Urban Road Traffic Light Real-Time Scheduling

Yicheng Zhang, Rong Su and Kaizhou Gao

Abstract—This paper addresses urban traffic signal control in a scheduling framework, where the dynamics of an urban traffic network controlled by traffic lights is described by a novel model, which inserts mixed logical constraints into a cell transmission flow dynamic model, capable of capturing the nonlinear relationship between each outgoing link flow rate and the corresponding upstream and downstream link capacities and the past traffic light signals. With a control goal of minimizing the total network-wise delay time, we translate the traffic signal control problem into a centralized mixed integer linear programming problem solvable by several existing tools, e.g., CPLEX. To overcome the potentially high complexity involved in the centralized approach, we propose a distributed scheduling strategy based on Lagrangian relaxation and subgradient method. Simulation results are provided to demonstrate the effectiveness of the proposed traffic light scheduling approach.

Index Terms—macroscopic traffic flow model, mixed logical constraints, mixed integer linear programming, Lagrangian relaxation, subgradient method, genetic algorithm

I. INTRODUCTION

The importance of effective urban traffic signal control can never be underestimated due to the exponentially increasing traffic demands for economic development, which has been significantly constrained by increasingly saturated space. An urban traffic network consists of a set of road links connecting with each other via intersections. Each intersection consists of a number of approaches and the crossing area. An approach may have one or more lanes but has a unique, independent queue. Approaches are used by corresponding traffic streams (veh/h). Two compatible streams can safely cross the intersection simultaneously, while antagonistic streams cannot. There are basically four types of characteristics that distinguish different traffic signal control, i.e., fixed time strategies versus traffic responsive strategies, and isolated strategies versus coordinated strategies. Isolated fixed-time strategies, e.g., [5][6], are only applicable to under-saturated traffic conditions. Isolated traffic-responsive strategies, see, e.g., [7], make use of real-time measurements provided by inductive loop detectors to execute some more or less sophisticated vehicle-actuation logic. Fixed time coordinated strategies have been widely used in practice, e.g., [8][9], which are usually ineffective in coping with highly dynamic traffic situations. Coordinated traffic-responsive strategies have recently gained significant supports. Typical works in this direction include SCOOT [10], Model-Based Optimization Methods, see e.g., CRONOS [11], RHODES [12], which do not consider explicitly splits, offsets, or cycles, but rather calculate in real time the optimal values of the next few switching times based on realistic traffic models with a sampling time of 2-5 seconds over a future time horizon; control theoretical approaches, e.g., the usage of the store-and-forward modelling of traffic networks with a LQR optimal control solution [14].

In this paper we formulate the urban road traffic light control problem as a scheduling problem, aiming to reduce the total waiting time over a given finite horizon. A traffic network is described by a flow dynamic model similar to the Daganzo’s cell transmission model [15]. Our key contribution in this model is to describe each outgoing flow rate as a nonlinear mixed logical switching function over the source link’s density, the destination link’s density and capacity, and the driver’s potential psychological response to the past traffic light signals. This outgoing flow rate model makes our approach applicable to both under-saturated and over-saturated situations. Each traffic light has two signals: GREEN and RED, which essentially specify which traffic streams should not be allowed simultaneously. The traditional concepts of cycles, splits and offsets are not adopted in this framework, making our approach fall in the class of model-based optimization methods, where each traffic light is assigned with a green light period in a real-time manner by the network controller. We assume that the traffic demands at the boundary of the concerned network are known in advance, and the turning ratios in each link are also known. These assumptions will be relaxed in our future research by using real-time data-driven time series regression and forecast. The whole problem is converted into a centralized mixed integer linear program by using techniques introduced in [1], which is solved by using standard optimization tools such as CPLEX [4]. To overcome the high computational complexity involved in the centralized strategy, we propose a distributed strategy based on Lagrangian relaxation and subgradient method. It turns out that application of Lagrangian relaxation in traffic signal scheduling also renders feasible traffic light assignments, even though some boundary flow consistency constraints may not hold. Numerical experiments are presented to illustrate the effectiveness of both centralized and distributed scheduling strategies.

The rest of the paper is organized as follow. A novel formulation of a centralized traffic light scheduling problem is presented in Section II, and a distributed scheduling strategy is described in Section III. Conclusions are drawn in Section IV.
II. CENTRALIZED TRAFFIC LIGHT SCHEDULING

A. A traffic network model

A traffic network consists of a set of road links and intersections (or junctions). Fig. 1 depicts a simple unidirectional traffic network, where each intersection has only two antagonistic traffic streams (or flows). We adopt a discrete-time model similar to a cell transmission model. Let \( \Delta \) be the sampling interval, e.g., \( \Delta = 5 \text{s} \). We use \( C_i(k) \) to denote the number of vehicles (or volume) in link \( i \) in time interval \( k \), and \( f_{ij}(k) \geq 0 \) for the exit flow rate from link \( i \) to link \( j \) in interval \( k \). Let \( L \) be the set of all one-way links, and \( F \) the set of all intersections. For each intersection \( J \in F \), let \( \Omega_J \) be the set of stages in intersection \( J \), and \( F_J \subseteq L \times L \) the set of all streams in intersection \( J \), i.e., \( (i,j) \in F_J \) means that there exists a traffic stream from link \( i \) to link \( j \) via intersection \( J \). Let \( h_j : \Omega_J \rightarrow 2^{F_J} \) be the association of each stage to relevant compatible streams. Here, for simplicity, we assume that for any two different stages \( w_i \) and \( w_j \), \( h_j(w_i) \cap h_j(w_j) = \emptyset \). We make the following assumptions about a traffic network under consideration, which is suitable for a deterministic analysis:

- **A1**: No traffic demand is generated inside the network.
- **A2**: The network boundary demand (or entrance) and service (or exit) models are known.
- **A3**: The link turning ratios in the network are known.
- **A4**: Each vehicle inside the network will leave the network, delayed only by traffic signals.
- **A5**: The vehicle speed in each link has a finite number of known values.

In each given interval \( k \), there exists only one active stage for an intersection \( J \), which is captured by the following constraints:

\[
\begin{align*}
\forall w \in \Omega_J \quad \theta_w(k) = 0 & \quad \Rightarrow \quad \forall (i,j) \in h_j(w) \quad f_{ij}(k) = 0 \quad (1a) \\
\sum_{w \in \Omega_J} \theta_w(k) &= 1 \quad \forall (i,j) \quad (1b) \\
\forall w \in \Omega_J \quad \forall k \in N \quad \theta_w(k) \in \{0,1\} & \quad (1c)
\end{align*}
\]

where \( \theta_w(k) = 0 \) and \( \theta_w(k) = 1 \) denote the RED and GREEN traffic lights associated with stage \( w \) respectively, and \( N \) denotes the set of natural numbers. Condition (1a) indicates that if the stage traffic light is RED, then all relevant flow rates are zero. Condition (1b) indicates that there can be only one GREEN traffic stage at any time.

Due to the conservation of vehicles, each link \( j \in L \) has the following volume dynamics:

\[
\begin{align*}
C_j(k+1) &= C_j(k) + \Delta(d_j(k) - s_j(k)) \quad (2a) \\
(\forall k \in N) \quad C_j(k) &\in N \quad (2b)
\end{align*}
\]

where \( d_j(k) \) and \( s_j(k) \) are entrance and exit flow rates of link \( j \) in interval \( k \) respectively, i.e.,

\[
\begin{align*}
d_j(k) &= \sum_{i \in L_j} f_{ij}(k) \quad (6a) \\
f_{ij}(k) &\leq \lambda_{ij}(k)(\hat{C}_j - C_j(k)) \quad (6b) \\
&\quad [f_{ij}(k) + 1 \geq \lambda_{ij}(k)(\hat{C}_j - C_j(k)) + \epsilon] \quad (6c)
\end{align*}
\]

where \( g_{ij}(\cdot) \) is a nonlinear function. The motivation behind this model is that if stage \( w \) has been active for the past \( r \) intervals, then the drivers intend to keep a high speed as long as the downstream link has sufficient capacity to receive the flow. There are perhaps many ways to define \( g_{ij}(\cdot) \). The following is one potential definition. Assume that there are \( r+1 \) monotonically non-increasing speed categories: \( \ell_0 \geq \cdots \geq \ell_2 \geq 0 \), which denotes speed ranges from high to low.

Let \( [0,r-1] = 0, \ldots , r-1 \). The actual speed category \( l_{ij}(k) \) is determined as follows:

\[
l_{ij}(k) = \sum_{p=0}^{r} \delta_{ij}^p(k) \ell_p \quad (4a)
\]

\[
\sum_{p=0}^{r} \delta_{ij}^p(k) - \theta_w(k) = 0 \quad (4b)
\]

\[
(\forall i,j \in J, k \geq 0) \quad (1 - \theta_w(k-q-1)) \prod_{p=0}^{q} \theta_w(k-p) = 1 \quad \Rightarrow \quad \delta_{ij}^q(k) = 1 \quad (4c)
\]

\[
\prod_{p=0}^{r} \theta_w(k-p) = 1 \quad \Rightarrow \quad \delta_{ij}^r(k) = 1 \quad (4d)
\]

\[
(\forall p \in [0,r]) \quad \delta_{ij}^p \in \{0,1\} \quad (4e)
\]

Condition (4b) indicates that if the traffic light of stage \( w \) is RED, i.e., \( \theta_w(k) = 0 \), then \( \sum_{p=0}^{r} \delta_{ij}^p(k) = 0 \), which by Condition (4a) means \( l_{ij}(k) = 0 \), i.e., the speed in the stage \( w \) must be zero; if the traffic light of stage \( w \) is GREEN, i.e., \( \theta_w(k) = 1 \), then by Condition (4a), \( l_{ij}(k) \) can only choose one speed category because of \( \sum_{p=0}^{r} \delta_{ij}^p(k) = 1 \). Conditions (4c)-(4d) indicate that the actual speed category depends on the number of consecutive green light intervals from \( k \) backward in time - the larger the number of consecutive green intervals, the higher the speed category. If \( l_{ij}(k) \) is determined, the link flow rate \( f_{ij}(k) \) is given as follows:

\[
f_{ij}(k) = \min \{ \lambda_{ij}(k) C_i(k), l_{ij}(k)(\hat{C}_j - C_j(k)) \} \quad (5a)
\]

\[
f_{ij}(k) \in N \quad (5b)
\]

where \( \lfloor \cdot \rfloor \) is the largest integer not greater than input argument, and \( \lambda_{ij}(k) \) is the turning ratio of vehicles in link \( i \) towards link \( j \) at \( k \), which is assumed to be known in advance. Clearly, \( \sum_{j \in L_i \cup F_{i,j} \in \Omega_i} \lambda_{ij}(k) = 1 \), meaning that each vehicle in link \( i \) will move into some downstream link \( j \). Condition (5a) can be converted into the following mixed logical constraints:

\[
f_{ij}(k) \leq \lambda_{ij}(k) C_i(k) \quad (6a)
\]

\[
f_{ij}(k) \leq l_{ij}(k)(\hat{C}_j - C_j(k)) \quad (6b)
\]

\[
[f_{ij}(k) + 1 \geq \lambda_{ij}(k) C_i(k) + \epsilon] \quad \lor \quad [f_{ij}(k) + 1 \geq l_{ij}(k)(\hat{C}_j - C_j(k)) + \epsilon] \quad (6c)
\]
where $\epsilon$ is a very small positive real number, e.g., the smallest positive precision for a computer.

The total network-wise delay time within $N$ time intervals can be estimated as follows:

$$\sum_{i \in \mathcal{L}} \sum_{k=1}^{N} C_i(k) \left[ 1 - \frac{\bar{v}_i(k)}{v_{i,max}} \right] \Delta = \sum_{i \in \mathcal{L}} \sum_{k=1}^{N} \left[ C_i(k) - \frac{L_i}{v_{i,max}} s_i(k) \right] \Delta$$

where $\bar{v}_i(k)$ is the average speed of link $i$ in interval $k$, which can be approximated by the ratio of the exit flow rate $s_i(k)$ and the average link density $C_i(k)/L_i$, where $L_i$ is the length of the link $i$. Here, we assume that vehicles in link $i$ are uniformly distributed with identical speeds, i.e., the acceleration step is negligible. In reality, this assumption usually does not hold. But for the scheduling purpose, this is a sufficiently representative performance index. Since $s_i(k) = \sum_{j \in \mathcal{L}:(i,j) \in \mathcal{J} \in \mathcal{F}_j} f_{ij}(k)$, we have our cost function as

$$\min \sum_{i \in \mathcal{L}} \sum_{k=1}^{N} \left[ C_i(k) - \frac{L_i}{v_{i,max}} \right] \sum_{j \in \mathcal{L}:(i,j) \in \mathcal{J} \in \mathcal{F}_j} f_{ij}(k) \Delta.$$  

**B. Conversion of mixed logical constraints**

In the above description we have derived a linear cost function with a set of constraints describing traffic staging (1a)-(1c), link volume dynamics (2a)-(2b) and exit flow rates (4a)-(4e) and (5b), (6a)-(6c). Among these constraints, (1a), (4c) and (4d) are mixed logical constraints, which can be converted into mixed integer linear constraints by using the following transformation.

Let $\bar{M}$ be chosen to be sufficiently big, e.g., $\bar{M} > \max_{i \in \mathcal{L}} C_i$. Then (1a) can be converted into

$$(\forall w \in \Omega_j)(\forall (i,j) \in h_j(w)) f_{ij}(k) \leq \bar{M} \theta_w(k). \quad (7)$$

**Proposition 1:** Replacing Condition (1a) with Condition (7) in the urban network traffic signal scheduling formulation leads to the same solution. □

Condition (4c) and (4d) are equivalent to the following: for all $q \in [0, r - 1]$,

$$- (1 - \theta_w(k - q - 1)) + \delta_{ij}^{l-q}(k) \leq 0 \quad (8a)$$

$$\langle \forall p \in [0, q] \rangle - \theta_w(k - p) + \delta_{ij}^{l+q}(k) \leq 0 \quad (8b)$$

$$\langle \forall p \in [0, q] \rangle - \theta_w(k - p) - \delta_{ij}^{r+q}(k) \leq q + 1 \quad (8c)$$

$$\sum_{p=0}^{r} \theta_w(k - p) - \delta_{ij}^{l-0}(k) \leq r \quad (8d)$$

**Proposition 2:** Replacing Condition (4c) and (4d) with Condition (8a)-(8e) in the urban network traffic signal scheduling formulation leads to the same solution. □

Condition (6b) can be rewritten in the following mixed logical constraint: for all $q \in [0, r - 1]$,

$$\delta_{ij}^{l-0}(k) = 1 \iff f_{ij}(k) \Delta \leq \delta_{ij}^{l-0}(\hat{C}_j - C_j(k)), \quad (9)$$

where [1] can be converted into the following mixed integer logical constraints: for all $q \in [0, r - 1]$,

$$f_{ij}(k) \Delta - \delta_{ij}^{l-0}(\hat{C}_j - C_j(k)) \leq \bar{M}_q (1 - \delta_{ij}^{l-0}(k)) \quad (10a)$$

$$f_{ij}(k) \Delta - \delta_{ij}^{l-0}(\hat{C}_j - C_j(k)) \geq \epsilon + (\bar{m}_q - \epsilon) \delta_{ij}^{l-0}(k) \quad (10b)$$

where $\bar{M}_q$ is chosen to be sufficiently big, e.g., $\bar{M}_q \geq \max_k (f_{ij}(k) - \delta_{ij}^{l-0}(\hat{C}_j - C_j(k)))$, and $\bar{m}_q$ is chosen to be sufficiently small, e.g., $\bar{m}_q \leq \min_k (f_{ij}(k) - \delta_{ij}^{l-0}(\hat{C}_j - C_j(k)))$. Condition (6c) is equivalent to the mixed integer constraints below:

$$m_1 (1 - \eta_1(k)) - f_{ij}(k) \Delta - 1 + \lambda_1 \eta_1(k) C_i(k) + \epsilon \leq 0 \quad (11a)$$

$$(\forall q \in [0, r]) m_2 (1 - \eta_2(k)) - f_{ij}(k) \Delta - 1 + \lambda_2 \eta_2(k) (\hat{C}_j - C_j(k)) + \epsilon \leq \bar{M}_q (1 - \delta_{ij}^{l-0}(k)) \quad (11b)$$

$$(\forall q \in [0, r]) m_2 (1 - \eta_2(k)) - f_{ij}(k) \Delta - 1 + \lambda_2 \eta_2(k) (\hat{C}_j - C_j(k)) + \epsilon \leq \bar{M}_q (1 - \delta_{ij}^{l-0}(k)) \quad (11c)$$

$$\eta_1(k) + \eta_2(k) \geq 1 \quad (11d)$$

$$\eta_1(k), \eta_2(k) \in \{0, 1\} \quad (11e)$$

where $m_1$ is sufficiently small, e.g., $m_1 = \min_k f_{ij}(k) \Delta - 1 - \lambda_1(k) C_i(k) - \epsilon$, and $m_2$ is sufficiently small, e.g., $m_2 = \min_k f_{ij}(k) \Delta - 1 - \lambda_2(k) (\hat{C}_j - C_j(k)) - \epsilon$. $\bar{M}_q$ is sufficiently big, e.g., $\bar{M}_q \geq \max_k (m_2 (1 - \eta_2(k)) - f_{ij}(k) \Delta - 1 + \lambda_2 \eta_2(k) (\hat{C}_j - C_j(k))) + \epsilon$, and $\bar{m}_q$ is sufficiently small, e.g., $\bar{m}_q \leq \min_k (m_2 (1 - \eta_2(k)) - f_{ij}(k) \Delta - 1 + \lambda_2 \eta_2(k) (\hat{C}_j - C_j(k))) + \epsilon$. □

**Proposition 3:** Replacing Conditions (6b)-(6c) with Conditions (10a)-(10b), (11a)-(11e) in the urban network traffic signal scheduling formulation leads to the same solution. □

In all, the Urban Traffic Signal Scheduling Problem (UTSSP) is described as an MILP problem with a linear cost function and a set of constraints describing traffic staging (1a)-(1c), link volume dynamics (2a)-(2b) and exit flow rates (4a)-(4b), (4e), (5b), (6a)-(6e), (6a), (10a)-(10b) and (11a)-(11e).

**C. A case study with a receding horizon implementation**

In this section we consider a simplified urban traffic network shown in Fig. 2, which has $n_h$ horizontal roads and $n_v$ vertical roads. The total number of intersections is $n_j = n_h n_v$. Vehicles on the horizontal roads only move from left to right, and vehicles on the vertical roads only move from top to bottom. In this model, each intersection has only two stages: the traffic stream from left to right horizontally, and the traffic stream from top to bottom vertically. We consider two speed categories: fast denoted by $l_0$ and slow denoted by $l_1$, i.e., $r = 1$ in the general problem formulation.
TABLE I

<table>
<thead>
<tr>
<th>n_v = n_h</th>
<th>2</th>
<th>5</th>
<th>7</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>N_e</td>
<td>612</td>
<td>3660</td>
<td>7112</td>
<td>14420</td>
</tr>
<tr>
<td>N_con</td>
<td>1528</td>
<td>9385</td>
<td>18333</td>
<td>37320</td>
</tr>
</tbody>
</table>

Denote \( B \) as the set of links which constitute the boundary of the network. As an example, in Fig. 1 link \( i \in B \). The dynamics of \( C_j \) for \( i \in B \) obeys the following update:

\[
C_j(k+1) = C_j(k) + b_k(k) - f_{ij}(k),
\]

(12)

where the entrance flow rate \( b_k(k) \) is assumed to be given in advance. We apply the aforementioned centralized UTSSP formulation in this case study. As an illustration of the potential complexity involved in solving Problem UTSSP, Table I lists the total number of integer and binary decision variables \( N_e \) and the total number of constraints \( N_{con} \) with \( N = 10 \). These numbers indicate that for a large urban traffic network we need to solve a large mixed integer linear programming problem, which requires efficient computational algorithms in order to ensure proper real-time decisions.

The information required by solving UTSSP at time \( k \) includes the current link volume \( C_j(k) \), the past traffic light status \( \theta_i, (k-1) \) and a prediction of incoming flow \( b_k(k) \) to \( b_k(k+N) \) for \( i \in B \), which are measurable by sensors or provided by other modules in traffic management systems. An optimization solver solves the problem and obtains an optimal traffic control signal profile. The first step of the optimal profile is implemented. At time \( k+1 \), the sensors and traffic management system will provide fresh data. The solver will repeatedly calculate the optimal traffic light profile based on the latest data in a receding horizon manner, allowing real-time response to changing traffic conditions.

We apply our centralized mixed integer linear programming based scheduling strategy to the network shown in Fig. 2 in order to illustrate how fast the scheduling problem can be solved, compared with the scheduling horizon \( H_p = N \Delta \). The incoming vehicle flow rate \( b_k(k) \) is given and \( L_i/v_{i,\text{max}} = 1 \) is used in the simulation. The maximal volume \( C_j \) for each link \( j \) is chosen as 30. Fast and low speed ratios are chosen as \( p_{ij} = 0.4 \) and \( l_{ij} = 0.2 \), respectively. The sampling interval is 6 seconds. The optimization problem is solved by OPTI [16] based on MATLAB on a PC with an Intel Core(TM) i7-4770 @3.40GHz CPU and RAM 8GB.

We have tested cases of \( n_v = n_h = 2, 5, 7, 10, 20 \) and \( H_p = 6, 12, 18, 30, 48, 60 \) seconds, respectively. Table II summarizes the corresponding computation time in seconds. Each entry with a dash line means that MATLAB fails to run OPTI due to memory insufficiency.

In Table II we can see that OPTI can solve the \( 10 \times 10 \) intersections with 6-second scheduling horizon in just 0.0485 second, which is equivalent to solving a mixed integer linear programming problem with 1530 integer and binary decision variables and 3780 constraints. In a real-time scheduling environment, this means that within each 6-second scheduling horizon, only 0.048/6 = 0.78% of the time is used for computation, which is practically workable. Nevertheless, for intersections with more stages, the number of intersections that can be handled in this centralized framework with a reasonable short computation time will reduce. Table II indicates that the centralized scheduling approach cannot handle a \( 10 \times 10 \) network with a 18-second scheduling horizon for a real-time application, as the computation time (96.9287s) is much bigger than the sampling period (6s), which is also the traffic signal updating period in the receding horizon strategy. The computational challenge is one of the main reasons that prompts us to consider a distributed scheduling strategy, which will be discussed in the next section.

To show the effectiveness of the proposed scheduling strategy, we provide a comparison between the traffic delay time incurred by the proposed strategy and by the fixed time strategy, which reverses the traffic signal after a fixed time interval. Here the sampling period is chosen as 6 seconds and scheduling prediction horizon is 12 seconds. The results presented in Table III show that the proposed scheduling strategy reduces the total delay time significantly.

### III. Distributed Traffic Light Scheduling

A. A distributed scheduling formulation via Lagrangian relaxation

Given a large-scale traffic network, we partition it into regions. Each intersection belongs to only one region, and so does each link, except for a few inside the network which are shared by two regions. To formally describe the concept of regions, we consider the network as a directed graph \( G = (V,E) \), where the vertex set is \( V = J \cup \{\text{ext}\} \) with \( \text{ext} \) denoting the external of the network, and the edge set is...
Let $\lambda$ be the vector consisting of all $\{\lambda_{b,v,v'}/b,v,v' \in B\}$ and define $H(\lambda, R)$ as follows:
\[
\min_{b,v,v'\in B, v,v'\in R} \lambda_{b,v,v'} d(b,v,v') \\
+ \sum_{b,v,v'\in B, v\in R} \lambda_{b,v,v'} s(b,v, v') \\
\text{subject to } \Phi(R)
\]
(16a)

Then the Lagrangian dual problem (16a)-(16b) can be rewritten in the following separable form,
\[
\max_{\lambda \geq 0} \sum_{R \in \mathcal{R}} H(\lambda, R)
\]
(17)

To solve the Lagrangian dual problem (17), we can apply a distributed subgradient method [13], whose computational procedure is described below: It is known that the choice of the step size at each iteration determines the convergence speed. An interesting observation is that the Lagrangian dual problem formulation (17) permits different methods to compute $H(\lambda, R)$, e.g., in our case study we also consider using genetic algorithms to speed up computation when global optimality is not our main goal.

It is well known that an optimal solution to the Lagrangian dual problem need not be a feasible solution to the original optimization problem, let alone its optimality. But luckily in the traffic signal scheduling case, our control variables are traffic (GREEN and RED) lights assignments to intersections, which are always feasible because signal staging constraints are strictly enforced within each individual region. Nevertheless, due to the relaxation of regional boundary constraints specified in (13c), an optimal solution to the Lagrangian dual problem may not satisfy (13c) any more, which means the obtained (predicted) network total delay time by solving the Lagrangian dual problem (17) only serves as a lower bound of the optimal network delay time of the original problem defined in (13a)-(13c).

**B. Experimental results**

We apply the aforementioned distributed scheduling strategy to the traffic network depicted in Fig 2. We consider each region consisting of $5 \times 5$ intersections. Thus, for a network of $10 \times 10$ intersections, we have 4 regions to consider, and for a network of $20 \times 20$ we have 16 regions to consider.
Table IV indicates that we can shorten the scheduling procedure from 96.9287 seconds in the centralized case to 3.63 seconds in the distributed case for a 10 x 10 network with a horizon of 18 seconds, and the computation time for a 20 x 20 network with a horizon of 12 seconds from 36.0907 seconds to 2.3355 seconds - the latter makes real-time scheduling look possible as it is within one sampling period, although it still has room to be improved significantly due to those non-optimized communication processes among regions and the central module, that significantly slow down the total computation time at this moment. For some other cases like \( n_v = n_h = 10 \) with \( H_p = 6(s) \) or \( H_p = 12(s) \), the distributed traffic scheduling strategy obtains slightly worse results than what the centralized strategy can achieve. That is because when the prediction horizon is short and the scale of traffic network is not large, the processing time for centralized model is short. Thus, the distributed traffic scheduling does not possess any advantage as it needs possibly a large amount of communication between the central coordination module and regional schedulers. This suggests that the distributed scheduling strategy may only be useful when the prediction horizon is long or the size of the traffic network is large - in this case the portion of the communication time in the total computation time is negligible.

As discussed before, when applying Algorithm 1 to solve the Lagrangian dual problem, the predicted optimal network delay time may only be a lower bound of the actual optimal network delay time of the original problem. To illustrate this issue, we apply the optimal traffic GREEN light assignments obtained from Algorithm 1 to the traffic network and calculate the actual network delay times. It turns out that such actual delay times are always bigger than the predicted ones, which matches the statements. Table V summarizes the relevant experimental data, where the prediction horizon is 6 seconds to simplify computation.

### IV. Conclusions

In this paper we have proposed novel centralized and distributed scheduling formulations for traffic light assign-ments aiming for minimizing the total network delay. The novel link flow rate functions, which are defined over the upstream volume, the downstream available capacity and the past stage GREEN assignments, allow users to describe nonlinear acceleration (or deceleration) effects on traffic flow rates via mixed logical constraints that can be effectively converted into mixed integer linear constraints. Our simulation results have indicated that the centralized formulation allows efficient real-time scheduling when the total number of intersections and stages, and the prediction horizon are not too big. For a large urban traffic network with more traffic stages, our proposed distributed scheduling strategy based on Lagrangian relaxation performs well. We also show that it is possible to incorporate other computational methods such as genetic algorithms in our Lagrangian relaxation formulation. The current approach assumes that the turning ratios and the network boundary flow rates are known in advance, which in practice are usually stochastic. These issues will be addressed in our future work on real-time system model identification.

### References


