Optimizing Signal Timing Control for Large Urban Traffic Networks Using an Adaptive Linear Quadratic Regulator Control Strategy

Hong Wang, *Fellow, IET*, Meixin Zhu, Wanshi Hong, Chieh (Ross) Wang, *Member, IEEE*, Gang Tao, *Fellow, IEEE*, and Yinhai Wang, *Senior Member, IEEE*

***Abstract*—Traffic signal control is important for intersection safety and efficiency. However, most traffic signal control methods are designed for individual intersections or corridors. Although some adaptive control systems have been developed, the methods used are often proprietary and not published, making it difficult to evaluate their effectiveness. This study proposes an adaptive multi-input and multi-output traffic signal control method that can not only improve network-wide traffic operations in terms of reduced traffic delay and energy consumption, but also is more computationally feasible than existing centralized signal control methods. Considering intersection interactions, a linear dynamic traffic system model was built and adaptively updated to reflect how the signal control input of each intersection affects network-wide vehicle travel delay. Based on the system model, an adaptive linear-quadratic regulator (LQR) was designed to minimize both traffic delay and incremental changes in the control input. The proposed control method was evaluated in a microscopic traffic simulation environment with a 35-intersection network of Bellevue City, Washington. Simulation results show that the proposed method had shorter average traffic delays in the network when compared with the traffic delays controlled by the state-of-the-art max-pressure, self-organizing traffic lights, and independent deep Q network methods.**

***Index Terms*—Urban traffic network, traffic signal, linear quadratic regulator (LQR) control, multi-input and multi-output (MIMO) system.**

I. Introduction

TRAFFIC congestion caused by intersections leads to unnecessary travel delays, reduced traffic safety, and increased energy consumption and environmental pollution [1, 2]. To relieve traffic network congestion in large cities, efficient traffic signal control methods have always been highly demanded, especially with the rapid advances in communication and computing technologies [3, 4].

H. Wang, M. Zhu, W. Hong, and C. Wang are with the Energy and Transportation Science Division, Oak Ridge National Laboratory Oak Ridge, TN 37831 USA (e-mail: wangh6@ornl.gov; zhum@ornl.gov; hongw@ornl.gov; cwang@ornl.gov).

G. Tao is with the Department of Electrical and Computer Engineering, University of Virginia, Charlottesville, VA 22904 USA (e-mail: gt9s@virginia.edu).

Y. Wang is with the Department of Civil and Environmental Engineering, University of Washington, Seattle, WA 98195 USA (e-mail: yinhai@uw.edu).

Manuscript received December 20, 2019.

Traffic signal control includes four primary components: phase specification (sequence of the phases), signal timing/split control (relative green duration of each phase), cycle duration, and offset of cycles for coordination among multiple intersections that are spatially close. Among these components, signal timing/split control can have the most profound impact on traffic operations [3, 5]; hence, this study investigated this research topic.

The early investigation of urban traffic signal control dates back to the 20th century when traffic lights were first invented. Since then, a variety of signal timing control methods have been developed. Generally, these methods can be divided into two categories: pretimed and traffic-responsive [1]. Pretimed methods adopt constant green time splits based on historical traffic demands over the considered signalized urban area. One of the representative pretimed traffic signal control systems is TRANSYT (TRAffic Network StudY Tool) [6], which tries to minimize the overall travel time, delays, and number of stops. With a fixed green time duration, nevertheless, pretimed control methods may not be able to handle the dynamics of real-time traffic conditions.

Aiming at addressing the limitation of pretimed control, traffic-responsive methods optimize signal timing based on real-time traffic data. The most common traffic-responsive signal control method is actuated control. It uses sensors (e.g., inductive loop detectors) located upstream of the stop line to sense the request of green time. The method also predefines minimum and maximum green times and passage time. On the basis of minimum green time, actuated control extends the green time by the amount of passage time once a vehicle is detected until the maximum green time is reached [1].

Although actuated control is responsive to traffic dynamics to some extent, it is ideally suited for isolated intersections in which traffic demands and patterns vary widely throughout the day [7]. To improve traffic efficiency at a network-wide level, plenty of emerging traffic signal control methods, such as evolutionary algorithms [8, 9], heuristics approaches (e.g., max pressure [10] and self-organizing traffic light controls (SOTL) [11]), fuzzy logic control [12, 13], neural networks control [14, 15], reinforcement learning based control [16, 17], and deep reinforcement learning control [18, 19, 20, 21], have been proposed. To consider multiple evaluation metrics (e.g., traffic throughput, traffic delay, safety, and energy consumption), multi-objective optimization algorithms have also been applied in traffic signal controls to optimize the system performance

[4, 22, 23]. However, they are not widely deployed in real-world intersections because of their complex or black-box logics, high performance variances, and special hardware requirements [24].

For network-level traffic signal control, SCATS (Sydney Coordinated Adaptive Traffic System) [25], SCOOT (Split Cycle Offset Optimization Technique) [26], OPAC (Optimization Policies for Adaptive Control) [27], and TUC (Traffic-responsive Urban Control) [28, 29] are the most widely used traffic-responsive signal control systems [1, 3]. Both SCOOT and OPAC incorporate a network model that uses real-time traffic data as an input and computes the corresponding traffic network performance indices, such as the total number of vehicle stops. The primary difference between SCOOT and OPAC is that the former uses heuristic rules whereas the latter uses dynamic programming. For SCOOT, the central control computer repeatedly runs the network model and determines the effect of incremental changes of splits, offsets, and cycle lengths at individual intersections. The changes will be submitted to the local signal controllers if they are predicted to be beneficial in terms of performance indices. OPAC assumes the phase switching time as a discrete variable and calculates the optimal switching time with dynamic optimization [1]. TUC adopts a store-and-forward modeling of the urban network traffic and uses the linear-quadratic regulator (LQR) theory [30]. The design of TUC results in a multivariable regulator for traffic-responsive coordinated network-wide signal control [29].

Although widely used, the aforementioned traffic-responsive control systems have several limitations and challenges:

- **Relying on off-line traffic-network models:** Although using real-time traffic data as an input, the network models used by SCOOT and OPAC are built based on historical data and are not updated during their real-time operations. Because historical data may not accurately reflect current traffic conditions [12], these control strategies may potentially result in poor performance.
- **Exponential complexity for global minimization:** Similar to OPAC, because of the presence of discrete variables that require exponential-complexity algorithms for a global minimization, the control strategies may not be real-time feasible for large-scale traffic networks [31].
- **Ignoring interactions between intersections:** In TUC, traffic systems are modeled at intersection level, and the impacts of the traffic states in neighborhood intersection and control inputs on the travel delays of investigated intersection are ignored.

To address these limitations, this study proposes a new multi-input and multi-output (MIMO) traffic signal control method that not only improves network-wide traffic operations in terms of reduced delay and energy consumption, but also is more computationally feasible than existing centralized signal control methods. Considering intersection interactions, a linear dynamic traffic system model was built and updated online to reflect how signal control inputs at each intersection would affect network-wide vehicle delays. Based on the system model, an LQR was built to minimize both traffic delay and control-input changes [32]. We select LQR because it has the following advantages over other control strategies from the control design perspectives: 1) LQR is a simple optimal control that can be made adaptive in combination with system parameter estimation; 2) LQR is robust with respect to model uncertainties and errors as well as unexpected disturbances; and 3) LQR is easy to be implemented as a MIMO control strategy that produces a feedback control for a large networked intersection while automatically accounting for intersection interactions. The proposed adaptive LQR traffic control algorithm does not rely on large historic datasets to identify the unknown traffic model. Moreover, using LQR control with a MIMO linearized model can achieve minimal traffic delay while considering neighboring intersection interactions. The proposed control method was implemented and evaluated in a microscopic traffic simulation environment with a 35-intersection network of Bellevue City, Washington. Simulation results show that the proposed method had shorter average travel delays in the network when compared with the delays controlled by the methods proposed in the state-of-the-art max-pressure [10], self-organizing traffic lights (SOTL, [11]), and independent deep Q network (IDQN) control [33].

Major contributions of this paper include:

- Proposed a globalized modeling framework for network-wide traffic signal control using calibrated VISSIM model via real traffic flow data. In this framework, intersection interactions were explicitly modeled by a MIMO linear time-varying traffic system model that reflects how the signal control input at each intersection affects network-wide vehicle delay measurements.
- Established an adaptive LQR based traffic signal control method for traffic networks. The method can identify unknown system dynamics online, thus being able to handle system uncertainties caused by traffic and environmental randomness.
- Cross-compared the performances of several traffic signal control methods based on a real-world data-based microscopic traffic simulation model.

The remainder of this paper is organized as follows: Section II introduces the urban traffic network from downtown Bellevue, Washington and its simulation model in VISSIM, which is calibrated using real traffic flow data. Section III presents the adaptive LQR traffic signal control design. Section IV describes the traffic signal control methods to be compared with our design. Section V shows the results of a simulation study that evaluates and compares the performance of different traffic signal control designs with the proposed method. Section VI draws conclusions and discusses future works.

II. Urban Traffic Network and Simulation Model

A. *Traffic Network*

This study focuses on urban road networks with signalized intersections. Specifically, a grid road network from downtown Bellevue, Washington was selected as the study area. This study area covered from Main Street (the south end) to NE 12th Street (the north end) and from Bellevue Way NE (the west end) to 112th Ave NE (the east end). It included 35 intersections and 57 major bi-directional road links, with the average link length being 664.4 ft.

To replicate real-world traffic conditions, traffic count data by movement were collected for each intersection in the midday off-peak period (i.e., 1-2 p.m.). Fig. 1a shows the traffic counts of the northwest corner intersection (i.e., NE 12th Street and Bellevue Way NE) of the study area as an example. Link traffic volumes were calculated by aggregating traffic movement counts in the same direction as shown in Fig. 1b [34].

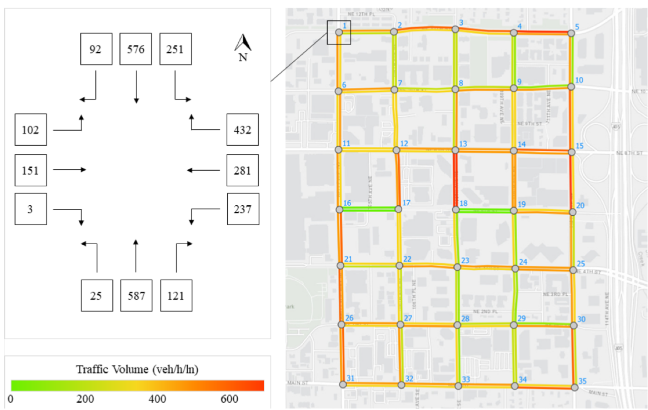


Fig. 1: Illustration of (a) traffic count by movement data (left) and (b) traffic volume data in the road network [34].

B. Microscopic Traffic Simulation Model

In this study, PTV VISSIM [35], a commonly used microscopic software for traffic simulation and signal controls, was used to facilitate the development and testing of different traffic signal control methods. VISSIM uses Wiedemann car-following and lane-changing models [36, 37] to model the movements and interactions of vehicles. The VISSIM traffic model shown in Fig. 2 was developed based on actual road geometries of the study area. This microscopic simulation model has been calibrated by the City of Bellevue with actual traffic data and has been used for planning and management purposes in downtown Bellevue [34]. The calibration process involved adjusting parameters including vehicle composition, speed distributions, conflict areas, priority rules, reduced speed areas, and car-following and lane-change behaviors to ensure that the simulation was consistent with the traffic characteristics of the real world.

III. Traffic Signal Controller Design

A. Traffic System Modeling

The subject traffic network comprises 35 intersections, and the green time for each intersection could affect the traffic flow performance of the whole system. In this study, we assume a fixed signal cycle length (90 s) with two phases: the east and west (E-W) approaches share one phase, and the north and south (N-S) approaches share the other phase so as to simplify the control algorithm formulation. As shown in Fig. 3, the N-S direction green time of intersection i is denoted as v_i with $i = 1, 2, ..., 35$. In practice, both the signal cycle length and the initial green time can be derived based on the widely used Webster's method [38], which considers both the saturation flow rate and the flow ratio for each lane group. It is assumed that each intersection had two delay measurements: one for the N-S direction and the other for the E-W direction. Therefore, there are a total of 70 delay measurements for the 35 intersections. Mathematically, the traffic network described in Section II-A can be expressed as a discrete-time input-output and real-data calibrated model in VISSIM [35] as:

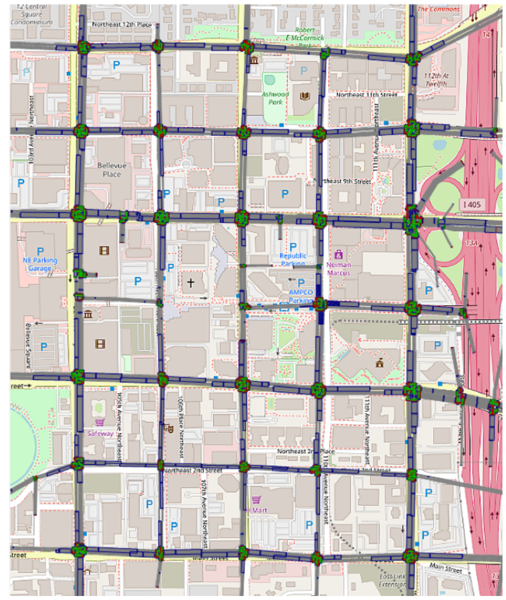


Fig. 2: Illustration of the VISSIM simulation model for the investigated urban traffic network [34].

$$z(k+1) = F(z(k), v(k)) \tag{1}$$

$$z(k) = [z_1, z_2, ..., z_{70}]^T \tag{2}$$

where $z(k) \in R^{70}$ are the system outputs, which are the measurable traffic delays of N-S and E-W direction vehicle flows at each node; $v \in [v_{min}, v_{max}] \in R^{35}$ are the system control inputs comprised of all v_i, indicating the N-S direction green times of the intersections; and k denotes the time step (i.e., the cycle number). F stands for the nonlinear relationship between traffic delays and the green signal period of the N-S direction as represented by VISSIM simulation [35].

Assuming that the nonlinearity of the traffic system is linearizable, one can linearize the system (1) to the state-space form as

$$\Delta z(k+1) = A\Delta z(k) + B\Delta v(k) + w(k), \tag{3}$$

where

$$A = \begin{bmatrix} a_{1,1} & \dots & a_{1,70} \\ \vdots & \ddots & \\ a_{70,1} & \dots & a_{70,70} \end{bmatrix} \in R^{70\times 70} \tag{4}$$

$$B = \begin{bmatrix} b_{1,1} & \dots & b_{1,35} \\ \vdots & \ddots & \\ b_{70,1} & \dots & b_{70,35} \end{bmatrix} \in R^{70\times 35}, \tag{5}$$

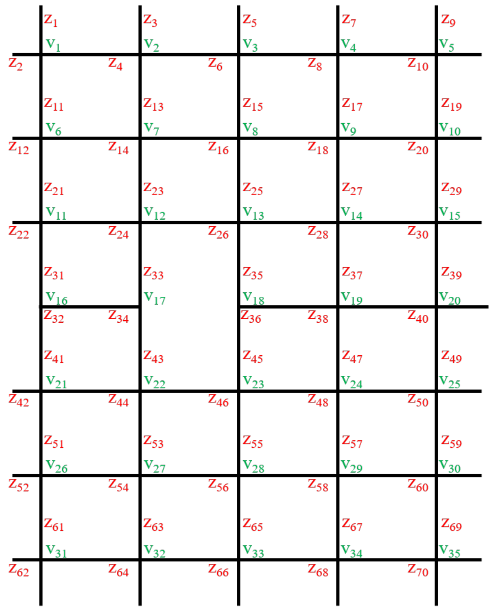


Fig. 3: Illustration of the notations for signal control inputs and traffic delay measurements.

are the system parameters, and $\Delta z(k) = z(k) - z(k-1)$, $\Delta v(k) = v(k) - v(k-1)$ are the increment of z and v at time step k. $w(k)$ is the bounded linearization error, which satisfies $||w(k)||_2 \leq w_b$ for $k = 1, 2, \ldots$, and w_b is the bound for the error $w(k)$. Based upon our comprehensive linearization around all the possible operating points of model (1), the value of $w_b = 4.5$ s in terms of travel delay. This will be used as a threshold in the normalized least squares algorithm to be described for estimating A and B matrices. To simplify notation, we denote $\Delta z = y$ and $\Delta v = u$, and rewrite the linearized state-space form as

$$y(k+1) = Ay(k) + Bu(k) + w(k). \tag{6}$$

Because the direct mathematical relationship between traffic delay and green signal period is unknown, the linearized parameter matrices A and B are unknown and possibly time-varying, and we need to design a parameter estimation scheme to estimate the system parameter matrices A and B online.

Control Objectives. The control objective is to design an online estimation based LQR controller with unknown system parameter matrices A and B to minimize the traffic delay characterized by $y(k)$ with low control energy. This constitutes the minimization of the following cost function by selecting an optimal signal timing strategy $u(k)$:

$$J = \sum_{k=0}^{\infty} [y^T(k)Qy(k) + u^T(k)Ru(k)], \tag{7}$$

where $Q = Q^T > 0$ and $R = R^T > 0$ are positive definite matrices with relevant dimensions. They denote the pre-specified state-cost weighting matrix and input-cost weighting matrix, respectively. This is the standard LQR cost function widely used in the literature [39].

B. Online Parameter Estimation Scheme

Before starting the LQR controller design, one needs to identify the system parameter matrices A and B using system identification techniques [40, 41, 42, 43]. To this end, the linearized system (6) can be parameterized as follows:

$$y(k+1) = \Theta\Phi(k) + w(k), \tag{8}$$

where

$$\Theta = \begin{bmatrix} A & B \end{bmatrix} \tag{9}$$

$$= \begin{bmatrix} a_{1,1} & \ldots & a_{1,70} & b_{1,1} & \ldots & b_{1,35} \\ \vdots & \ddots & & \vdots & \ddots & \\ a_{70,1} & \ldots & a_{70,70} & b_{70,1} & \ldots & b_{70,35} \end{bmatrix} \in R^{70\times 105} \tag{10}$$

$$\Phi = \begin{bmatrix} y^T & u^T \end{bmatrix}^T \tag{11}$$

$$= \begin{bmatrix} y_1 & y_2 & \ldots & y_{70} & u_1 & u_2 & \ldots & u_{35} \end{bmatrix}^T \in R^{105}, \tag{12}$$

where Φ groups all the measurable inputs and outputs and is the information vector used in the estimation of Θ.

Estimation Error. To estimate the system parameter matrices A and B, we denote $\Theta(k) \in R^{70\times 105}$ as the estimate of Θ at cycle k. Then the estimation error is defined as

$$\varepsilon(k) = \Theta(k-1)\Phi(k-1) - y(k) \tag{13}$$

$$= \Theta(k-1)\Phi(k-1) - \Theta\Phi(k-1) - w(k-1) \tag{14}$$

In this work, we assume that the bound for the linearization error term $w(k)$ is small, and the effect of the term $w(k)$ can be neglected in the control design due to the inherent strong robustness property of LQR control strategy and the use of the normalized least squares estimation algorithm [44].

Parameter Update Law. As it has been shown that the normalized least squares identification is robust to possible modeling uncertainties, such as the linearization errors in (6) [44], it is used here to provide the required estimates for parameter matrices A and B. The estimation updating rule for $k = 0, 1, 2, ...$ is therefore given by:

$$\Theta(k+1) = \begin{cases} \Theta(k) - \frac{P(k-1)\Phi(k)\varepsilon(k)}{m^2(k)} & ||\varepsilon(k)|| > w_b \\ \Theta(k) & ||\varepsilon(k)|| \leq w_b \end{cases} \tag{15}$$

$$P(k) = P(k-1) - \frac{P(k-1)\Phi(k)\Phi^T(k)P(k-1)}{m^2(k)} \tag{16}$$

$$m(k) = \sqrt{\kappa + \Phi^T(k)P(k-1)\Phi(k)}, \tag{17}$$

where $\kappa > 0$ is the pre-specified design parameter normally less than 0.05. $P(k)$ is a positive definite variance matrix and its initial value is $P(0) = I$ with I being an identity matrix. $\Theta(0) = \Theta_0$ is the chosen initial estimates of parameter matrices A and B. The above parameter estimation can be regarded as an online learning process where A and B are learned using input and output data grouped in Φ when the cycle number k increases.

It can be seen that (15) has a switching function, when the error $\epsilon(k)$ is smaller than or equal to the lower bound of $w(k)$, the parameter updating is stopped and the $k+1$ cycle estimate is its estimate at k. This functionality enhances the robustness of the estimation algorithm so as to make it robust with respect to linearization error $w(k)$ [44].

C. LQR Controller Design

When system parameter matrices A and B are known, one can choose the following LQR state feedback control law as

$$u(k) = -Ky(k), \tag{18}$$

where the state feedback gain matrix K is obtained by solving the following Riccati equation for matrices $S = S^T > 0$ and K simultaneously [39]:

$$A^T SA - S - A^T SBK + Q = 0 \tag{19}$$
$$K = (B^T SB + R)^{-1} B^T SA. \tag{20}$$

Adaptive LQR Controller. Because system parameter matrices A and B are estimated according to the parameter update law (15) – (17), one can obtain $A(k)$ and $B(k)$ from $\Theta(k) = [A(k)\ B(k)]$ and (15) at each cycle k. Therefore, the adaptive LQR controller can be obtained by replacing A and B in (19) and (20) with their estimates at each cycle k. This leads to the following adaptive LQR control law

$$u(k) = -K(k)y(k) \tag{21}$$

where adaptive gain matrix $K(k)$ is obtained by solving the Riccati equation for matrices $S(k) = S^T(k) > 0$ and $K(k)$ at each cycle k as follows:

$$0 = A^T(k)S(k)A(k) - S(k) - A^T(k)S(k)B(k)K(k) + Q \tag{22}$$
$$K(k) = (B^T(k)S(k)B(k) + R)^{-1} B^T(k)S(k)A(k). \tag{23}$$

D. Adaptive LQR Algorithm Summary

With the real-data calibrated system model (1), the above adaptive LQR control algorithm is realized in the following steps:

1) Calibrate VISSIM model (1) using real traffic data so that system model (1) gives a desired reflection of the actual system dynamics;
2) At time k, use input and output data from VISSIM nonlinear system (1) to estimate parameter matrices $\{A(k), B(k)\}$ by (15) – (17);
3) For the estimated $\{A(k), B(k)\}$, solve the Riccati equation (22) – (23) for adaptive control gain $K(k)$;
4) Apply control input (21) to the signal timing to all intersections in Fig. 1 and 3 as represented by system nonlinear model (1); and
5) Set $k = k + 1$ and go to Step 2.

Fig. 4 shows the closed-loop control structure and the information flow chart of the above adaptive LQR control algorithm.

The above algorithm shows that as travel delay state vector $y(k)$ is assumed measurable, it is used as a state feedback information in the construction of the closed loop control.

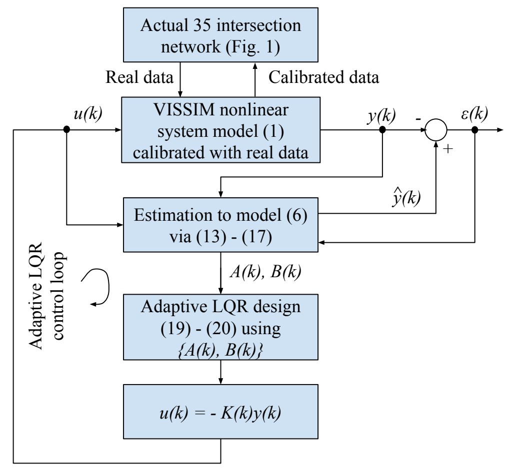


Fig. 4: The adaptive LQR closed loop control structure and the information flow chart.

E. Robustness of the Proposed Algorithm

It is worth noting that the control signal calculated from (21) is directly applied to the nonlinear system (1) which had been calibrated using real-traffic flow data, rather than to linearized system (6). The simulation results in the following sections would show the desired robustness of the proposed algorithm with respect to the linearization error $w(k)$ in (6). Indeed, as it has been shown that LQR has a good robustness with respect to model uncertainties in terms of providing infinite gain margin [39], the proposed control algorithm is also robust with regard to the model uncertainties. In addition, the normalized least squares with switching functionality as shown in (15) is robust with respect to modelling error $w(k)$ of the system. As a result, the proposed algorithm is robust because of the use of LQR. together with the normalized least squares algorithm for the estimation of parameter matrices $\{A(k), B(k)\}$.

Moreover, although adding more signal phases at each intersection could increase the complexity of dynamic signal timing updates in VISSIM, which is still programmatically feasible, it will not constitute difficulties in the control design as this would just increase the dimensionality of input $u(k)$ and $y(k)$ in (6), while the control design methods remains the same as those in (21) – (23).

F. Coordination and Scalability Issues

Since the proposed adaptive LQR control is obtained using the MIMO model in (6), it is a multivariable control that automatically takes into account of the interactions among all the intersections as the off-diagonal elements in matrix A are normally non-zeros, where the coupling feature among intersections are taken care of when control input vector (21) is applied to the system (1). This is the advantage of using multivariable control to deal with the signal timing for the networked intersection area as shown in Fig. 1.

Moreover, if more intersections in the traffic network need to be added, instead of rebuilding the entire traffic model (6), we can build subsystems relating only to the neighboring intersections and use our MIMO control algorithm in a decentralized way, which does not alter the core LQR design as represented in (21) – (23). In this context, our adaptive LQR control algorithm can be made scalable in practice. Indeed, the same method can also be made decentralized by using sub-blocks in matrices A and B, which leads to de-centralized design equations of the same form as expressed in LQR control gain solution (see (18) – (23)).

IV. Baseline Methods and Ablation Study

The performance of the proposed method (adaptive LQR control) was compared with that of three baseline methods: max-pressure, SOTL, and independent deep Q network (IDQN) control methods. Additionally, an ablation study was conducted to test the effectiveness of the proposed method in modeling the system (compared with linear feedback control) and updating system parameters online (compared with offline LQR control). All of these signal control methods were implemented using the VISSIM COM interface.

A. Baseline Methods

Max Pressure. The max-pressure traffic signal control method was inspired by the max pressure algorithm [45] in the field of communication network, which considers the routing and scheduling of packet transmission in a wireless network. Specifically, the max-pressure traffic signal control method models traffic flows as substances in a pipe and optimizes traffic signals to maximize the relief of pressure between incoming and outgoing lanes [33]. The pressure for a certain phase p is defined as:

$$\text{Pressure}(p) = \sum_{l \in L_{p,\text{inc}}} |q_l| - \sum_{l \in L_{p,\text{out}}} |q_l| \tag{24}$$

where l stands for traffic lanes; q_l is the average vehicle queue length in lane l during the last updating time interval; $L_{p,inc}$ is the set of incoming lanes that have green lights in phase p; and $L_{p,out}$ is the set of outgoing lanes from all incoming lanes in $L_{p,inc}$.

The max-pressure method does not have a fixed cycle length. Instead, the algorithm predefines an updating time interval (40 s in this study). At the beginning of each time interval, the algorithm examines the pressures of all signal phases and chooses phases with the maximum pressure as green phases in the incoming time interval. According to an open-source evaluation conducted by Genders and Razavi [33], the max-pressure method has the best performance compared with Webster's method [38], SOTL [46], DQN [47] and deep deterministic policy gradient (DDPG) [48].

SOTL. Similar to the max-pressure method, SOTL [11] does not have a fixed cycle length. Each red-light phase counts the cumulative number of vehicles that arrived during its red light period (k_i). When k_i is greater than a predefined threshold, the current green-light phase switches to red with $k_i = 0$, while the red light that counted turns green. During implementation, minimum green phase duration is applied.

IDQN. The DQN-based traffic signal control has a predefined updating time interval, similar to that of max-pressure control. At the beginning of each time interval, the DQN agent outputs an action based on the current state. This action determines which phase will turn green for the upcoming time interval. We have implemented the independent DQN-based traffic signal control using the algorithm in [33], where each intersection had a local DQN agent that took local state as input and determined actions for that intersection. The Q networks for these agents were updated independently. The state, action, reward, and Q network architecture were defined as follows.

- State: queue and delay of incoming lanes, and the most recent green phase at each intersection. The queue and delay are continuous variables, while the most recent green phase is encoded as a one-hot vector;
- Action: which phase should turn green for the upcoming time interval? In this study, two phases (N-S and E-W) were considered;
- Reward: global reward was used, i.e., the negative average vehicle delay for the whole traffic network was used. Since the reward function is used only during offline training, we can assume that the average network delay is available;
- Q network: two hidden layers of $3|S|$ fully connected neurons with the ReLU activation function, where $|S| = 6$ is the dimension of the state. The output layer has two neurons corresponding to the two phases.

B. Ablation Study

An ablation study was conducted to further assess the performance of the proposed adaptive LQR control method. Two variants of the proposed method were considered in this ablation study: 1) a linear feedback control, which models the network system in a simple linearization form between delay and green time; and 2) an offline LQR method that uses the same initial A, B matrices as those used in the proposed method. Considering the performance difference between the proposed method and the offline LQR method might be insignificant under normal off-peak volumes since the A, B matrices were estimated based on off-peak traffic conditions, the ablation study was conducted using 150% of the off-peak traffic volumes in VISSIM.

Linear Feedback Control. For the linear feedback control [49], system model (1) is linearized to a simple form:

$$\Delta z(k) = H \Delta v(k), \tag{25}$$

where $z \in R^{70}$ are the traffic delays of N-S and E-W direction vehicle flows at each node, $v \in [v_{min}, v_{max}] \in R^{35}$ are N-S direction green times of the intersections, and H is the input-output gain matrix. The feedback control strategy is designed as:

$$\Delta z(k) = -\Gamma z(k), \tag{26}$$

where $\Gamma > 0$ is a design parameter. Applying (26) to (25), one can obtain the controller as:

$$\Delta v(k) = -(H^T H)^{-1} H^T \Gamma z(k). \tag{27}$$

This variant uses a simplified system modeling approach and can be used to determine how system modeling contributes to the performance of the proposed method.

Offline LQR. For the offline LQR method, the system matrices A and B defined in (4) and (5) were estimated from historical data and were not updated during the control. This variant will help show how much gain the adaptive parameter updating design provides.

V. RESULTS

The proposed method was evaluated in the VISSIM simulation environment with the 35-intersection network of Bellevue, Washington, as mentioned in Section II. VISSIM traffic simulation software is a widely used simulation tool especially in the transportation engineering field. VISSIM uses Wiedemann car-following and lane-changing models to model the dynamic movements and interactions of vehicles. Each simulation test lasted for 5,000 seconds, and 30 runs with different random seeds were conducted for each test case. The parameter initial estimates $\Theta(0)$ were chosen within the range $[-10, 10]$ based upon the linearization tests. The random seed initialized a random number generator. With various random seeds, different value sequences were assigned to the stochastic functions in VISSIM, and the simulation of stochastic variations of vehicle arrivals in the network was achieved [32]. This setting affects the following aspect of the simulation.

- Traffic volume: Given an input traffic volume, the actual traffic volume will be a stochastic variable with its expectation being the set one.
- Vehicle arrival: Vehicles arrive with their time headways following a Poisson distribution. With different random seeds, vehicle arrival sequences will be different.
- Turn movements: With turn ratio predefined, random seeds will affect which vehicles will turn right, go straight, or left at an intersection.
- Driving behavior: The car-following and lane-change behavioral models used in VISSIM contain several random parts (e.g., at steady-state car following, drivers will have stochastic accelerations around zero).

The average vehicle delays across the whole simulation period were used as the primary evaluation metric. To investigate how the proposed method can handle various initial traffic states, different initial N-S green times (20 s, 40 s, and 60 s) were chosen. The idea is that an unbalanced initial green time (e.g., 60 s) will result in a congested initial traffic state, whereas a balanced initial green time (e.g., 40 s) may lead to a less congested initial traffic state. In practice, both the signal cycle length and the initial green time can be derived based on Webster's method [38], which considers both the saturation flow rate and the flow ratio for each lane group. For max-pressure, SOTL, and IDQN controls, there is no initial green time because they do not have a fixed cycle length.

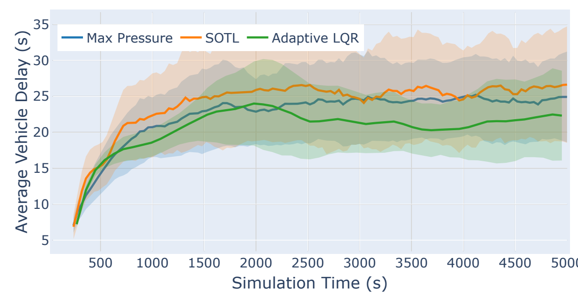
A. Comparison with Baseline Methods

TABLE I presents the performance of the proposed method against the baseline methods. Given different initial green times, it can be seen that the LQR method outperformed max-pressure, SOTL, and IDQN methods, especially when the initial N-S green time was 40 s. The performance of the IDQN method was much worse than the other methods, and this result is consistent with the results presented in [18]. The poor performance of IDQN might be caused by the independence of the reinforcement learning (RL) agents in different traffic intersections, where the control algorithm tries to optimize local traffic situations but fail to cooperate with each other.

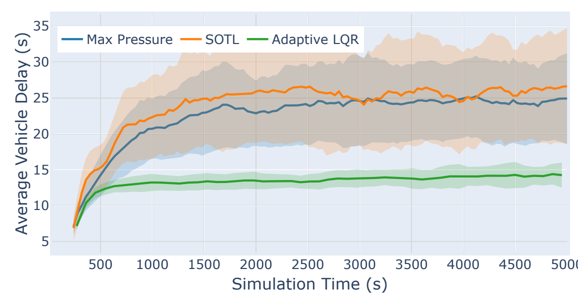
TABLE I: Travel delays with different initial green times and control methods, under normal off-peak traffic volume

Init. Green Time (s)	Control Method	Avg. Veh. Delay (s)	Std. Dev.
20	Adaptive LQR	20.34	5.84
40	Adaptive LQR	**13.24**	2.16
60	Adaptive LQR	16.01	3.59
N/A	SOTL	23.73	8.46
N/A	Max pressure	22.24	6.93
N/A	IDQN	96.83	107.81

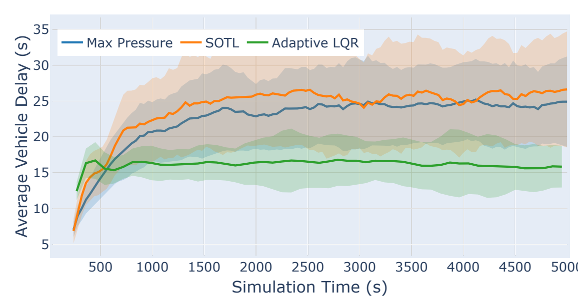
Fig. 5 presents the changing of average vehicle delay along with the progression of the simulation time, with initial green time = 20 s, 40 s, and 60 s. With any initial green times, the proposed LQR method outperformed other baseline methods.



(a) Initial N-S green time = 20 s



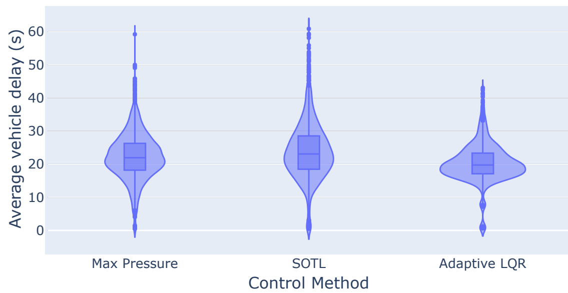
(b) Initial N-S green time = 40 s



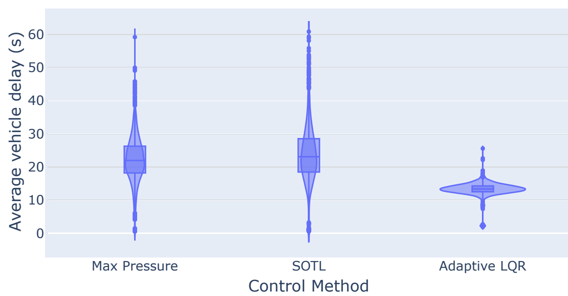
(c) Initial N-S green time = 60 s

Fig. 5: Average vehicle delay during the simulation under normal off-peak traffic volume. Solid colored lines represent the mean, and shaded areas represent the mean $\pm$ one standard deviation. Results of the IDQN method were not visualized because of its large delays.

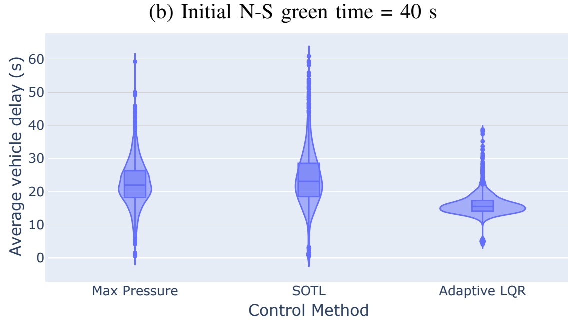
Fig. 6 shows the distribution (violin plot) of average vehicle delays across the whole simulation period. The violin plot shows the probability density of the data at different values. Based on the interquartile range (IQR) inside the violin plots, it can be observed that the LQR method constantly maintained smaller values of lower (25%), median (50%), and upper (75%) quartiles, indicating that its overall performance was consistently better than other methods compared.



(a) Initial N-S green time = 20 s



(b) Initial N-S green time = 40 s



(c) Initial N-S green time = 60 s

Fig. 6: Distribution of average vehicle delays over the simulation period under normal off-peak traffic volume for comparison among different initial green times. Results of IDQN were not visualized due to its much larger delays.

Fig. 7 depicts how the LQR method optimized signal splits at each intersection cycle-by-cycle over the simulation period. In the beginning, all the intersections shared the same initial N-S green time (e.g., 20 s, 40 s, or 60 s). As time progressed, the N-S green time at each intersection was updated every cycle (90 s) to reflect the control output designed to minimize the average travel delay. It can be observed that, at the end of the simulation, the N-S green times of many intersections centered around 40 s, which is the balanced green time between N-S and E-W directions (5 s of yellow and red time between green phases), indicating that traffic volumes in both N-S and E-W directions at these intersections were similar. The N-S green time in some other intersections, nevertheless, continued to change throughout the simulation period. For example, with the initial N-S green time being 20 s, almost all intersections' N-S green time continued to increase throughout the simulation. This was caused by insufficient N-S green time from the beginning and, since the control was adjusted cycle-by-cycle, traffic in the N-S direction was never cleared at those intersections. As a result, the proposed LQR method continued to increase the N-S green time at these intersections to help push traffic through and minimize average delays.

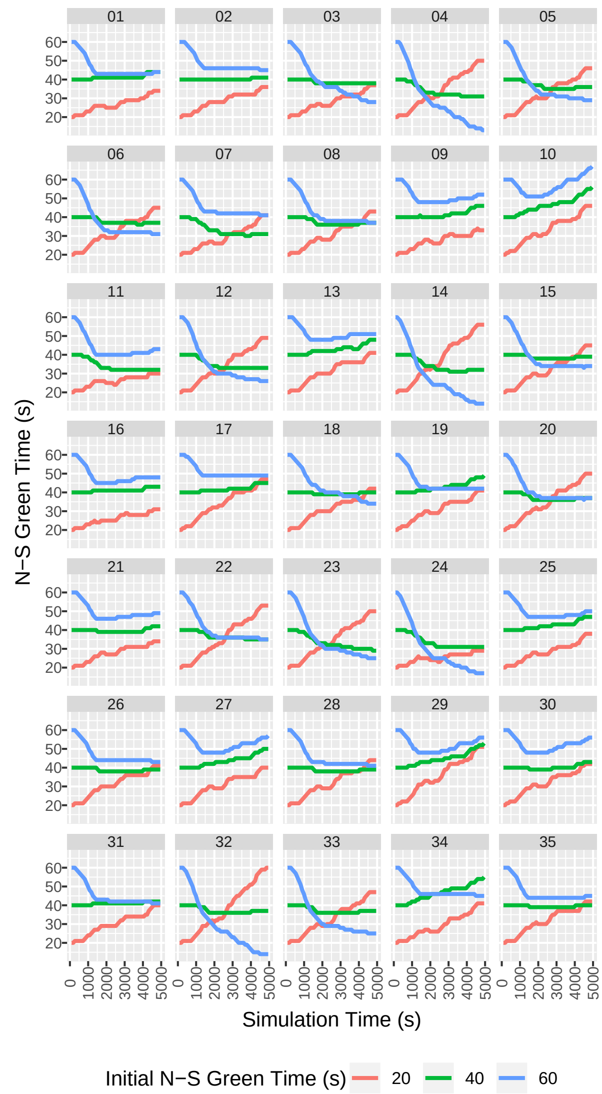


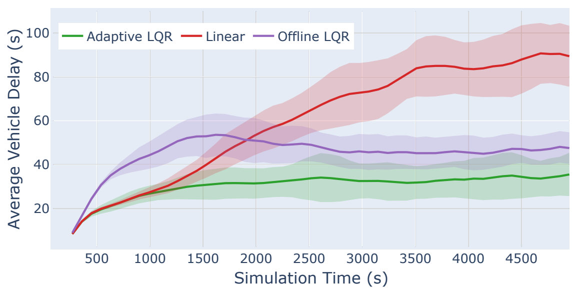
Fig. 7: LQR control and changes of N-S green time of the 35 intersections during the simulation test period with different initial N-S green time under normal off-peak traffic volume.

B. Ablation Study

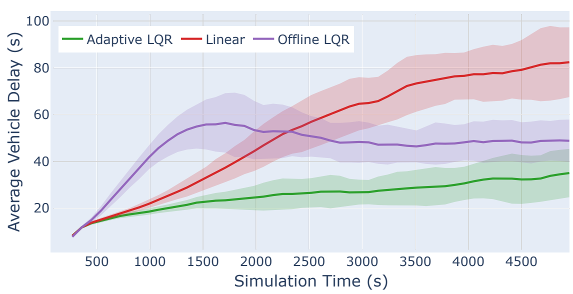
Table II and Fig. 8 present the results of the ablation study. For different initial N-S green times, the proposed adaptive LQR method outperformed both the linear feedback and the offline LQR control methods. The results demonstrate that the system modeling approach and the online parameter updating strategy of the proposed adaptive LQR method have resulted in improved control performance.

TABLE II: Results of ablation study: travel delays with different initial green times and control methods, with 150% off-peak traffic volume.

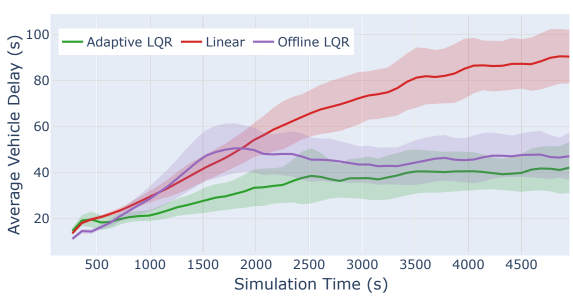
Init. Green Time (s)	Control Method	Avg. Veh. Delay (s)	Std. Dev.
20	Adaptive LQR	**29.99**	10.07
	Linear Feedback	59.69	28.24
	Offline LQR	44.35	12.37
40	Adaptive LQR	**25.28**	9.38
	Linear Feedback	52.51	26.33
	Offline LQR	44.92	14.97
60	Adaptive LQR	**32.99**	11.70
	Linear Feedback	60.41	26.61
	Offline LQR	40.56	14.42
N/A	SOTL	32.09	10.11
N/A	Max pressure	41.70	27.52
N/A	IDQN	138.42	149.86



(a) Initial N-S green time = 20 s



(b) Initial N-S green time = 40 s



(c) Initial N-S green time = 60 s

Fig. 8: Results of ablation study: changing of average vehicle delay during the simulation test period with 150% off-peak traffic volume. Solid colored lines represent the mean and shaded areas represent the mean $\pm$ standard deviation interval.

From the above descriptions, it can be seen that the adaptive LQR control is actually performed for the nonlinear system in an adaptive way. It is used for the large-scale nonlinear model for 35 intersections as given in (1). To the best of our knowledge, this type of adaptive LQR has not been applied to such a large network. In the literature below, they tested LQR-based perimeter control for 16-intersection network, [50].

VI. Conclusion and Future Work

This study proposed an adaptive LQR based signal control method for large urban traffic networks. A linear dynamic traffic system model was built and adaptively updated to reflect how each intersection's signal control input affects network-wide vehicle delays. A linear-quadratic regulator was then built to minimize both traffic delays and control-input changes. With a predefined cycle length, the proposed algorithm adjusted traffic signal control strategies according to observed vehicle delays. The proposed control method was evaluated in the VISSIM traffic simulation environment with a 35-intersection network of Bellevue city, Washington. Traffic counts by approach and turn movement were collected for each intersection to replicate real-world traffic conditions. Simulation results indicate that the proposed method yielded shorter average travel delays in the network when compared with the state-of-the-art max-pressure, SOTL, and IDQN methods. Results of the ablation study also show that the proposed method outperformed the linear feedback control and the offline LQR control under 150% traffic volumes, indicating that the system modeling approach and the online parameter updating strategy of the proposed adaptive LQR method can effectively improve the performance.

Despite the demonstrated advantages, the proposed LQR method in this study has potential limitations, and future research in the following areas is needed.

- Currently, the proposed method only models a specific cycle length (90 s) with two phases to find optimal timing and phasing strategies, different cycle lengths and phasing combinations should be explored. Looking at system model (1) or (6), it can be seen that adding more phases at each intersection is not expected to constitute much difficulty as this would merely increase the dimensionality of input $u(k)$ and $y(k)$ in (6), while the control design method remains the same as those in (21) – (23).
- The current system model assumes no noise in (1). Future work is needed to incorporate a stochastic noise term into the dynamic traffic system model to better account for the variability of the traffic system.
- The proposed model assumes that the system can be linearized as shown in (3) or (6). Future work will consider nonlinear traffic system modeling approaches, such as bi-linear and neural network models, aiming to better represent true characteristics of the traffic system.
- The proposed method has been tested in a traffic simulation environment. Implementing the proposed algorithm in a real-world traffic corridor/network is needed to test its effectiveness.

Acknowledgment

This work is supported by the US Department of Energy, Vehicle Technologies Office, Energy Efficient Mobility Systems (EEMS) program, Systems and Modeling for Accelerated Research in Transportation (SMART) Mobility Consortium, and the Laboratory Directed Research and Development Program of Oak Ridge National Laboratory, managed by UT-Battelle, LLC, for the US Department of Energy under contract DE-AC05-00OR22725. These are gratefully acknowledged.

References

[1] M. Papageorgiou, C. Diakaki, V. Dinopoulou, A. Kotsialos, and Y. Wang, "Review of road traffic control strategies," *Proceedings of the IEEE*, vol. 91, no. 12, pp. 2043–2067, 2003.

[2] K. Triantis, S. Sarangi, D. Teodorović, and L. Razzolini, "Traffic congestion mitigation: combining engineering and economic perspectives," *Transportation Planning and Technology*, vol. 34, no. 7, pp. 637–645, 2011.

[3] L. B. De Oliveira and E. Camponogara, "Multi-agent model predictive control of signaling split in urban traffic networks," *Transportation Research Part C: Emerging Technologies*, vol. 18, no. 1, pp. 120–139, 2010.

[4] X. Li and J.-Q. Sun, "Multi-objective optimal predictive control of signals in urban traffic network," *Journal of Intelligent Transportation Systems*, vol. 23, no. 4, pp. 370–388, 2019.

[5] A. Di Febbraro, D. Giglio, and N. Sacco, "A deterministic and stochastic petri net model for traffic-responsive signaling control in urban areas," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 2, pp. 510–524, 2015.

[6] D. I. Robertson, "'TANSYT' method for area traffic control," *Traffic Engineering & Control*, vol. 8, no. 8, 1969.

[7] P. Koonce and L. Rodegerdts, "Traffic signal timing manual." United States. Federal Highway Administration, Tech. Rep., 2008.

[8] S. Mikami and Y. Kakazu, "Genetic reinforcement learning for cooperative traffic signal control," in *Proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence*. IEEE, 1994, pp. 223–228.

[9] L. Singh, S. Tripathi, and H. Arora, "Time optimization for traffic signal control using genetic algorithm," *International Journal of Recent Trends in Engineering*, vol. 2, no. 2, p. 4, 2009.

[10] P. Varaiya, "Max pressure control of a network of signalized intersections," *Transportation Research Part C: Emerging Technologies*, vol. 36, pp. 177–195, 2013.

[11] S. Cools, C. Gershenson, and B. D'Hooghe, "Self-organizing traffic lights: A realistic simulation," in *Advances in applied self-organizing systems*. Springer, 2013, pp. 45–55.

[12] P. Balaji, X. German, and D. Srinivasan, "Urban traffic signal control using reinforcement learning agents," *IET Intelligent Transport Systems*, vol. 4, no. 3, pp. 177–188, 2010.

[13] Y. Bi, D. Srinivasan, X. Lu, Z. Sun, and W. Zeng, "Type-2 fuzzy multi-intersection traffic signal control with differential evolution optimization," *Expert systems with applications*, vol. 41, no. 16, pp. 7338–7349, 2014.

[14] A. Nagare and S. Bhatia, "Traffic flow control using neural network," *Traffic*, vol. 1, no. 2, pp. 50–52, 2012.

[15] H. Yin, S. Wong, J. Xu, and C. Wong, "Urban traffic flow prediction using a fuzzy-neural approach," *Transportation Research Part C: Emerging Technologies*, vol. 10, no. 2, pp. 85–98, 2002.

[16] B. Abdulhai, R. Pringle, and G. J. Karakoulas, "Reinforcement learning for true adaptive traffic signal control," *Journal of Transportation Engineering*, vol. 129, no. 3, pp. 278–285, 2003.

[17] I. Arel, C. Liu, T. Urbanik, and A. G. Kohls, "Reinforcement learning-based multi-agent system for network traffic signal control," *IET Intelligent Transport Systems*, vol. 4, no. 2, pp. 128–135, 2010.

[18] T. Chu, J. Wang, L. Codecà, and Z. Li, "Multi-agent deep reinforcement learning for large-scale traffic signal control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, pp. 1086–1095, 2019.

[19] H. Wei, G. Zheng, H. Yao, and Z. Li, "Intellilight: A reinforcement learning approach for intelligent traffic light control," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 2496–2505.

[20] X. Liang, X. Du, G. Wang, and Z. Han, "A deep reinforcement learning network for traffic light cycle control," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 2, pp. 1243–1253, 2019.

[21] X. Wang, L. Ke, Z. Qiao, and X. Chai, "Large-scale traffic signal control using a novel multi-agent reinforcement learning," *arXiv preprint arXiv:1908.03761*, 2019.

[22] X. Li and J.-Q. Sun, "Signal multiobjective optimization for urban traffic network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 11, pp. 3529–3537, 2018.

[23] X. Li and J. Sun, "Turning-lane and signal optimization at intersections with multiple objectives," *Engineering Optimization*, vol. 51, no. 3, pp. 484–502, 2019.

[24] A. Stevanovic, *Adaptive traffic control systems: domestic and foreign state of practice*, 2010, no. Project 20-5 (Topic 40-03).

[25] A. G. Sims, "The sydney coordinated adaptive traffic system," in *Engineering Foundation Conference on Research Directions in Computer Control of Urban Traffic Systems, 1979, Pacific Grove, California, USA*, 1979.

[26] P. Hunt, D. Robertson, R. Bretherton, and M. C. Royle, "The SCOOT on-line traffic signal optimisation technique," *Traffic Engineering & Control*, vol. 23, no. 4, 1982.

[27] N. H. Gartner, *OPAC: A demand-responsive strategy for traffic signal control*, 1983, no. 906.

[28] C. Diakaki, M. Papageorgiou, and K. Aboudolas, "A multivariable regulator approach to traffic-responsive

network-wide signal control," *Control Engineering Practice*, vol. 10, no. 2, pp. 183–195, 2002.

[29] W. Kraus, F. A. de Souza, R. C. Carlson, M. Papageorgiou, L. D. Dantas, E. Camponogara, E. Kosmatopoulos, and K. Aboudolas, "Cost effective real-time traffic signal control using the TUC strategy," *IEEE Intelligent Transportation Systems Magazine*, vol. 2, no. 4, pp. 6–17, 2010.

[30] H. Kwakernaak and R. Sivan, *Linear optimal control systems*. Wiley-interscience New York, 1972, vol. 1.

[31] W. Hu, H. Wang, B. Du, and L. Yan, "A multi-intersection model and signal timing plan algorithm for urban traffic signal control," *Transport*, vol. 32, no. 4, pp. 368–378, 2017.

[32] J. A. Laval and H. Zhou, "Large-scale traffic signal control using machine learning: some traffic flow considerations," *arXiv preprint arXiv:1908.02673*, 2019.

[33] W. Genders and S. Razavi, "An open-source framework for adaptive traffic signal control," *arXiv preprint arXiv:1909.00395*, 2019.

[34] W. Zhu, "A connected vehicle based coordinated adaptive navigation system," Ph.D. dissertation, University of Washington, 2019.

[35] *Vissim 9 User Manual*, PTV Group, Karlsruhe, Germany, 2016.

[36] R. Wiedemann, *Simulation des Straßenverkehrsflusses. Schriftenreihe Heft 8*. Instituts für Verkehrswesen der Universitä Karlsruhe, 1974.

[37] M. Zhu, X. Wang, A. Tarko *et al.*, "Modeling car-following behavior on urban expressways in shanghai: A naturalistic driving study," *Transportation research part C: emerging technologies*, vol. 93, pp. 425–445, 2018.

[38] F. Webster, "Traffic signal settings, road research technical," *London, UK*, no. 39, 1958.

[39] B. D. O. Anderson and J. B. Moore, *Linear Optimal Control*. Prentice Hall, 1971.

[40] G. Tao, *Adaptive control design and analysis*. John Wiley & Sons, 2003, vol. 37.

[41] S. Tu and B. Recht, "Least-squares temporal difference learning for the linear quadratic regulator," *arXiv preprint arXiv:1712.08642*, 2017.

[42] ——, "The gap between model-based and model-free methods on the linear quadratic regulator: An asymptotic viewpoint," *arXiv preprint arXiv:1812.03565*, 2018.

[43] H. Mania, S. Tu, and B. Recht, "Certainty equivalent control of lqr is efficient," *arXiv preprint arXiv:1902.07826*, 2019.

[44] G. C. Goodwin and K. S. Sin, *Adaptive filtering prediction and control*. Courier Corporation, 2014.

[45] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," in *29th IEEE Conference on Decision and Control*. IEEE, 1990, pp. 2130–2132.

[46] C. Gershenson, "Self-organizing traffic lights," *arXiv preprint nlin/0411066*, 2004.

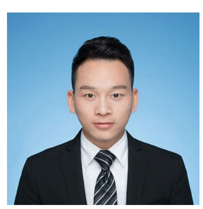
[47] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[48] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.

[49] H. Wang, C. Wang, M. Zhu, and W. Hong, "Globalized modeling and signal timing control for large-scale networked intersections," in *Proceedings of The 2nd ACM/EIGSCC Symposium On Smart Cities and Communications (SCC 2019)*. Portland, OR, USA: ACM, 2019.

[50] H. Zhang, L. Zhang, Z. He, and M. Li, "LQR-based perimeter control approach for urban road traffic networks," in *2017 6th Data Driven Control and Learning Systems (DDCLS)*. IEEE, 2017, pp. 745–749.

Hong Wang received master and doctorate degrees from the Huazhong University of Science and Technology, Wuhan, China, in 1984 and 1987, respectively. He was a Research Fellow with Salford University (Salford, UK), Brunel University (Uxbridge, UK) and Southampton University (Southampton, UK) before joining the University of Manchester Institute of Science and Technology (UMIST), Manchester, U.K., in 1992. Wang was a Chair Professor in Process Control of complex industrial systems at the University of Manchester (UK) from 2002 to 2016, where he was the deputy head of the Paper Science Department, director of the UMIST Control Systems Centre between 2004 - 2007 (which is the birthplace of Modern Control Theory established in 1966). He was University Senate member and a member of general assembly during his time in Manchester. Between 2016 and 2018, he was with Pacific Northwest National Laboratory (PNNL, Richland, WA, USA) as a Laboratory Fellow and Chief Scientist, and was the co-leader and chief scientist for the Control of Complex Systems Initiative (https://controls.pnnl.gov/). Wang joined Oak Ridge National Laboratory in Jan 2019, he was an associate editor for IEEE Transactions on Automatic Control, IEEE Transactions on Control Systems Technology and IEEE Transactions on Automation Science and Engineering. He is also a member for three IFAC Committees. Hong Wang's research focuses on stochastic distribution control, fault diagnosis and tolerant control and intelligent controls with applications to transportation system area.

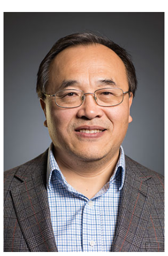


Meixin Zhu is a Ph.D. candidate at the University of Washington. He also serves as a research assistant in Smart Transportation Applications and Research Laboratory (STAR Lab) at the University of Washington, advised by Prof. Yinhai Wang. He received the BSc and MSc degree in Traffic Engineering in 2015 and 2018 respectively at Tongji University. Zhu's research interests include autonomous driving, artificial intelligence, big data analytics, driving behavior, traffic-flow modeling and simulation, and naturalistic driving study.

Wanshi Hong is a Ph.D. candidate at the University of Virginia. She received her BSC degree in Electrical Engineering in Xi'an Jiaotong University, in 2016. She then received her masters degree in Electrical and Computer Engineering at University of Virginia, in 2017. Wanshi's research interests include adaptive control, optimal control, on-line parameter estimation, and machine learning.

Chieh (Ross) Wang is an R&D Staff in the Energy and Transportation Science Division at the Oak Ridge National Laboratory. He earned his Ph.D. in Civil and Environmental Engineering from the Georgia Institute of Technology in 2017. He received his M.S. and B.S. degrees in Civil Engineering from the National Taiwan University in 2007 and 2005. His research revolves around intelligent transportation asset management, transportation data analytics and visualization, intelligent vehicles, and smart cities.

Gang Tao received his B.S. degree in Electrical Engineering from the University of Science and Technology of China in 1982, his M.S. degrees in Electrical Engineering, Computer Engineering and Applied Mathematics in 1984, 1987, 1989, respectively, and Ph.D. degree in Electrical Engineering in 1989 all from the University of Southern California. He was a visiting assistant professor at Washington State University from 1989 to 1991, an assistant research engineer at the University of California at Santa Barbara from 1991 to 1992, and an assistant professor at the University of Virginia from 1992 to 1998, where he is now a professor. He was a guest editor for International Journal of Adaptive Control and Signal Processing, and is currently an associate editor for IEEE Transactions on Automatic Control and for the discipline's most important journal Automatica. He has been a program committee member for numerous international conferences, among them, American Control Conference and the Conference on Decision and Control in 2001 and 2002. He has published six books, and over 400 technical papers and book chapters on adaptive control, nonlinear control, multivariable control, optimal control, and control applications and robotics. He is a Fellow of IEEE. His research has been supported by NSF, ARMY, NASA, MedQuest, SCEEE and Edison Power.



Yinhai Wang is a professor in both the Civil and Environmental Engineering (CEE) and Electrical and Computer Engineering (ECE) Departments of the University of Washington (UW). Dr. Wang has a Ph.D. in transportation engineering from the University of Tokyo (1998), a master's degree in computer science and engineering from the UW, and another master's degree in construction management and a bachelor's degree in civil engineering from Tsinghua University, China. Dr. Wang is the founder of the Smart Transportation Applications and Research Laboratory (STAR Lab) at the UW. He also serves as the director for Pacific Northwest Transportation Consortium (PacTrans), USDOT University Transportation Center for Federal Region 10. Dr. Wang has conducted extensive research in traffic sensing, transportation big data management and analytic, large-scale transportation system analysis, transportation data science, traffic operations, and decision support.