Data-driven urban traffic model-free adaptive iterative learning control with traffic data dropout compensation

Dai Li | Zhongsheng Hou

1 Advanced Control Systems Laboratory, School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing, China
2 School of Automation, Qingdao University, Qingdao, China

Correspondence
Zhongsheng Hou, Advanced Control Systems Laboratory, School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China.
Email: zshou@qdu.edu.cn

Funding information
National Natural Science Foundation of China, Grant/Award Numbers: 61433002, 61833001

Abstract
In this paper, to fully utilize the urban traffic flow characteristics of similarity and repeatability without using a mathematical traffic model, a data-driven urban traffic control strategy based on model-free adaptive iterative learning control (MFAILC) scheme is put forward. Firstly, by dynamically linearizing the urban traffic dynamics along the iteration axis, the traffic network system is transformed into a MFAILC data model with the help of repetitive pattern of urban traffic flow. Then, the traffic controller is designed based on the derived MFAILC data model only using the I/O data of the traffic network. Finally, a traffic data compensation method is proposed to deal with data dropout problem. Simulation study verifies the feasibility and effectiveness of the proposed control method.

1 INTRODUCTION

Urban traffic congestion is a severe problem all over the world causing a large amount of damage to economy and society, and many kinds of traffic control strategies have been developed to alleviate urban traffic congestion.

For fixed-time control strategies, such as Webster [1], MAXBAND [2] and TRANSYT [3], traffic signal settings are obtained based on off-line traffic data of the network, and kept fixed for a specific period of time. The fixed-time strategies are not able to cope with real-time variation of traffic flow although they are easy to be carried out in practice. To solve this problem, traffic responsive control strategies were designed based on various mathematical traffic models, such as store-forward model [4, 5] and S model [6–8]. Meanwhile, all the detailed dynamics of links and intersections are needed for the responsive network-wide traffic control strategies, resulting in large computation burden and weak reliability.

Then, in order to relieve regional traffic congestion just by metering the amount of vehicles entering the congested region without considering traffic dynamics in the level of links and intersections, perimeter control has become an attractive research direction in which a macroscopic level method called macroscopic fundamental diagram (MFD) is utilized. MFD provides a unimodal, low-scatter relationship between the network vehicle number and network space-mean traffic flow [9, 10]. So far, many control and optimization theories have been applied in perimeter control, such as linear quadratic regulator [11], adaptive control [12, 13], robust control [14, 15] and model predictive control [16, 17].

However, mathematical traffic models are needed for all the aforementioned traffic control strategies, which makes them hard to use and to implement in practice. Urban traffic networks are so complex and huge that it is quite difficult or even impossible to establish an accurate mathematical traffic model. Even if a precise traffic model is available, it could be too complex with strong nonlinearity and large uncertainties, thus the resulting traffic controller would be time-consuming and too complicated for on-line applications. Considering that the urban traffic systems produce a lot of traffic data every day, it is possible to make use of traffic data instead of mathematical traffic models to deal with urban traffic control problems.

In this case, a novel data-driven model-free adaptive control (MFAC) scheme was utilized in urban traffic control [18, 19]. MFAC is a pure data-driven control approach developed to cope with control problems for a class of unknown nonlinear and non-affine systems, and it was originally proposed in [20]. The basis of MFAC is to dynamically linearize the system...
In this paper, an urban traffic control strategy based on MFAILC scheme is put forward. In the proposed control method, the traffic controller can iteratively improve the control performance by learning from the previous executions utilizing the MFAILC scheme without using any mathematical traffic model. Moreover, inspired by the idea of the MFAC scheme with data dropout compensation [43, 44], we design a data dropout compensation method for the proposed urban traffic MFAILC strategy in the case of traffic data dropout.

The main contributions of this work are summarized as follows:

1. Compared with the work in [38], the proposed control method is a microscopic MFAILC-based urban traffic control strategy. That is, instead of macroscopic characteristic of the network, the detailed signal timing scheme in the network is also considered in this work. In a sequel, the proposed control method can be directly used for practical traffic networks.

2. We extend the work proposed in [39] to be a network-wide traffic control strategy, which gives a more comprehensive consideration for urban traffic network control.

3. A data dropout compensation method is designed for the urban traffic MFAILC strategy based on the idea of MFAC scheme with data dropout compensation [43, 44].

This paper is organized as follows. In Section 2, the basic notations and dynamics of urban traffic networks are introduced. Then the MFAILC data model and the MFAILC-based urban traffic control strategy are presented in Section 3. The traffic data dropout compensation method is proposed in Section 4. Afterwards, simulation study of a real traffic network is conducted in Section 5. Finally, conclusions and future works are given in Section 6.

To facilitate the description of the proposed control strategy, the following variables which will be used in the remainder of the paper are summarized in Table 1.

## 2 DYNAMICS OF URBAN TRAFFIC NETWORKS

An urban network constitutes a number of intersections and links, and an illustrative example of urban traffic networks is shown in Figure 1. There are five intersections $i, j_1, j_2, j_3, j_4 \in I$ in Figure 1, with $I$ the set of intersections in the network; $(j, i) \in Z$ is a link on which vehicles travel from intersection $j$ to intersection $i$, where $Z$ is the set of links in the network.

A signal phase of an intersection represents a specific set of traffic flows that have the right of way to cross the intersection simultaneously. Then, the green time for a signal phase is defined as the period of time that the corresponding set of traffic flows can cross the intersection during each signal cycle. Finally, the signal cycle of an intersection equals the summation of green times for all signal phases, plus the lost time of the intersection, which is imposed to avoid interference among incompatible traffic flows of consecutive signal phases [45].
TABLE 1  List of variables

| Variable | Definition |
|----------|------------|
| $I$      | Set of intersections in the network |
| $Z$      | Set of links in the network |
| $T$      | Sampling interval of the traffic system, and common cycle length of the intersections |
| $K$      | Number of time instants during each iteration |
| $L_j$    | Lost time of intersection $i$ within a signal cycle |
| $e_{i,j}(k)$ | Number of vehicles entering link $(j, i)$ at time instant $k$ |
| $l_{i,j}(k)$ | Number of vehicles leaving link $(j, i)$ at time instant $k$ |
| $n_{i,j}(k)$ | Number of vehicles arriving at the queue tail of link $(j, i)$ at time instant $k$ |
| $q_{i,j}(k)$ | Number of queueing vehicles on link $(j, i)$ at time instant $k$ |
| $n_{i,j}(k)$ | Number of vehicles on link $(j, i)$ at time instant $k$ |
| $y(k, b)$ | Number of vehicles in the entire network at time instant $k$, iteration $b$ |
| $y^{\text{ref}}$ | Reference value of $y(k, b)$ |
| $g_{i,j}(k)$ | Green time of $i$th signal phase of intersection $j$ at time instant $k$ |
| $g_{i,j,\text{min}}$ | Minimum green time of $i$th signal phase of intersection $j$ |
| $p_i$ | Number of signal phases of intersection $i$ |
| $g_i(k)$ | Vector of phase green times for all signal phases in the network at time instant $k$, iteration $b$ |
| $\phi(k, b)$ | Pseudo-gradient of the MFAILC data model of the traffic system at time instant $k$, iteration $b$ |
| $\| \cdot \|$ | Absolute value of a scalar |
| $\| \|_2$ | Euclidean norm of a vector |
| $\Delta_t$ | Difference of a variable or a vector between adjacent iterations of the same time instant |

An illustrative diagram of an intersection with 4 signal phases is shown in Figure 2. Then, we define $g_i(k) \in \mathbb{R}^{p_i}$ as the vector of phase green times of intersection $i$ at time instant $k$, which can be described as

$$g_i(k) = \left[ g_{i,1}(k), \ldots, g_{i,p_i}(k) \right]^T,$$

where $p_i$ is the number of signal phases of intersection $i$.

The green times of the signal phases of each intersection $i \in I$ should satisfy the following cycle time constraint, the minimum and maximum green time constraints:

$$\sum_{j=1, \ldots, p_i} g_{i,j}(k) = T - L_j,$$

$$g_{i,j}(k) \geq g_{i,j,\text{min}}, \quad j = 1, \ldots, p_i,$$

where $T$ is the common cycle length of all intersections in the network, and $L_j$ is the total lost time of intersection $i$ within a signal cycle. The cycle time constraint can be derived from the definition of the signal cycle, and $g_{i,j,\text{min}}$ is the minimum green time of $i$th signal phase of intersection $j$ introduced to ensure enough green time for the vehicles and pedestrians to cross the intersection.

Assuming the sampling interval of the traffic system equals the common cycle length $T$, the dynamics of the number of vehicles on link $(j, i)$ and in the entire network can be described, respectively, as

$$n_{i,j}(k + 1) = n_{i,j}(k) + e_{i,j}(k) - l_{i,j}(k),$$

$$y(k) = \sum_{i,j \in \mathcal{E}} n_{i,j}(k),$$

where $n_{i,j}(k)$ and $y(k)$ are the number of vehicles on link $(j, i)$ and in the entire network, respectively, at time instant $k$; $e_{i,j}(k)$ and $l_{i,j}(k)$ are the number of vehicles entering and leaving link $(j, i)$, respectively, at time instant $k$.

In this paper, we adopt the S model [6, 46], which gives a good balance between accuracy and complexity, to describe the dynamics of $e_{i,j}(k)$ and $l_{i,j}(k)$ as follows:

$$e_{i,j}(k) = \sum_{m \in I} \min \left( \lambda_{(m,i,j)} g_{i,p_{i,m,j}}(k), a_{(m,j)}(k) \right) + q_{(m,j)}(k), \beta_{(m,i,j)} \left( C_{i,j} - n_{i,j}(k) \right),$$

$$l_{i,j}(k) = \sum_{m \in I} \min \left( \lambda_{(i,m)} g_{i,p_{i,m,j}}(k), a_{(i,m)}(k) \right) + q_{(i,m)}(k), \beta_{(i,m)} \left( C_{i,m} - n_{i,m}(k) \right).$$
where $\lambda_{(j,i,m)}$ is the saturation flow rate leaving link $(j,i)$ towards link $(i,m)$, $g_{(j,i,m)}(k)$ is the green time for traffic flow from link $(j,i)$ towards link $(i,m)$ at time instant $k$, $a_{(j,i,m)}(k)$ is the number of vehicles arriving at the queue tail of link $(j,i)$ towards link $(i,m)$ at time instant $k$, $q_{(j,i,m)}(k)$ is the number of queuing vehicles on link $(j,i)$ towards link $(i,m)$ at time instant $k$, $\beta_{(j,i,m)}$ is the ratio of vehicles leaving from link $(j,i)$ towards link $(i,m)$, $C_{(j,i)}$ is the capacity of link $C_{(j,i)}$ measured in number of vehicles. Note that other mathematical traffic models, such as the store-and-forward model [4, 47] and the BLX model [48, 49], can also be utilized to describe the urban traffic dynamics.

Based on (3)–(6), the dynamics of the number of vehicles in the entire network can be transformed into

$$y(k + 1) = \sum_{(j,i) \in Z} n_{(j,i)}(k) + \sum_{(j,i) \in Z} \left( n_{(j,i)}(k) + e_{(j,i)}(k) - l_{(j,i)}(k) \right)$$

$$= y(k) + \sum_{(j,i) \in Z} \left( e_{(j,i)}(k) - l_{(j,i)}(k) \right)$$

$$= y(k) + f(G(k)), \quad (7)$$

where $f(\cdot)$ is an unknown nonlinear function, $G(k) = [g_i(k)]_{i \in I}$ is the vector of phase green times of all intersections at time instant $k$. By optimizing $G(k)$, the number of vehicles $y(k)$ in the network can be determined.

It can be seen from (5) and (6) that $\lambda_{(j,i,m)}$, $\beta_{(j,i,m)}$ and $C_{(j,i)}$ for all $(j,i) \in Z$ and $(i,m) \in Z$ are known constants, while $n_{(j,i)}(k)$, $q_{(j,i,m)}(k)$ and $a_{(j,i,m)}(k)$ can be directly measured by the traffic detectors, thus $G(k)$ is the only variable that needs to be optimized in traffic dynamics (7). In this case, the traffic system (7) can be viewed as a multi inputs single output (MISO) system with $G(k)$ as vector of system inputs, and $y(k)$ as system output.

It is noted that the traffic model described above is just an illustrative example to help readers understand the basic traffic dynamics, but it does not participate in the controller design. Instead, MFAILC scheme is utilized to design the traffic controller due to the three outstanding features of the urban traffic networks. Firstly, the dynamics of urban traffic systems is highly nonlinear so that the traffic modelling and the model-based controller design are difficult to be executed, and their performances are also questionable. Then, the traffic I/O data is easy to be obtained by the traffic detectors in modern society. Finally, urban traffic flow has repetitive operation pattern. Therefore, these features motivate us to use the MFAILC scheme to deal with urban traffic control problem.

3  |  MFAILC STRATEGY FOR URBAN TRAFFIC NETWORKS

3.1  |  Control problem formulation

The objective of the proposed MFAILC-based traffic controller is to regulate the number of vehicles in the network $y(k)$ tracking the reference value $y^{ref}$ by optimizing $G(k)$, thus the traffic flow efficiency can be maximized, and traffic congestion can be alleviated.

Specifically, we first transform the MISO traffic system (7) into a MFAILC data model along the iteration axis with the help of a novel concept, the pseudo-gradient; the pseudo-gradient in the linearized data model is updated only using the measured I/O data of the traffic system without the need of any information of a mathematical traffic model. Then the derived MFAILC data model can be utilized to design the traffic controller.

There are two attractive properties of the proposed MFAILC-based urban traffic control method. Firstly, no mathematical traffic model is required in the controller design by virtue of the MFAILC dynamic linearization data model, thus the difficulties of accurate traffic modelling and model-based traffic controller design can be avoided. Secondly, the performance of traffic controller can be improved iteratively with the help of the iterative learning ability of the MFAILC scheme and repetitive pattern of urban traffic flow.

3.2  |  MFAILC data model of the traffic networks

Generally, the dynamics of $y(k)$ shown in (7) can be rewritten as the following form of the discrete-time nonlinear system:

$$y(k + 1) = f(y(k), ..., y(k - m_j), G(k), ..., G(k - m_a)), \quad (8)$$

where $f(\cdot)$ is an unknown nonlinear function; $m_j$ and $m_a$ are the memory lengths of the traffic system for the number of vehicles in the network and for the system inputs. It is not necessary to know the values of $m_j$ and $m_a$ as they do not participate in the controller design.

Note that urban traffic flow usually exhibits iterative pattern in a daily manner, although they vary with the time in a day. For instance, the traffic flow always starts from a very low level at midnight, and increases gradually up to the first peak during morning rush hour, and to the second peak during evening rush hour.

In this case, to distinguish the traffic dynamics of different days, an extra indicator $b$ is added to the variables to present the number of a day (iteration). Then (8) can be further transformed into the following form under the MFAILC framework:

$$y(k + 1, b) = f(y(k, b), ..., y(k - m_j, b), G(k, b), ..., G(k - m_a, b)), \quad (9)$$

where $G(k, b)$ and $y(k, b)$ are the vector of system inputs and system output at time instant $k$, iteration $b$, respectively; $b = 1, 2, ..., K \in \{1, 2, ..., K\}$, where $K$ is the total number of time instants at each iteration $b$.

Note that the detailed form of (9) is not needed, as it will be transformed into the equivalent MFAILC data model. Similar to the MFAC scheme, the MFAILC data models can also be classified into the CFDL, the PFDL and the FFDL forms.
Without the loss of generality and for simplicity, the CFDL-MFAILC data model \([36, 37]\) will be used for the MFAILC controller design. Before the CFDL-MFAILC data model is elaborated, some assumptions are made on system (9).

**Assumption 1.** The partial derivatives of \(\tilde{J}(\cdot)\) with respect to every entry of the variable \(G(k, b)\) are continuous.

**Assumption 2.** Subsystem (9) satisfies the generalized Lipschitz condition along the iteration axis if \(\Delta G(k, b) \neq 0\) for any \(k \in \{1, 2, ..., K\}\) and \(b = 1, 2, ...,\), that is:

\[
|\Delta y(k + 1, b)| \leq b\|\Delta G(k, b)\|,
\]

where \(|| \cdot ||\) denotes the Euclidean norm of a vector, \(\Delta G(k, b) = G(k, b) - G(k, b - 1)\), \(\Delta y(k, b) = y(k) - y(k, b - 1)\), and \(b\) is a positive constant.

**Remark 1.** Assumption 1 is easy to be verified from the mathematical relationship between the green times and number of vehicles in the network. Assumption 2 is a physical constraint by the inherent nature of urban traffic system, that is, finite change of green times of the signal phases at each time instant \(k\) of iteration \(b\) would not lead to infinite change on the number of vehicles in the network at the next time step.

**Theorem 1.** \([21, 38]\) Given the nonlinear MISO system (9), if Assumptions 1 and 2 are satisfied and \(\Delta G(k, b) \neq 0\), there must exist a time-varying vector \(\hat{\phi}(k, b) \in R^{([k]+3[2]}\) named pseudo-gradient, such that the original system (9) could be transformed equivalently into the MISO CFDL-MFAILC data model:

\[
\Delta y(k + 1, b) = \phi^T(k, b)\Delta G(k, b),
\]

\[
\phi(k, b) = [\phi_1(k, b), ..., \phi_p(k, b)]^T,
\]

where \(P\) is the total number of signal phases in the network, and \([Z]\) is the total number of elements in set \(Z\), namely, the number of links in the network.

**Proof.** See Appendix A.

It should be noted that the pseudo-gradient \(\hat{\phi}(k, b)\) in (11) is unknown. In order to estimate \(\hat{\phi}(k, b)\), the following cost function is utilized:

\[
J(\hat{\phi}(k, b)) = \left|\Delta y(k + 1, b - 1) - \hat{\phi}^T(k, b)\Delta G(k, b - 1)\right|^2 + \mu \left|\phi(k, b) - \hat{\phi}(k, b - 1)\right|^2,
\]

where \(\mu > 0\) is a weighting factor, and \(\hat{\phi}(k, b)\) is the estimation of the pseudo-gradient. The first term of (12) is the difference between the true measured number of vehicles in the network and the MFAILC data model output; while the second term penalizes large variation of the pseudo-gradient, which intends to enhance the robustness of the estimation algorithm to disturbances and outliers.

Minimizing (12) with respect to \(\hat{\phi}(k, b)\), we get the following estimation algorithm with help of matrix inversion [21]:

\[
\hat{\phi}(k, b) = \hat{\phi}(k, b - 1) + \frac{\eta\Delta G(k, b - 1)}{\mu + \|\Delta G(k, b - 1)\|^2} \times (\Delta y(k + 1, b - 1) - \hat{\phi}^T(k, b - 1)\Delta G(k, b - 1)),
\]

where the step factor \(\eta \in (0, 1]\) is added to make the estimation algorithm more generic.

**Remark 2.** By transforming urban traffic dynamics (9) into MFAILC data model (11), some impocrine problems in existing linearization methods, such as the dropout of high order terms in Taylor’s linearization, and the requirement of model information in piecewise linearization, can be avoided. By virtue of the traffic detectors, the traffic data can be directly utilized to estimate the pseudo-gradient \(\hat{\phi}(k, b)\) without using any information of a mathematical traffic model. In the next, the derived MFAILC data model (11)–(13) will be used to design the traffic controller.

### 3.3 Traffic controller design

The purpose of the urban traffic controller is to improve the traffic flow efficiency in a network, as well as to prevent large fluctuations of traffic signal settings. In this case, we adopt the following cost function for the MFAILC-based traffic controller:

\[
\min_{G(k, b)} J = \left|y(k + 1, b) - y_{\text{ref}}\right|^2 + \gamma\|G(k, b) - G(k - 1, b)\|^2,
\]

where \(0 < \gamma < 1\). The first term of (14) aims to improve the traffic efficiency in a network by penalizing the difference between \(y(k + 1, b)\) and its predefined reference value \(y_{\text{ref}}\). \(y_{\text{ref}}\) is set as a constant and corresponds to the maximum space-mean traffic flow in the network, and it can be determined off-line based on the MFD [9, 10] of a network, or by a traffic management engineer. While the second term is used to restrict the changes of signal settings between adjacent time instants, as too large fluctuations of signal settings are not safe and realistic to the operation of a real traffic network.

**Remark 3.** Theoretically, \(y_{\text{ref}}\) should have different values in different period of time in a day, for instance, in peak and off-peak hours. However, \(y_{\text{ref}}\) is always set as a constant in practical traffic operations for the sake of simplicity.
The traffic control problem can be expressed as follows:

$$
\min_{G(k,b)} J = \left[ y(k+1, b) - y^{\text{ref}} \right]^2 + \gamma \left\| G(k, b) - G(k - 1, b) \right\|^2,
$$

(15)

s.t. (2)

$$
y(k+1, b) = y(k+1, b-1) + \hat{\phi}^T(k, b) \Delta G(k, b),
$$

(16)

$$
\hat{\phi}(k, b) = \hat{\phi}(k, b-1) + \eta \Delta G(k, b-1) \left( \frac{\eta \Delta G(k, b-1)}{\mu + \| \Delta G(k, b-1) \|^2} \right)
\times \left( \Delta y(k+1, b-1) - \hat{\phi}^T(k, b-1) \Delta G(k, b-1) \right),
$$

(17)

where (16) is the MFAILC data model of $y(k, b)$ which has been elaborated in (11); (17) is utilized to determine the pseudo-gradient of the traffic system.

It is easy to be verified that optimization problem (2), (15)–(17) is a convex quadratic programming (QP) problem, which can be solved efficiently by the QP solver. Compared with the nonlinear optimization problems, a QP problem needs much less computation time. Furthermore, one can see from (2), (15)–(17) that the information used to compute $y(k+1, b)$ and $\hat{\phi}(k, b)$ at time instant $k$, iteration $b$ is the historical data which can be directly measured by the traffic system, that is, $G(k, b-1), G(k, b-2), y(k+1, b-1), y(k+1, b-2)$ and $\hat{\phi}(k, b-1)$.

### 4 TRAFFIC DATA DROPOUT COMPENSATION METHOD

In this section, a traffic data dropout compensation algorithm is proposed for the MFAILC traffic controller, which is utilized to attenuate the effect of traffic data dropout on the control performance of the traffic controller caused by the failure of traffic detectors. For practical urban traffic control, if there is no data displayed on the traffic detector, or the value of the detected traffic data is abnormal, we say that the traffic data is dropped.

According to traffic control problem (2), (15)–(17), $y(k+1, b-1)$ is needed when we calculate $\hat{\phi}(k, b)$ and $y(k+1, b)$ at time instant $k$, iteration $b$. If $y(k+1, b-1)$ is dropped, the control performance of the traffic controller would be affected, and then the optimization problem (2), (15)–(17) should be modified.

Define

$$
\tilde{y}(k+1, b-1) = \begin{cases} 
y(k+1, b-1), & \alpha = 1, \\
y(\hat{k}+1, b-1), & \alpha = 0,
\end{cases}
$$

(18)

for $\alpha = \begin{cases} 
1, & \text{if } y(k+1, b-1) \text{ is not dropped}, \\
0, & \text{if } y(k+1, b-1) \text{ is dropped},
\end{cases}$

where $\hat{y}(k+1, b-1)$ is the estimation of $y(k+1, b-1)$, and the following estimation equation can be used to determine the value of $\tilde{y}(k+1, b-1)$:

$$
\tilde{y}(k+1, b-1) = \tilde{y}(k+1, b-2) + \phi^T(k, b-1) \Delta G(k, b-1).
$$

(19)

Then, the calculation of $\hat{\phi}(k, b)$ is modified as:

$$
\hat{\phi}(k, b) = \hat{\phi}(k, b-1) + \alpha \frac{\eta \Delta G(k, b-1)}{\mu + \| \Delta G(k, b-1) \|^2}
\times \left( \Delta \tilde{y}(k+1, b-1) - \hat{\phi}^T(k, b-1) \Delta G(k, b-1) \right).
$$

(20)

**Remark 4.** It is noted that $y(k+1, b-1)$ can be compensated by $y(k+1, b-2)$ using (19) if $y(k+1, b-1)$ is dropped; similarly, if $y(k+1, b-2)$ is also dropped, it can be compensated by $y(k+1, b-3)$, and so on. Therefore, for each time instant $k$, iteration $b$, the traffic data compensation method shown by (18)–(20) is available as long as not all the traffic data at the previous iterations are dropped.

Therefore, the MFAILC strategy with data dropout compensation can be designed as follows:

$$
\min_{G(k,b)} J = \left[ y(k+1, b) - y^{\text{ref}} \right]^2 + \gamma \left\| G(k, b) - G(k - 1, b) \right\|^2,
$$

(21)

s.t. (2)

$$
y(k+1, b) = \tilde{y}(k+1, b-1) + \hat{\phi}^T(k, b) \Delta G(k, b),
$$

(22)

$$
\hat{\phi}(k, b) = \hat{\phi}(k, b-1) + \alpha \frac{\eta \Delta G(k, b-1)}{\mu + \| \Delta G(k, b-1) \|^2}
\times \left( \Delta \tilde{y}(k+1, b-1) - \hat{\phi}^T(k, b-1) \Delta G(k, b-1) \right),
$$

(23)

$$
\tilde{y}(k+1, b-1) = \begin{cases} 
y(k+1, b-1), & \alpha = 1, \\
y(\hat{k}+1, b-1), & \alpha = 0 \end{cases}
$$

(24)

$$
\tilde{y}(k+1, b-1) = \tilde{y}(k+1, b-2) + \phi^T(k, b-1) \Delta G(k, b-1).
$$

(25)

It can be seen that optimization problem (2), (21)–(25) is still a convex QP problem after introducing the data dropout compensation mechanism compared with problem (2), (15)–(17).

### 5 SIMULATION

#### 5.1 Traffic network and simulation settings

To assess the proposed traffic control strategy, the traffic network and the real traffic data from Linfen, Shanxi province,
China are used in this simulation. The network is represented in Figure 3, and it is composed of 23 intersections, 102 links, 16 vehicle inflow points $V_1 - V_{16}$, and 86 signal phases. The network is simulated using VISSIM [50] with control algorithms programmed in MATLAB.

The performance of the following five control strategies is compared in the case study:

1. **Fixed-time control (FT)**
   As a benchmark strategy for comparison, the actual fixed-time signal settings in the peak hours of Linfen city is applied in the simulation, which is tuned based on off-line traffic data using the well-known Webster method [1].

2. **Model-based urban traffic control (MB-UTC)**
   In order to illustrate the data-driven feature of the proposed control method, the model-based traffic control strategy proposed in [6] is applied in simulation for comparison. In this strategy, the traffic network is controlled by a model-based traffic controller that optimizes the signal settings in the network dynamically at each time instant $k$ with S model [6, 46] being utilized as the mathematical model of the traffic controller.

3. **MFAC-based urban traffic control (MFAC-UTC)**
   To verify the iterative learning ability of the MFAILC traffic control strategy, a MFAC-based traffic control method is utilized as a benchmark strategy. The detailed process of this control method comes from the inner level control framework of the traffic controller designed in [19], where the MFAC scheme is applied to derive the detailed signal settings in a network region at each time instant $k$ aiming to balance the traffic flow distribution. Different from MFAILC, the MFAC scheme linearizes the traffic dynamics along the time axis instead of iteration axis, thus it has no iterative learning ability.

4. **Pure MFAILC strategy (P-MFAILC)**
   To address the importance of the traffic data compensation mechanism proposed in Section 4, the MFAILC-based urban traffic control strategy without compensation for traffic data dropout is introduced as a benchmark strategy. That is, the optimization problem (2), (15)–(17) is utilized by the traffic controller at each time instant $k$, iteration $h$.

5. **MFAILC strategy with data dropout compensation (MFAILC-DDC)**
   Compared with the P-MFAILC strategy, traffic data dropout compensation method proposed in Section 4 is adopted. Specifically, the optimization problem (2), (21)–(25) is utilized by the traffic controller at each time instant $k$, iteration $h$.

Under all control strategies, the initial traffic condition (i.e. vehicle distribution, signal settings etc.) of the network is the same to ensure the fairness of the simulation. Based on trial-and-error experiments carried out for this particular problem settings, the weighting factor in (15) and (21) is set as $\gamma = 0.1$. For the parameters of the MFAILC data models for the P-MFAILC and the MFAILC-DDC strategies, we have selected $\eta = 0.6, \mu = 0.01$ which are used in (12), (13), (17), (20) and (23), respectively. The total number of iterations for the P-MFAILC and the MFAILC-DDC strategies is set as 20, and the traffic signal settings for the first iteration of these three strategies are set to be the same signal setting plan as the FT strategy.

The sampling interval, control interval, and common cycle length of all intersections are set as $T = 120s$. In (2), the minimum green time is set as 10 s for all signal phases, and the lost time is set as 8 s for all intersections. The total simulation time period is 12000 s, which corresponds to $K = 100$ time instants during the simulation period (for FT, MB-UTC and MFAC-UTC) or during an iteration process (for P-MFAILC and MFAILC-DDC). The simulation period corresponds to the evening rush hour of Linfen, Shanxi province, China. The traffic demand profile of the network provided by the Linfen traffic management bureau is listed in Table 2.

### Table 2: Traffic demand of the network (in veh/h)

| Vehicle inflow points | Period of time (s) |
|-----------------------|--------------------|
|                       | 0–1200  | 1200–6000 | 6000–9600 |
| 1                     | 800     | 1200      | 600       |
| 2                     | 650     | 1100      | 460       |
| 3                     | 900     | 1500      | 700       |
| 4                     | 800     | 1350      | 600       |
| 5                     | 840     | 1400      | 660       |
| 6                     | 850     | 1500      | 680       |
| 7                     | 480     | 800       | 320       |
| 8                     | 480     | 800       | 320       |
| 9                     | 540     | 900       | 360       |
| 10                    | 450     | 740       | 288       |
| 11                    | 450     | 750       | 300       |
| 12                    | 510     | 850       | 340       |
| 13                    | 540     | 900       | 360       |
| 14                    | 900     | 1500      | 600       |
| 15                    | 900     | 1500      | 600       |
| 16                    | 540     | 900       | 360       |
Then, in order to compare the control performance under different strategies, the following evaluation criteria are utilized:

1. The total time spent $TTS(k)$ at time instant $k$ is the cumulative time that all vehicles spent in the traffic network, which is described as

$$TTS(k) = \sum_{m=1}^{k} T \cdot y(m).$$  \hspace{20pt} (26)

2. The average flow rate $AFR(k)$ is defined as average traffic flow among the links in the network during time interval which is expressed as

$$AFR(k) = \frac{\sum_{(j,i) \in \mathcal{E}} \ell_{(j,i)}(k)}{T \cdot |\mathcal{Z}|}. \hspace{20pt} (27)$$

3. The average value of $AFR(k)$ during the simulation period (for FT, MB-UTC and MFAC-UTC), or during an iteration process (for MFAILC-IIC, MFAILC-NC and MFAILC-DDC), which is expressed as

$$AFR_{avg} = \frac{\sum_{k=1}^{K} AFR(k)}{K}. \hspace{20pt} (28)$$

To obtain the value of $y^{ref}$ off-line in this work, the MFD of the traffic network is needed. In this case study, we use a fifth-order polynomial function to obtain the unimodal MFD of the network, that is, the relationship between the number of vehicles and average flow rate of the network:

$$AFR(k) = a_1 y^{5}(k) + a_2 y^{4}(k) + a_3 y^{3}(k) + a_4 y^{2}(k) + a_5 y(k) + a_6,$$  \hspace{20pt} (29)

where $a_1, \ldots, a_6$ are the parameters to be estimated.

The MFD of the network is depicted in Figure 4, which is obtained under fixed-time control tuned using Webster’s method. Based on the ‘polyfit’ function in MATLAB, we get the parameters in (29) as follows: $a_1 = 1.721 \times 10^{-20}$, $a_2 = -8.138 \times 10^{-16}$, $a_3 = 1.412 \times 10^{-11}$, $a_4 = -1.109 \times 10^{-7}$, $a_5 = 3.681 \times 10^{-4}$, $a_6 = -0.027$ and $y^{ref} = 3063$ veh.

It is noted that, the MFD in this work is not a necessary. In fact, we can use other methods to get the $y^{ref}$, such as, the experimental setting from a traffic engineer.

5.2 | Performance evaluations of different control strategies in the case of no traffic data dropout

In this subsection, the control performance of the traffic network under different control strategies is compared in the case of no traffic data dropout. In this condition, the P-MFAILC and the MFAILC-DDC strategies are the same, thus we only use MFAILC-DDC to represent both two strategies in the following.

The evolutions of $TTS(k)$ and $AFR(k)$ of the traffic network are depicted in Figure 5, and the results of $TTS(100)$ and $AFR_{avg}$ are listed in Table 3. The learning process and control performance improvement over iterations of the MFAILC-DDC strategy is shown in Figure 6 and Table 3. It should be pointed out that the control performance of MFAILC-DDC of its 20th iteration is selected as the simulation results presented in Figure 5.

From Figure 5 and Table 3, we can see that the MB-UTC strategy yields lower $TTS$ and higher $AFR$ than FT as the signal settings in the traffic network can be optimized to adapt to the real-time traffic condition. Compared with the MB-UTC strategy, the MFAC-UTC strategy leads to better control performance by directly utilizing real-time measured traffic data instead of a mathematical traffic model, which can avoid the problem of model mismatch and traffic uncertainties in the control process. Finally, MFAILC-DDC can further improve the control performance in comparison with MFAC-UTC due to its iterative learning ability by taking advantage of the similarity and the repeatability of urban traffic flow.

Next, it is noteworthy from Table 3 and Figure 6 that FT and the first iteration of the MFAILC-DDC strategy have exactly the same control performance as their signal settings in the network are the same. Then, it can be seen that the proposed MFAILC-DDC strategy can gradually decrease the $TTS(100)$ and increase the $AFR_{avg}$ over the iterations by learning from the previous executions.

In conclusion, the proposed MFAILC-DDC strategy outperforms the FT, the MB-UTC and the MFAC-UTC strategies under the case of no traffic data dropout due to its data-driven feature and iterative learning ability.

5.3 | Performance evaluations of P-MFAILC and MFAILC-DDC in the case of traffic data dropout

In the simulation case of traffic data dropout, we suppose the probability of $y(k,b)$ being dropped is 20% at each time instant $k$, iteration $b$. For the MB-UTC and MFAC-UTC strategies, we
set $y(k) = y(k-1)$ if $y(k)$ is dropped; while for the P-MFAILC strategy, we compensate $y(k, h)$ by setting $y(k, h) = y(k, h-1)$. The simulation results of the control strategies in the case of data dropout are presented in Figure 7 and Table 4.

From Figures 6 and 7 and Tables 3 and 4, we can see that all control strategies except FT give worse control performance than in the case of no data dropout during the control process, as the traffic controller is affected by the dropped traffic data. Nevertheless, it can be seen from Figure 8 and Table 4 that MFAILC-DDC performs better than P-MFAILC comparatively, that is, TTS(100) decreases, and AFR$_{avg}$ increases faster over the iterations under the MFAILC-DDC strategy than under the other one. Therefore, the MFAILC-DDC strategy shows stronger robustness and iterative learning ability over P-MFAILC, because the impact of dropped traffic data on the control performance can be attenuated by utilizing the data dropout compensation method proposed in Section 4.

### 5.4 Comparison of computation times

The computation times under different control strategies within the simulation period are listed in Table 5. These computation times are obtained using the ‘etime’ function of MATLAB. The computation times of the P-MFAILC and the MFAILC-DDC strategies are obtained by averaging the values of total 20 iterations.

It can be seen from Table 5 that the MFAC-UTC, the P-MFAILC, and the MFAILC-DDC strategies only need around 10 s (corresponds to only 0.1 s average per step) in the
FIGURE 7 Comparison of total time spent and average flow rate of the network under different control strategies in the case of traffic data dropout. (a) Total time spent. (b) Average flow rate

FIGURE 8 TTS(100) and AFR\textsubscript{avg} over iterations under P-MFAILC and MFAILC-DDC in the case of traffic data dropout. (a) TTS(100). (b) AFR\textsubscript{avg}.

TABLE 4 Simulation results of TTS(100) and AFR\textsubscript{avg} under different control strategies in the case of traffic data dropout

| Control strategy | Estimation criteria | TTS(100)       | AFR\textsubscript{avg} |
|------------------|---------------------|-----------------|--------------------------|
| FT               |                      | 1.9886 × 10\(^8\) | 0.3023                   |
| MB-UTC           |                      | 1.8918 × 10\(^8\) | 0.3162                   |
| MFAC-UTC         |                      | 1.8257 × 10\(^8\) | 0.3231                   |
| P-MFAILC         | 1st iteration       | 1.9886 × 10\(^8\) | 0.3023                   |
|                  | 2nd iteration       | 1.9619 × 10\(^8\) | 0.3066                   |
|                  | 5th iteration       | 1.8814 × 10\(^8\) | 0.3175                   |
|                  | 10th iteration      | 1.7912 × 10\(^8\) | 0.3295                   |
|                  | 15th iteration      | 1.7361 × 10\(^8\) | 0.3377                   |
|                  | 20th iteration      | 1.7193 × 10\(^8\) | 0.3408                   |
| MFAILC-DDC       | 1st iteration       | 1.9886 × 10\(^8\) | 0.3023                   |
|                  | 2nd iteration       | 1.9573 × 10\(^8\) | 0.3067                   |
|                  | 5th iteration       | 1.8762 × 10\(^8\) | 0.3196                   |
|                  | 10th iteration      | 1.7737 × 10\(^8\) | 0.3356                   |
|                  | 15th iteration      | 1.7169 × 10\(^8\) | 0.3424                   |
|                  | 20th iteration      | 1.6958 × 10\(^8\) | 0.3440                   |

6 CONCLUSIONS AND FUTURE WORKS

In this paper, the urban traffic control problem is addressed by a novel data-driven method called MFAILC. By dynamically simulation or in an iteration process; while the MB-UTC strategy needs 286s (2.86s average per step), which is far larger than the former three strategies. This is because a nonlinear and non-convex optimization problem needs to be solved at each time instant \( k \) for the MB-UTC, while a convex quadratic optimization problem is just needed for the other three strategies.

TABLE 5 Computation times of different control strategies

| Control strategy | Total computation time (s) |
|------------------|---------------------------|
| FT               | -                         |
| MB-UTC           | 286                       |
| MFAC-UTC         | 9.8                       |
| P-MFAILC         | 10.1                      |
| MFAILC-DDC       | 10.3                      |
linearizing the urban traffic dynamics along the iteration axis, the traffic network system is transformed into a MISO CFDL-MFAILC data model. With the help of the I/O traffic data, the derived MFAILC data model can be used in the controller design without the need of a mathematical traffic model. The resulting controller can gradually improve the control performance by learning from previous executions due to its iterative learning ability. After that, a MFAILC-based traffic data compensation method is designed to deal with the data dropout problem in the traffic networks.

Simulation results of the traffic network of Linfen, Shanxi province, China show that the proposed MFAILC-DDC method yields better control performance than the FT, the MB-UTC and the MFAC-UTC strategies, and can attenuate the performance deterioration caused by traffic data dropout compared with the P-MFAILC strategy. Moreover, the proposed MFAILC-DDC strategy needs much less computation time than the MB-UTC strategy in the control process.

Our future researches will be on the rigorous proof of convergence and robustness of the proposed MFAILC-DDC strategy, and the investigation of more complicated simulation cases, such as existence of measurement noise, iteration-varying traffic demands and different data drop rates etc.

FUNDING
This work was supported by the National Natural Science Foundation of China under Grant Nos. 61433002 and 61833001.

PERMISSION STATEMENT TO REPRODUCE THE MATERIALS FROM THE OTHER SOURCES
None

ORCID
Dai Li https://orcid.org/0000-0001-6763-0128
Zhongsheng Hou https://orcid.org/0000-0001-5278-3420

REFERENCES
1. Webster, F.V.: Traffic signal settings. Road Research Technical Paper (39), (1958)
2. Little, J.D.C., et al.: MAXBAND: a program for setting signals on arteries and triangular networks. Transp. Res. Rec., J. Transp. Res. Board 795, 40–46 (2001)
3. Hale, D.L.: Traffic Network Study Tool–TRANSYT-7F. McTrans Center of University of Florida (2005)
4. Diakaki, C., et al.: A multivariable regulator approach to traffic-responsive network-wide signal control. Control Eng. Pract. 10(2), 183–195 (2002)
5. Aboudolas, K., et al.: Store-and-forward based methods for the signal control problem in large-scale congested urban road networks. Transp. Res. Part C, Emerg. Technol. 17(2), 163–174 (2009)
6. Lin, S., et al.: Efficient network-wide model-based predictive control for urban traffic networks. Transp. Res. Part C, Emerg. Technol. 24(9), 122–140 (2012)
7. Zhou, Z., et al.: Multi-agent model-based predictive control for large-scale urban traffic networks using a serial scheme. IET Control Theory Appl. 9(3), 475–484 (2014)
8. Zhou, Z., et al.: Two-level hierarchical model-based predictive control for large-scale urban traffic networks. IEEE Trans. Control Syst. Technol. 25(2), 496–508 (2017)
9. Godfrey, J.W.: The mechanism of a road network. Traffic Eng. Control 11(7), 325–327 (1969)
10. Geroliminis, N., Daganzo, C.F.: Existence of urban-scale macroscopic fundamental diagrams: Some experimental findings. Transp. Res. Part B, Methodol. 42(9), 759–770 (2008)
11. Aboudolas, K., Geroliminis, N.: Perimeter and boundary flow control in multi-reservoir heterogeneous networks. Transp. Res. Part B, Methodol. 55(9), 265–281 (2013)
12. Haddad, J., Mirkin, B.: Coordinated distributed adaptive perimeter control for large-scale urban road networks. Transp. Res. Part C, Emerg. Technol. 77, 495–515 (2017)
13. Wang, Y., et al.: Perimeter control of multiregion urban traffic networks with time-varying delays. IEEE Trans. Syst., Man, Cybern. 50(8), 2795–2803 (2020)
14. Aparntoulas, K., et al.: Macroscopic modelling and robust control of bi-modal multi-region urban road networks. Transp. Res. B, Methodol. 104, 616–637 (2017)
15. Mohajerpoor, R., et al.: Haib robust perimeter flow control in urban networks with partial information feedback. Transp. Res. B, Methodol. 137, 47–73 (2019)
16. Sirmatel, I.I., Geroliminis, N.: Economic model predictive control of large-scale urban road networks via perimeter control and regional route guidance. IEEE Trans. Intell. Transp. Syst. 19(4), 1112–1121 (2018)
17. Sirmatel, I.I., Geroliminis, N.: Nonlinear moving horizon estimation for large-scale urban road networks. IEEE Trans. Intell. Transp. Syst. 21(12), 4983–4994 (2019). https://doi.org/10.1109/TITS.2019.2946324
18. Lei, T., et al.: Data-driven model free adaptive perimeter control for multi-region urban traffic networks with route choice. IEEE Trans. Intell. Transp. Syst. 21(7), 2894–2905 (2019)
19. Li, D., Hou, Z.S.: Perimeter control of urban traffic networks based on model-free adaptive control. IEEE Trans. Intell. Transp. Syst. (2020). https://doi.org/10.1109/TITS.2020.2992337
20. Hou, Z.S.: The parameter identification, adaptive control and model free learning adaptive control for nonlinear systems (In Chinese). Ph.D. thesis, Northeastern University, Shenyang, China (1994)
21. Hou, Z.S., Jin, S.T.: Model Free Adaptive Control: Theory and Applications. CRC Press, Boca Raton (2013)
22. Hou, Z.S., et al.: An overview of dynamic-linearization-based data-driven control and applications. IEEE Trans. Industrial Electronics 64(5), 4076–4090 (2017)
23. Hou, Z.S., Xiong, S.S.: On model free adaptive control and its stability analysis. IEEE Trans. Automat. Control 64(41), 4555–4569 (2019)
24. Ma, Y., et al.: Data-driven measurement of receiver sensitivity in wireless communication systems. IEEE Trans. Commun. 67(5), 3665–3676 (2019)
25. Fetanat, M., et al.: A physiological control system for an implantable heart pump that accommodates for interpatient and intrapatient variations. IEEE Trans. Biomed. Eng. 67(4), 1167–1175 (2020)
26. Zhu, Y.M., et al.: Dual RBFNNs-based model-free adaptive control with aspen HYSYS simulation. IEEE Trans. Neural Netw. Learn. Syst. 28(3), 759–765 (2017)
27. Hou, Z.S., Li, X.Y.: Repeatability and similarity of freeway traffic flow and long-term prediction under big data. IEEE Trans. Intell. Transp. Syst. 17(6), 1786–1796 (2016)
28. Hou, Z.S., et al.: An iterative learning approach for density control of freeway traffic flow via ramp metering. Transp. Res. Part C, Emerg. Technol. 16(1), 71–97 (2008)
29. Yan, F., et al.: Iterative learning approach for traffic signal control of urban road networks. IET Control Theory Appl. 11(4), 466–475 (2017)
30. Hou, Z.S., et al.: Freeway traffic control using iterative learning control-based ramp metering and speed signaling. IEEE Trans. Veh. Technol. 56(2), 466–477 (2007)
31. Arimoto, S., et al.: Bettering operation of robots by learning. J. Field Robot. 1(2), 123–140 (2010)
APPENDIX A: PROOF OF THEOREM 1
From the definition of $\Delta y(k, b)$ and (9), we have

$$
\Delta y(k+1, b) = \Delta \bar{y}(y(k, b), \ldots, y(k-m, b), G(k, b), \ldots, G(k-m, b))
$$

$$
= \Delta \bar{y}(y(k, b), \ldots, y(k-m, b), G(k, b-1), \ldots, G(k-m, b-1))
$$

Then, we denote

$$
\tau(k, b) = \Delta \bar{y}(y(k, b), \ldots, y(k-m, b), G(k, b-1), \ldots, G(k-m, b-1))
$$

According to the differential mean value theorem and Assumption 1, (A.1) can be rewritten as

$$
\Delta y(k+1, b) = \Delta \bar{y}(y(k, b), \ldots, y(k-m, b), G(k, b), \ldots, G(k-m, b)) + \tau(k, b)
$$

where $(\partial \bar{y}^{*}) / (\partial G(k, b))$ denotes the partial derivative of $\bar{y}^{*}()$ with respect to $G(k, b)$ at a certain point between

$$
[y(k, b), \ldots, y(k-m, b), G^{T}(k, b), \ldots, G^{T}(k-m, b)]^{T}
$$

and

$$
[y(k, b-1), \ldots, y(k-m, b), G^{T}(k, b-1), \ldots, G^{T}(k-m, b)]^{T}
$$

For every fixed time instant $k$ and fixed iteration $b$, consider the following equation with a vector $\beta(k, b)$:

$$
\tau(k, b) = \beta^{T}(k, b)\Delta G(k, b).
$$

Since $\| \Delta G(k, b) \| \neq 0$, there must exist at least one solution $\beta^{*}(k, b)$ to (A.4).

Let $\bar{\phi}(k, b) = (\partial \bar{y}^{*}) / (\partial G(k, b)) + \beta^{*}(k, b)$. Then, (A.3) can be rewritten as $\Delta y(k+1, b) = \bar{\phi}^{T}(k, b)\Delta G(k, b)$, which is the CFDL-MFAILC data model of the traffic system (11). The boundedness of $\bar{\phi}(k, b)$ is guaranteed directly by Assumption 2.