = N$. When $\tau = N$, the offline phase is complete, and go to step 7.

7: **Online phase:** Using the states x_k coming at each time step k from the traffic network, denote $w_k = [x_k\prime, d_k, \nu_{k-1}]'$. Calculate at each time step k

$$\nu_k^*(w_k) = \arg\min_{\nu \in M(w_k)} \left[r(x_k, \nu) + \gamma W'_{N,\nu,d}\phi(x_k)\right]$$

where $W'_{N,\nu,d}\phi(x_k)$ is the approximation of $V_N^*(w_k)$ based on the weights $W_{N,\nu,d}$ calculated at the last step of the offline stage.

as compared to [37], we are using only V_N^* for online control. This is because we want the feedback action to be active over an infinitely long time span. Note that for N long enough, the value function V_N^* will converge to the infinite-horizon discounted-cost solution. While exploiting a similar switched framework as [37], the major advance we will explore in the proposed approach regards the special structure of the approximation for the value function, as it will be explained in the next section.

IV. Traffic Network Benchmark

In this section, the proposed algorithm will be applied to a benchmark traffic network. The benchmark network is taken in a regular (Manhattan-like) network configuration, as it can be found in [30], [39], and [42]–[44]. The performance of the algorithm will be analyzed based on the resulting optimal cost. The Manhattan-like network is shown in Figs. 6 and 7, and the corresponding model for each phase can be written as

$$x_{k+1} = f_\nu(x_k) = \mathbb{P}[x_k + B_\nu T], \nu = 1, \ldots, 16 \tag{21}$$

where the matrices B_ν are reported in the Appendix, and they have been derived using a similar procedure as in Section II.

The basis functions were selected as polynomials of x (note that we do not need to add any polynomial in ν and d because the NN gains will be parameterized accordingly). The accuracy of the approximation capability of the NN can be adjusted by the selection of the order of the polynomials. The training was done over the domain $x = [0, 15]$. In these simulations, we use three different approximators to approximate the value function, which is done in order to highlight different features of the algorithm.

A. Full-States Quadratic Approximator

The basis function is selected as quadratic polynomials of all the states of four intersections

$$V_\tau(w_k) = W'_{\tau,d_k,\nu_{k-1}}\phi(x_k) \tag{22}$$

where the regressor depends only on x_k, because the NN weights $W_{\tau,d_k,\nu_{k-1}}$ depend on d_k and ν_{k-1}.

B. Distributed Quadratic Approximator

Each intersection uses a function approximator composed of local states, leading to a local value function (four value functions for the four intersections). The basis function in each intersection is selected as quadratic polynomials of the states in each intersection, and the value function is approximated as

$$\begin{aligned} V_\tau(w_k) = {} & W'_{1,\tau,d_k,\nu_{k-1}}\phi(x_{1,k}) + W'_{2,\tau,d_k,\nu_{k-1}}\phi(x_{2,k}) \\ & + W'_{3,\tau,d_k,\nu_{k-1}}\phi(x_{3,k}) + W'_{4,\tau,d_k,\nu_{k-1}}\phi(x_{4,k}) \end{aligned} \tag{23}$$

where $x_{1,k}$ contains only the states affecting intersection 1 (which are x_1, x_2, and x_4) and $\phi(x_{1,k})$ are all the monomials of order 2 of $x_{1,k}$ (similarly, for the other intersections). The weights $W_{1,\tau,d_k,\nu_{k-1}}, \ldots, W_{4,\tau,d_k,\nu_{k-1}}$ are indexed by four indices because they are related to a particular intersection. Note that the value function ends up being the sum of local value function, which justifies the term "distributed."

The main advantage of this type of approximator is not distributed computation (even if this is in principle possible, but outside the scope of this paper) because each approximator involves a state of smaller dimension, it allows more easily to test regressor beyond the quadratic one (with the purpose of increasing the precision of the approximation). This is explored in the following class of approximators.

C. Distributed Piecewise Quadratic Approximator

Each intersection uses a function approximator, giving four value functions. The basis function is selected as piecewise quadratic polynomials of the states in each intersection, so as to take explicitly into account the projection term $\mathbb{P}$, which gives rise to multiple linear dynamics. The intuition is that we want to use a quadratic approximator for each one of

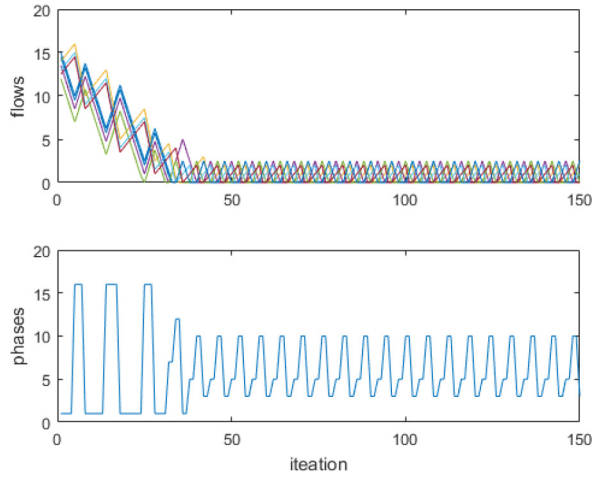

Fig. 3. Full-state approximator: states and phases. The steady-state sequence is 3-5-10.

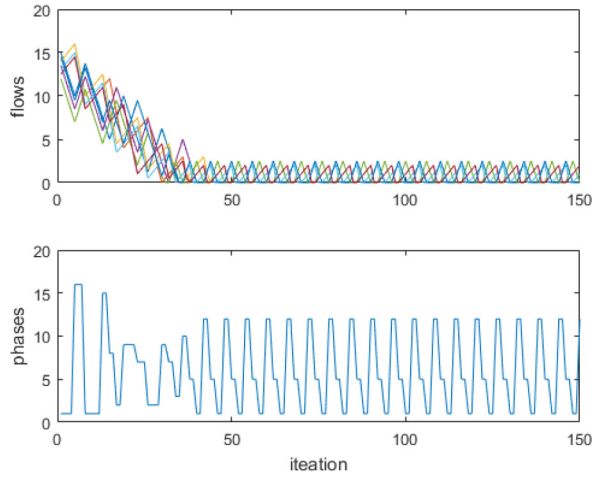

Fig. 4. Distributed quadratic approximator: states and phases. The steady-state sequence is 1-13-5.

these dynamics. First, we consider only two states for each intersection, as one state is common to more intersection. Then, for a value function depending on two states (call them x_a and x_b for simplicity), we have three different piecewise linear dynamics, according to:

1) *Case 1:* $x_a > 0, x_b > 0$;
2) *Case 2:* $x_a = 0, x_b > 0$;
3) *Case 3:* $x_a > 0, x_b = 0$.

This indicates that no more than one state can saturate at a certain time instant. Therefore, each intersection has at most three piecewise linear dynamics: no saturation, saturation of the first or of the second flow. This leads to

$$V_\tau(w_k) = \hat{W}'_{1,\tau,d_k,\nu_{k-1}}\hat{\phi}(x_{1,k}) + \hat{W}'_{2,\tau,d_k,\nu_{k-1}}\hat{\phi}(x_{2,k}) + \hat{W}'_{3,\tau,d_k,\nu_{k-1}}\hat{\phi}(x_{3,k}) + \hat{W}'_{4,\tau,d_k,\nu_{k-1}}\hat{\phi}(x_{4,k}) \quad (24)$$

where $\hat{W}$ and $\hat{\phi}$ are used to indicate the piecewise smooth weights and regressor.

D. Comparisons and Discussion

The simulation experiments are performed for the following initial states and turning rates:

$$\begin{aligned} x_0 &= [15 \quad 12.5 \quad 14 \quad 13.5 \quad 12 \quad 13 \quad 12.5 \quad 14.5]' \\ \alpha^{\text{in}} &= [0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1, 0.1] \\ \mu &= [0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5] \\ \beta_s &= [0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.5] \\ \beta_r &= [0.5, 0, 0, 0.5, 0.5, 0, 0, 0.5] \\ \beta_l &= [0, 0.5, 0.5, 0, 0, 0.5, 0.5, 0] \\ r(x_i) &= x_i' x_i, = \psi(x_N) = x_N' x_N, \quad \gamma = 0.999 \end{aligned}$$

where α^{in}, μ, β_s, β_l, β_r contain α_i^{in}, μ_i, $\beta_{i,s}$, $\beta_{i,l}$, $\beta_{i,r}$ for each link i. The turning rates $\beta_{i,s}, \beta_{i,l}, \beta_{i,r}$ obviously meet the condition $\beta_{i,s} + \beta_{i,l} + \beta_{i,r} = 1$. Both the running and the terminal costs are quadratic with respect to the state. Finally, we take $N = 50$, $D = 2$, and $T = 5$. The simulation results are given in Figs. 3–5 (for the three approximators, respectively). These figures show the evolution of the states starting from the same initial condition, as well as the optimal switching

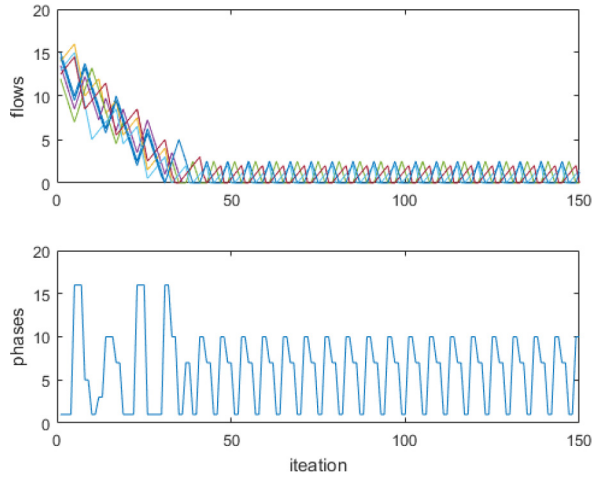

Fig. 5. Piecewise quadratic approximator: states and phases. The steady-state sequence is 1-10-7.

TABLE I
TRAINING TIME FOR THE THREE APPROXIMATORS (OFFLINE STAGE). THE PLATFORM USED IS A DELL PRECISION WORKSTATION WITH INTEL XEON PROCESSOR 3.2 GHZ, 8GB RAM, MATLAB R2017B

Class of approximator	*Training time*
Full states quadratic	43 min
Distributed quadratic	75 min
Piecewise quadratic	108 min

modes/phases. Before the online stage, the weights of the critic were tuned during the offline stage. According to Algorithm 1, the last W_N is obtained using the least-squares method, as in (19). Once W_N is found, (17) can be used for calculating W_{N-1}. Repeating this process backward, all the weights can be found from $k = N$ to $k = 0$ (offline). The training times for the different approximators are reported in Table I.

The actual costs for the different approximators are reported in Table II. Two costs are considered: 1) the total cost is the cost related to the entire simulation from 0 to 150 cycles and 2) the transient cost is the cost in the initial phase, from 0 to 45 cycles. Using local states in the NN approximation (in place of global states) does not lead to loss

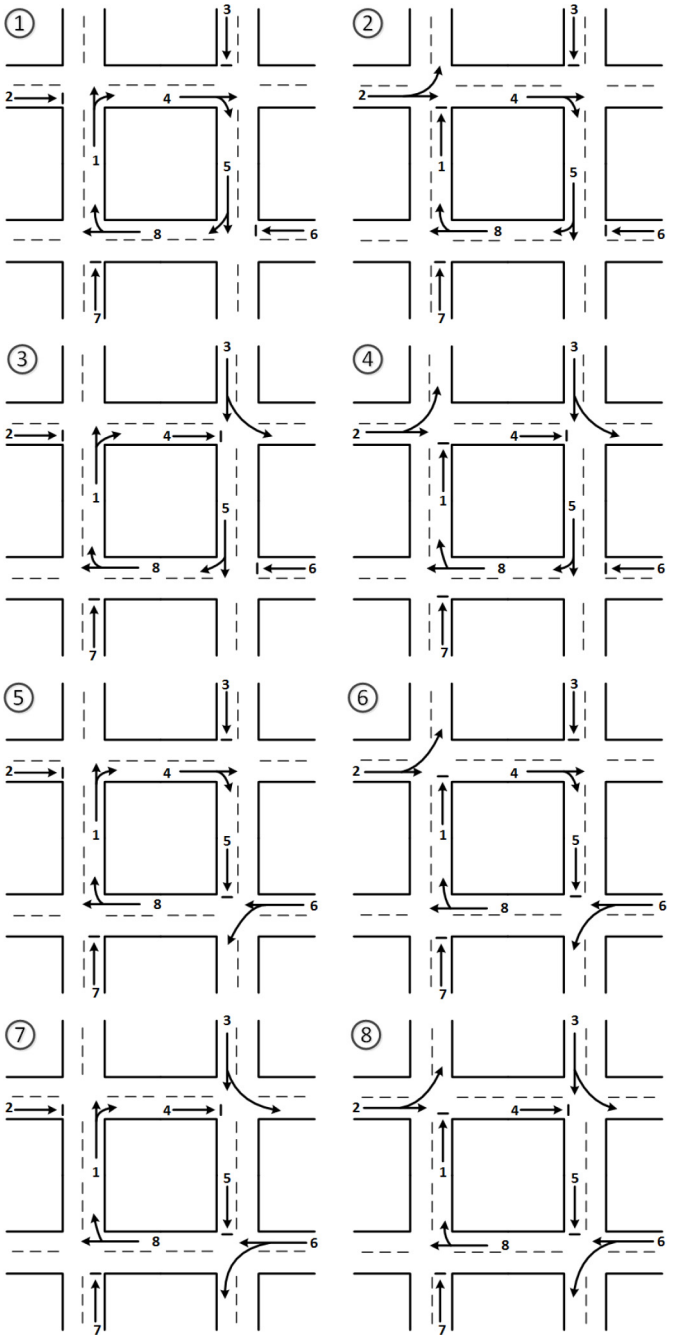

Fig. 6. Phases 1–8 for the Manhattan-like network with four intersections.

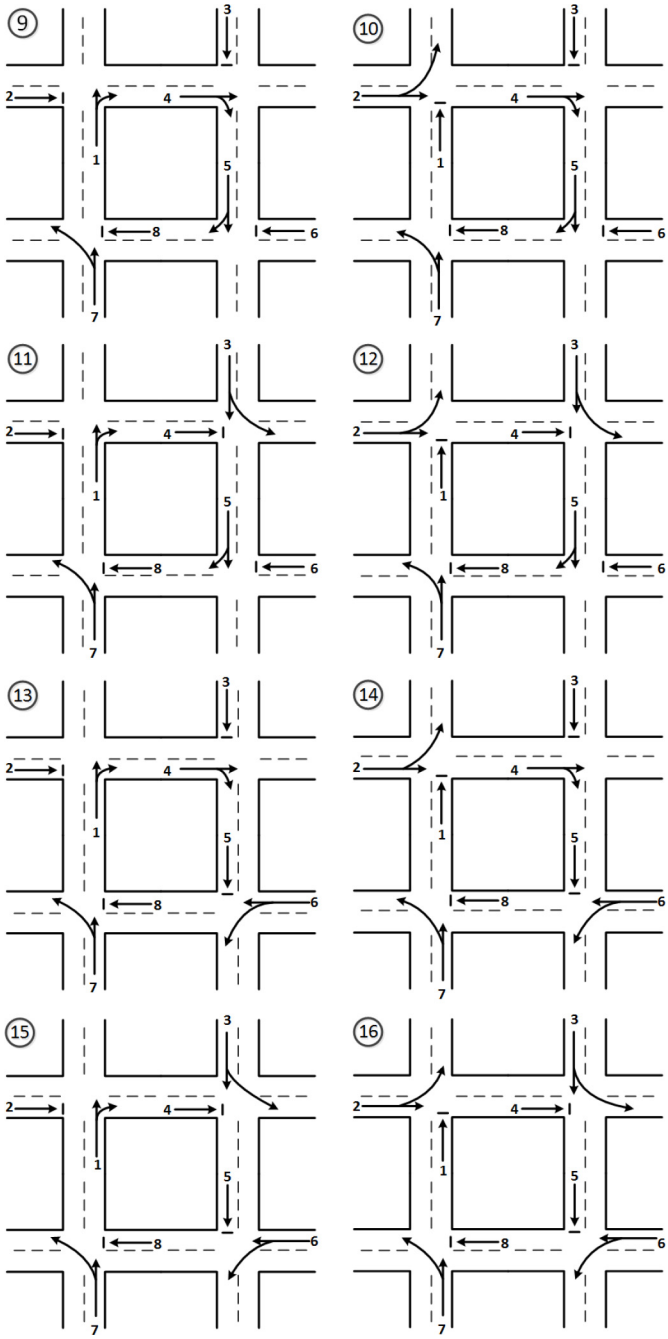

Fig. 7. Phases 9–16 for the Manhattan-like network with four intersections.

of performance; actually, the performance of the distributed approach is slightly better, which can be explained with the state dimension. In fact, because the full-state approximator must approximate a value function over a state space of large dimension (dimension 8), its approximation error might result bigger than multiple approximators working with a smaller state space (dimension 3). The most important result of the simulations is that the piecewise quadratic NN works better than the quadratic NN: this suggests that exploiting the structure of the traffic dynamics (taking into account the structural and nonlinear characteristics of the problem) helps to define new ADP heuristics in the form of the piecewise smooth neural network approximators leading to improved performance.

TABLE II
COST OF THE THREE APPROXIMATORS

Class of approximator	*Total cost (0-150)*	*Transient cost (0-45)*
Full states quadratic	20140	18953
Distributed quadratic	20031	18852
Piecewise quadratic	19921	18748

A final comment regards the steady-state sequence (i.e., the sequence achieved after the transient): from Figs. 3–5, it can be seen that this is sequence 3-5-10 for the full-state approximator, sequence 1-13-5 for the distributed quadratic approximator, and sequence 1-10-7 for the distributed piecewise quadratic approximator. From Figs. 6 and 7, it can be seen

TABLE III
STEADY-STATE COST OF THE THREE APPROXIMATORS

Class of approximator	*Steady-state cost per cycle*
Full states quadratic	11.2
Distributed quadratic	11.1
Smooth piecewise quadratic	11.1

that all these sequences allow the vehicles to circulate along the ring. In addition, Table III reveals that all the steady-state sequences have a very similar average cost, i.e., the sequences are almost equivalent. The true benefits of one approximator as compared to the other one come from the transient phase.

V. CONCLUSION

This paper proposed a novel model-based and switching-based framework for traffic signal operation. The framework used learning methods to seek the optimal traffic light policy, i.e., it can determine both when to switch and "which mode" to switch on when controlling the traffic lights operation. Minimum dwell-time constraints can be added to comply with a minimum green/red time in each phase. Compared with the Store-and-forward, BLX, and S-model models, the new model does not need to average the dynamics over one cycle time. This implies that the switching architecture can average the dynamics over one phase instead of one cycle, and different cycle times at different intersections can be handled without the need to impose synchronization constraints. We make use of the structure of the system to define new ADP heuristics (in the form of the piecewise smooth neural network approximators) that can taken into account some structural and nonlinear characteristics of the problem.

Relevant future work is to make the neural network training distributed, with the aim to overcome the curse of dimensionality arising from having exponential increasing phases. Another interesting future work could be to try the proposed methodology on a representative microscopic network created on simulation of urban mobility (SUMO), for example.

APPENDIX
MATRICES FOR THE MANHATTAN-LIKE NETWORK

$$B_1 = \big[\beta_{8,r}\mu_8 - (\beta_{1,r}+\beta_{1,s})\mu_1; \alpha_2^{in}; \alpha_3^{in}; \beta_{1,r}\mu_1 - (\beta_{4,r}+\beta_{4,s})\mu_4 \quad \beta_{4,r}\mu_4 - (\beta_{5,r}+\beta_{5,s})\mu_5; \alpha_6^{in}; \alpha_7^{in}; \beta_{5,r}\mu_5 - (\beta_{8,r}+\beta_{8,s})\mu_8\big]$$

$$B_2 = \big[\beta_{8,r}\mu_8; \alpha_2^{in} - (\beta_{2,r}+\beta_{2,s})\mu_2; \alpha_3^{in}; \beta_{2,s}\mu_2 - (\beta_{4,r}+\beta_{4,s})\mu_4 \quad \beta_{4,s}\mu_4 - (\beta_{5,r}+\beta_{5,s})\mu_5; \alpha_6^{in}; \alpha_7^{in}; \beta_{5,r}\mu_5 - (\beta_{8,r}+\beta_{8,s})\mu_8\big]$$

$$B_3 = \big[\beta_{8,r}\mu_8 - (\beta_{1,r}+\beta_{1,s})\mu_1; \alpha_2^{in}; \alpha_3^{in} - (\beta_{3,r}+\beta_{3,s})\mu_3; \beta_{1,r}\mu_1 \quad \beta_{3,s}\mu_3 - (\beta_{5,r}+\beta_{5,s})\mu_5; \alpha_6^{in}; \alpha_7^{in}; \beta_{5,r}\mu_5 - (\beta_{8,r}+\beta_{8,s})\mu_8\big]$$

$$B_4 = \big[\alpha_1^{in}; \alpha_2^{in} - (\beta_{2,r}+\beta_{2,s})\mu_2; \alpha_3^{in} - (\beta_{3,l}+\beta_{3,s})\mu_3; \beta_{2,s}\mu_2 \quad \beta_{3,s}\mu_3 - (\beta_{5,r}+\beta_{5,s})\mu_5; \alpha_6^{in}; \alpha_7^{in}; \beta_{5,r}\mu_5 - (\beta_{8,r}+\beta_{8,s})\mu_8\big]$$

$$B_5 = \big[\beta_{8,r}\mu_8 - (\beta_{1,r}+\beta_{1,s})\mu_1; \alpha_2^{in}; \alpha_3^{in}; \beta_{1,r}\mu_1 - (\beta_{4,r}+\beta_{4,s})\mu_4 \quad \beta_{4,r}\mu_4; \alpha_6^{in} - (\beta_{6,1}+\beta_{6,s})\mu_6; \alpha_7^{in}; \beta_{6,s}\mu_6 - (\beta_{8,r}+\beta_{8,s})\mu_8\big]$$

$$B_6 = \big[\beta_{8,r}\mu_8 - (\beta_{1,r}+\beta_{1,s})\mu_1; \alpha_2^{in} - (\beta_{2,l}+\beta_{2,s})\mu_2; \alpha_3^{in} \quad \beta_{2,s}\mu_2 - (\beta_{4,r}+\beta_{4,s})\mu_4; \beta_{4,r}\mu_4; \alpha_6^{in} - (\beta_{6,l}+\beta_{6,s})\mu_6; \alpha_7^{in} \quad \beta_{6,s}\mu_6 - (\beta_{8,r}+\beta_{8,s})\mu_8\big]$$

$$B_7 = \big[\beta_{8,r}\mu_8 - (\beta_{1,r}+\beta_{1,s})\mu_1; \alpha_2^{in}; \alpha_3^{in} - (\beta_{3,l}+\beta_{3,s})\mu_3; \beta_{1,r}\mu_1 \quad \beta_{3,s}\mu_3; \alpha_6^{in} - (\beta_{6,l}+\beta_{6,s})\mu_6; \alpha_7^{in}; \beta_{6,s}\mu_6 - (\beta_{8,r}+\beta_{8,s})\mu_8\big]$$

$$B_8 = \big[\beta_{8,r}\mu_8; \alpha_2^{in} - (\beta_{2,l}+\beta_{2,s})\mu_2; \alpha_3^{in} - (\beta_{3,l}+\beta_{3,s})\mu_3; \beta_{2,s}\mu_2 \quad \beta_{3,s}\mu_3; \alpha_6^{in} - (\beta_{6,1}+\beta_{6,s})\mu_6; \alpha_7^{in}; \beta_{6,s}\mu_6 - (\beta_{8,r}+\beta_{8,s})\mu_8\big]$$

$$B_9 = \big[\beta_{7,s}\mu_7 - (\beta_{1,r}+\beta_{1,s})\mu_1; \alpha_2^{in}; \alpha_3^{in}; \beta_{1,r}\mu_1 - (\beta_{4,r}+\beta_{4,s})\mu_4 \quad \beta_{4,r}\mu_4 - (\beta_{5,r}+\beta_{5,s})\mu_5; \alpha_6^{in}; \alpha_7^{in} - (\beta_{7,1}+\beta_{7,s})\mu_7; \beta_{5,r}\mu_5\big]$$

$$B_{10} = \big[\beta_{7,s}\mu_7; \alpha_2^{in} - (\beta_{2,l}+\beta_{2,s})\mu_2; \alpha_3^{in}; \beta_{2,s}\mu_2 - (\beta_{4,r}+\beta_{4,s})\mu_4 \quad \beta_{4,r}\mu_4 - (\beta_{5,r}+\beta_{5,s})\mu_5; \alpha_6^{in}; \alpha_7^{in} - (\beta_{7,1}+\beta_{7,s})\mu_7; \beta_{5,r}\mu_5\big]$$

$$B_{11} = \big[\beta_{7,s}\mu_7 - (\beta_{1,r}+\beta_{1,s})\mu_1; \alpha_2^{in}; \alpha_3^{in} - (\beta_{3,l}+\beta_{3,s})\mu_3; \beta_{1,r}\mu \quad \beta_{3,s}\mu_3 - (\beta_{5,r}+\beta_{5,s})\mu_5; \alpha_6^{in}; \alpha_7^{in} - (\beta_{7,1}+\beta_{7,s})\mu_7; \beta_{5,r}\mu_5\big]$$

$$B_{12} = \big[\beta_{7,s}\mu_7; \alpha_2^{in} - (\beta_{2,l}+\beta_{2,s})\mu_2; \alpha_3^{in} - (\beta_{3,l}+\beta_{3,s})\mu_3; \beta_{2,s}\mu_2 \quad \beta_{3,s}\mu_3 - (\beta_{5,r}+\beta_{5,s})\mu_5; \alpha_6^{in}; \alpha_7^{in} - (\beta_{7,1}+\beta_{7,s})\mu_7; \beta_{5,r}\mu_5\big]$$

$$B_{13} = \big[\beta_{7,s}\mu_7 - (\beta_{1,r}+\beta_{1,s})\mu_1; \alpha_2^{in}; \alpha_3^{in}; \beta_{1,r}\mu_1 - (\beta_{4,r}+\beta_{4,s})\mu_4 \quad \beta_{4,r}\mu_4; \alpha_6^{in} - (\beta_{6,l}+\beta_{6,s})\mu_6; \alpha_7^{in} - (\beta_{7,1}+\beta_{7,s})\mu_7; \beta_{6,s}\mu_6\big]$$

$$B_{14} = \big[\beta_{7,s}\mu_7; \alpha_2^{in} - (\beta_{2,r}+\beta_{2,s})\mu_1; \alpha_3^{in}; \beta_{2,s}\mu_2 - (\beta_{4,r}+\beta_{4,s})\mu_4 \quad \beta_{4,r}\mu_4; \alpha_6^{in} - (\beta_{6,l}+\beta_{6,s})\mu_6; \alpha_7^{in} - (\beta_{7,1}+\beta_{7,s})\mu_7; \beta_{6,s}\mu_6\big]$$

$$B_{15} = \big[\beta_{7,s}\mu_7 - (\beta_{1,r}+\beta_{1,s})\mu_1; \alpha_2^{in}; \alpha_3^{in} - (\beta_{3,l}+\beta_{3,s})\mu_3; \beta_{1,r}\mu_1 \quad \beta_{3,s}\mu_3; \alpha_6^{in} - (\beta_{6,l}+\beta_{6,s})\mu_6; \alpha_7^{in} - (\beta_{7,l}+\beta_{7,s})\mu_7; \beta_{6,s}\mu_6\big]$$

$$B_{16} = \big[\beta_{7,s}\mu_7; \alpha_2^{in} - (\beta_{2,l}+\beta_{2,s})\mu; \alpha_3^{in} - (\beta_{3,1}+\beta_{3,s})\mu_3; \beta_{2,s}\mu_2 \quad \beta_{3,s}\mu_3; \alpha_6^{in} - (\beta_{6,l}+\beta_{6,s})\mu_6; \alpha_7^{in} - (\beta_{7,1}+\beta_{7,s})\mu_7; \beta_{6,s}\mu_6\big].$$

REFERENCES

[1] L.-W. Chen and C. Chang, "Cooperative traffic control with green wave coordination for multiple intersections based on the Internet of Vehicles," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 7, pp. 1321–1335, Jul. 2017.

[2] W. Huang, Y. Wei, J. Guo, and J. Cao, "Next-generation innovation and development of intelligent transportation system in China," *Sci. China Inf. Sci.*, vol. 60, no. 11, 2017, Art. no. 110201.

[3] F. Ahmad, S. A. Mahmud, and F. Z. Yousaf, "Shortest processing time scheduling to reduce traffic congestion in dense urban areas," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 5, pp. 838–855, May 2017.

[4] D. Zhao, Y. Dai, and Z. Zhang, "Computational intelligence in urban traffic signal control: A survey," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 42, no. 4, pp. 485–494, Jul. 2012.

[5] Y. Wan, J. Cao, W. Huang, J. Guo, and Y. Wei, "Perimeter control of multiregion urban traffic networks with time-varying delays," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published.

[6] L. Li, D. Wen, and D. Yao, "A survey of traffic control with vehicular communications," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 1, pp. 425–432, Feb. 2014.

[7] Y. Zhang, M. Wang, X. Fang, and U. Ozguner, "Unifying analytical methods with numerical methods for traffic system modeling and control," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published.

[8] S. Baldi, I. Michailidis, E. B. Kosmatopoulos, A. Papachristodoulou, and P. A. Ioannou, "Convex design control for practical nonlinear systems," *IEEE Trans. Autom. Control*, vol. 59, no. 7, pp. 1692–1705, Jul. 2014.

[9] L. Guo, H. Chen, Q. Liu, and B. Gao, "A computationally efficient and hierarchical control strategy for velocity optimization of on-road vehicles," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 49, no. 1, pp. 31–41, Jan. 2019.

[10] A. D. Febbraro, D. Giglio, and N. Sacco, "A deterministic and stochastic petri net model for traffic-responsive signaling control in urban areas," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 2, pp. 510–524, Feb. 2016.

[11] S. Jin, Z. Hou, R. Chi, and X. Bu, "Model free adaptive predictive control approach for phase splits of urban traffic network," in *Proc. Chin. Control Decis. Conf. (CCDC)*, Yinchuan, China, 2016, pp. 5750–5754.

[12] M. A. Khamis and W. Gomaa, "Adaptive multi-objective reinforcement learning with hybrid exploration for traffic signal control based on cooperative multi-agent framework," *Eng. Appl. Artif. Intell.*, vol. 29, pp. 134–151, Mar. 2014.

[13] L. Li, Y. Lv, and F.-Y. Wang, "Traffic signal timing via deep reinforcement learning," *IEEE/CAA J. Automatica Sinica*, vol. 3, no. 3, pp. 247–254, Jul. 2016.
[14] X. Liang, X. Du, G. Wang, and Z. Han, "Deep reinforcement learning for traffic light control in vehicular networks," *arXiv:1803.11115*, 2018.
[15] K. Huang, Q. Zhang, C. Zhou, N. Xiong, and Y. Qin, "An efficient intrusion detection approach for visual sensor networks based on traffic pattern learning," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 10, pp. 2704–2713, Oct. 2017.
[16] P. Mannion, J. Duggan, and E. Howley, *An Experimental Review of Reinforcement Learning Algorithms for Adaptive Traffic Signal Control*. Cham, Switzerland: Springer Int., 2016, pp. 47–66.
[17] X. Xu, L. Zuo, X. Li, L. Qian, J. Ren, and Z. Sun, "A reinforcement learning approach to autonomous decision making of intelligent vehicles on highways," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published.
[18] Y. A. Harfouch, S. Yuan, and S. Baldi, "An adaptive switched control approach to heterogeneous platooning with intervehicle communication losses," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 3, pp. 1434–1444, Sep. 2018.
[19] S. Yang, W. Wang, C. Liu, and W. Deng, "Scene understanding in deep learning-based end-to-end controllers for autonomous vehicles," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 49, no. 1, pp. 53–63, Jan. 2019.
[20] M. Tahifa, J. Boumhidi, and A. Yahyaouy, "Swarm reinforcement learning for traffic signal control based on cooperative multi-agent framework," in *Proc. Intell. Syst. Comput. Vis. (ISCV)*, 2015, pp. 1–6.
[21] L. A. Prashanth and S. Bhatnagar, "Reinforcement learning with function approximation for traffic signal control," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 2, pp. 412–421, Jun. 2011.
[22] Y. Kim, T. Kato, S. Okuma, and T. Narikiyo, "Traffic network control based on hybrid dynamical system modeling and mixed integer nonlinear programming with convexity analysis," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 38, no. 2, pp. 346–357, Mar. 2008.
[23] M. V. den Berg, A. Hegyi, B. De Schutter, and J. Hellendoorn, "A macroscopic traffic flow model for integrated control of freeway and urban traffic networks," in *Proc. 42nd IEEE Int. Conf. Decis. Control*, vol. 3, 2003, pp. 2774–2779.
[24] E. Azimirad, N. Pariz, and M. B. N. Sistani, "A novel fuzzy model and control of single intersection at urban traffic network," *IEEE Syst. J.*, vol. 4, no. 1, pp. 107–111, Mar. 2010.
[25] E. Camponogara and L. B. de Oliveira, "Distributed optimization for model predictive control of linear-dynamic networks," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 39, no. 6, pp. 1331–1338, Nov. 2009.
[26] K. Aboudolas, M. Papageorgiou, and E. Kosmatopoulos, "Store-and-forward based methods for the signal control problem in large-scale congested urban road networks," *Transport. Res. C Emerg. Technol.*, vol. 17, no. 2, pp. 163–174, 2009.
[27] 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," *Transport. Res. C Emerg. Technol.*, vol. 18, no. 5, pp. 680–694, 2010.
[28] S. Lin, B. De Schutter, Y. Xi, and H. Hellendoorn, "An efficient model-based method for coordinated control of urban traffic networks," in *Proc. Int. Conf. Netw. Sens. Control (ICNSC)*, Chicago, IL, USA, 2010, pp. 8–13.
[29] Z. Zhou, B. De Schutter, S. Lin, and Y. Xi, "Two-level hierarchical model-based predictive control for large-scale urban traffic networks," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 2, pp. 496–508, Mar. 2017.
[30] S. Lin, B. De Schutter, Y. Xi, and H. Hellendoorn, "Fast model predictive control for urban road networks via MILP," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 3, pp. 846–856, Sep. 2011.
[31] S. Lin, B. De Schutter, Y. Xi, and H. Hellendoorn, "Efficient network-wide model-based predictive control for urban traffic networks," *Transport. Res. C Emerg. Technol.*, vol. 24, pp. 122–140, Oct. 2012.
[32] A. Hegyi, B. De Schutter, and J. Hellendoorn, "Optimal coordination of variable speed limits to suppress shock waves," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 1, pp. 102–112, Mar. 2005.
[33] L. Iannelli, K. H. Johansson, U. T. Jönsson, and F. Vasca, "Subtleties in the averaging of a class of hybrid systems with applications to power converters," *Control Eng. Pract.*, vol. 16, no. 8, pp. 961–975, 2008.
[34] L. I. Allerhand and U. Shaked, "Robust stability and stabilization of linear switched systems with dwell time," *IEEE Trans. Autom. Control*, vol. 56, no. 2, pp. 381–386, Feb. 2011.
[35] M. Jungers and J. Daafouz, "Guaranteed cost certification for discrete-time linear switched systems with a dwell time," *IEEE Trans. Autom. Control*, vol. 58, no. 3, pp. 768–772, Mar. 2013.
[36] C. Yuan and F. Wu, "Hybrid control for switched linear systems with average dwell time," *IEEE Trans. Autom. Control*, vol. 60, no. 1, pp. 240–245, Jan. 2015.
[37] A. Heydari, "Optimal switching with minimum dwell time constraint," *J. Frankl. Inst.*, vol. 354, no. 11, pp. 4498–4518, 2017.
[38] A. Heydari and S. Balakrishnan, "Optimal switching between autonomous subsystems," *J. Frankl. Inst.*, vol. 351, no. 5, pp. 2675–2690, 2014.
[39] N. Farhi, C. N. V. Phu, M. Amir, H. Haj-Salem, and J.-P. Lebacque, "A semi-decentralized control strategy for urban traffic," *Transport. Res. Procedia*, vol. 10, pp. 41–50, Jul. 2015.
[40] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *IEEE Circuits Syst. Mag.*, vol. 9, no. 3, pp. 32–50, Aug. 2009.
[41] C. Qin, H. Zhang, Y. Luo, and B. Wang, "Finite horizon optimal control of non-linear discrete-time switched systems using adaptive dynamic programming with ϵ-error bound," *Int. J. Syst. Sci.*, vol. 45, no. 8, pp. 1683–1693, 2014.
[42] R. Jiang *et al.*, "Network operation reliability in a manhattan-like urban system with adaptive traffic lights," *Transport. Res. C Emerg. Technol.*, vol. 69, pp. 527–547, Aug. 2016.
[43] M. D. Simoni and C. G. Claudel, "A simulation framework for modeling urban freight operations impacts on traffic networks," *Simulat. Model. Pract. Theory*, vol. 86, pp. 36–54, Aug. 2018.
[44] E. Thonhofer, T. Palau, A. Kuhn, S. Jakubek, and M. Kozek, "Macroscopic traffic model for large scale urban traffic network design," *Simulat. Model. Pract. Theory*, vol. 80, pp. 32–49, Jan. 2018.

Di Liu received the B.Sc. degree in electronic information science and technology from the Hebei University of Science and Technology, Shijiazhuang, China, in 2014, and the M.Sc. degree in control science and engineering from the Chongqing University of Posts and Telecommunications, Chongqing, China, in 2017. She is currently pursuing the Ph.D. degree in cyber engineering with the School of Cyber Science and Engineering, Southeast University, Nanjing, China.

Her current research interests include adaptive and learning systems and control, with application in intelligent transportation and automatic vehicles.

Wenwu Yu (S'07–M'12–SM'15) received the B.Sc. degree in information and computing science and the M.Sc. degree in applied mathematics from the Department of Mathematics, Southeast University, Nanjing, China, in 2004 and 2007, respectively, and the Ph.D. degree in electronic engineering from the Department of Electronic Engineering, City University of Hong Kong, Hong Kong, in 2010.

He is currently the Founding Director of the Laboratory of Cooperative Control of Complex Systems and the Deputy Associate Director of the Jiangsu Provincial Key Laboratory of Networked Collective Intelligence, an Associate Dean with the School of Mathematics, and a Full Professor with the Young Endowed Chair Honor, Southeast University. He held several visiting positions in Australia, China, Germany, Italy, The Netherlands, and the USA. He has published about 100 SCI journal papers with over 10 000 citations. His current research interests include multiagent systems, complex networks and systems, disturbance control, distributed optimization, neural networks, game theory, cyberspace security, smart grids, intelligent transportation systems, and big-data analysis.

Prof. Yu was a recipient of the Highly Cited Researchers Award in Engineering by Clarivate Analytics/Thomson Reuters in 2014–2018, the National Natural Science Fund for Excellent Young Scholars in 2013, the National Ten Thousand Talent Program for Young Top-Notch Talents in 2014, the Cheung Kong Scholars Programme of China for Young Scholars in 2016, and the Second Prize of State Natural Science Award of China in 2016. He severs as an Editorial Board Member of several flag journals, including the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—PART II: EXPRESS BRIEFS, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, *Science China Information Sciences*, and *Science China Technological Sciences*.

Simone Baldi received the B.Sc. degree in electrical engineering, and the M.Sc. and Ph.D. degrees in automatic control systems engineering from the University of Florence, Florence, Italy, in 2005, 2007, and 2011, respectively.

He is currently a Professor with the School of Mathematics, Southeast University, Nanjing, China, with a guest position with the Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands, where he was an Assistant Professor. He was a Post-Doctoral Researcher with the University of Cyprus, Nicosia, Cyprus, and the Information Technologies Institute, Centre for Research and Technology Hellas, Thessaloniki, Greece. His current research interests include adaptive and learning systems with applications in networked control systems, smart energy, and intelligent vehicle systems.

Prof. Baldi was a recipient of the Outstanding Reviewer Award of *Applied Energy* in 2016, *Automatica* in 2017, and *IET Control Theory and Applications* in 2018. Since March 2019, he has been a Subject Editor of the *International Journal of Adaptive Control and Signal Processing*.

Jinde Cao (M'07–SM'07–F'16) received the B.S. degree in mathematics/applied mathematics from Anhui Normal University, Wuhu, China, in 1986, the M.S. degree in mathematics/applied mathematics from Yunnan University, Kunming, China, in 1989, and the Ph.D. degree in mathematics/applied mathematics from Sichuan University, Chengdu, China, in 1998.

He is an Endowed Chair Professor and the Dean of the School of Mathematics, the Director of the Jiangsu Provincial Key Laboratory of Networked Collective Intelligence of China, and the Director of the Research Center for Complex Systems and Network Sciences, Southeast University, Nanjing, China.

Prof. Cao was a recipient of the National Innovation Award of China, the Obada Prize, and the Highly Cited Researcher Award in Engineering, Computer Science, and Mathematics by Thomson Reuters/Clarivate Analytics. He is a member of the Academy of Europe and the European Academy of Sciences and Arts, a fellow of Pakistan Academy of Sciences, and an IASCYS Academician.

Wei Huang received the B.S., M.S., and Ph.D. degrees in road engineering from Southeast University, Nanjing, China, in 1982, 1986, and 1995, respectively.

He is currently a Distinguished Professor in Civil Engineering with the Intelligent Transportation System Research Center, Southeast University. He is a member of the Chinese Academy of Engineering, Beijing, China. He enjoys the State Council special allowance and receives support from the New Century Talent Program, the National Outstanding Mid-Aged Experts Program, the National Talents Engineering Program, and the Yangtze Scholar Program from various agencies and organizations. He is one of the forerunners in the research fields of long-span steel bridge pavement and intelligent transportation systems of China. He has authored or coauthored 13 books.

Dr. Huang was a recipient of 26 awards from both the national and provincial level, as the leading awardee.