Real-Time Scheduling in Urban Road Traffic Light Control

ZHANG1 Yicheng, SU1 Yang and SU1 Rong

1 School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798; emails: yzhang088@e.ntu.edu.sg, {suyang, rsu}@ntu.edu.sg

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 mixed integer linear programming problem, which can be solved by several existing tools, e.g., CPLEX. Simulation results are provided to demonstrate the effectiveness of the proposed traffic light scheduling approach.

KEYWORD

urban traffic signal scheduling, mixed logical constraints, cell transmission model, mixed integer linear programming

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. In traditional traffic signal control, a signal cycle is one repetition of the basic series of stages at an intersection, where each stage consists of simultaneous traffic light signals allowing a predefined compatible traffic streams to cross the intersection simultaneously. The duration of a cycle is called cycle time. For safety reasons, constant lost (or intergreen) times of a few seconds are necessarily inserted between consecutive stages to avoid interference between antagonistic streams. For each traffic light, the ratio of the green time and the red time within one cycle is called the split, and the delay between the starting times of green periods of two neighboring traffic lights along the same traffic route is called offset.

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 are only applicable to undersaturated traffic conditions. Stage-based strategies under this class, see, e.g., SIGSET and SIGCAP proposed in (Allsop, 1971) and (Allsop, 1976), determine the optimal splits and cycle time so as to minimize the total delay or maximize the intersection capacity. Phase-based strategies, see, e.g., (Improta et al., 1984), determine not only optimal splits and cycle time but also the optimal staging, which may be an important feature for complex intersections. Isolated traffic-responsive strategies, see, e.g., (Miller, 1963)(Vincent et al., 1986), make use of real-time measurements provided by inductive loop detectors that are usually located some 40m upstream of the stop line, to execute some more or less sophisticated vehicle-actuation logic. Fixed time coordinated strategies have been widely used in practice. Typical works in this direction include MAXBAND (Vincent et al., 1986)(Robertson 1969), which considers a two-way arterial with η signals (intersections) and specifies the corresponding offsets so as to maximize the number of vehicles that can travel within a given speed range without stopping at any signal (green wave); TRANSYT (Li et al., 1999)(Hunt et al. 1982), which utilizes platoon dispersion (i.e., a dynamic first-order time-delay system) to model flow progression along a link, and uses heuristic
optimization algorithms such as hill-climb to determine splits, offsets and cycle times that optimize the corresponding performance index such as the total number of vehicle stops. Since fixed time strategies are usually ineffective in coping with highly dynamic traffic situations, coordinated traffic-responsive strategies have gained significant supports. Typical works in this direction include SCOOT (Hunt et al. 1982), which can be considered as the traffic-responsive version of TRANSYT that updates splits, offsets and cycle times based on actual traffic measurements; Model-Based Optimization Methods, see e.g., OPAC (Gartner, 1983), PRODYN (Farges et al., 1983), CRONOS (Boillot et al., 1992), RHODES (Sen et al., 1997), 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, starting from the current time and the currently applied stage; control theoretical approaches, e.g., the usage of the store-and-forward modeling of traffic networks (Gazis et al., 1963)(Gazis, 2002) with a LQR optimal control solution (Gazis, 1964).

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 (Daganzo, 1994). 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 (Bemporad et al., 1999), which is solved by using standard optimization tools such as CPLEX (IBM, 2015). Numerical experiments are presented to illustrate the effectiveness of this scheduling strategy. To improve scalability of this approach for large networks, a distributed strategy with the same modeling formalism is under development, which will be presented in our future reports.

The rest of the paper is organized as follow. A novel formulation of a traffic light scheduling problem is presented in Section II, and a case study of a simplified traffic network is described in Section III. After presenting simulation results in Section IV, conclusions are drawn in Section V.

II. FORMULATION OF A TRAFFIC LIGHT SCHEDULING PROBLEM

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. $\Delta = 5s$. 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 links, and $J$ the set of all intersections. For each intersection $j \in J$, 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 $w_j$, $h_j(w_i) \cap h_j(w_j) = \phi$. 

...
Figure 1 A traffic network

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

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

where $\theta_w(k) = 0$ and $\theta_w(k) = 1$ denote the RED and GREEN traffic lights associated with stage $w$ respectively, and $\mathbb{N}$ 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 $j \in \mathcal{L}$ has the following volume dynamics:

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

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

\[
\begin{align*}
& \left\{ 
\begin{array}{l}
\displaystyle d_j(k) = \sum_{i \in \mathcal{L}(i,j) \in \Omega_j \cap \mathcal{F}_j} f_{ij}(k) \\
\displaystyle s_j(k) = \sum_{i \in \mathcal{L}(i,j) \in \Omega_j \cap \mathcal{F}_j} f_{ji}(k)
\end{array} \right.
\end{align*}
\]

For the example shown in Fig. 1, $d_j(k) = f_{ij}(k)$ and $s_j(k) = f_{jr}(k)$. For each stage $w \in \Omega_J$ and each stream $(i,j) \in h_j(w)$, the exit flow $f_{ij}(k)$ is determined by the current upstream link volume $C_i(k)$, the current remaining downstream link capacity $\hat{C}_j - C_j(k)$, where $\hat{C}_j$ is the maximum volume of link $j$, and the traffic light signals in the past $r + 1$ time intervals $\theta_w(k-r), \ldots, \theta_w(k)$, i.e.,

\[
f_{ij}(k) = g_{ij}(C_i(k), \hat{C}_j - C_j(k), \theta_w(k-r), \ldots, \theta_w(k))
\]

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: $l^0_{ij} \geq \cdots \geq l^r_{ij} > 0$, which denotes speed ranges from high to low. The actual speed category $l_{ij}(k)$ is determined as follows:
where $\lambda_{ij}(k)$ is the turning ratio of vehicles in link $i$ towards link $j$ at $k$, which is assumed to be known in advance. Clearly, 

$$\sum_{j \in L(i,j) \cup \{j\}} \lambda_{ij}(k) = 1$$

means that each vehicle in link $i$ will move into some downstream link $j$. Conditions (5a)-(5b) indicate that the number of vehicles in one time interval, $f_{ij}(k)\Delta$, is the largest integer that is upper bounded by the upstream volume $\lambda_{ij}(k)C_i(k)$ of link $i$ and the downstream remaining capacity $\hat{C}_j - C_j(k)$ weighted by the speed category $l_{ij}(k)$. Condition 5a) can be converted into the following mixed logical constraints:

$$f_{ij}(k)\Delta \leq \lambda_{ij}(k)C_i(k)$$

(6a)

$$f_{ij}(k)\Delta \leq l_{ij}(k)\left(\hat{C}_j - C_j(k)\right)$$

(6b)

$$\left[f_{ij}(k)\Delta + 1 \geq \lambda_{ij}(k)C_i(k) + \epsilon\right] \lor \left[f_{ij}(k)\Delta + 1 \geq l_{ij}(k)\left(\hat{C}_j - C_j(k)\right) + \epsilon\right]$$

(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:

$$\sum_{l \in L} \sum_{k=1}^{N} C_l(k) \left[1 - \frac{v_{\text{avg}}(k)}{v_{\text{max}}(k)}\right] \Delta = \sum_{l \in L} \sum_{k=1}^{N} C_l(k) \left[1 - \frac{S_l(k)}{v_{\text{avg}}(k)}\right] \Delta = \sum_{l \in L} \sum_{k=1}^{N} \left[\frac{C_l(k)}{v_{\text{avg}}(k)} - \frac{L_l}{v_{\text{max}} s_l(k)}\right] \Delta$$

where $v_{\text{avg}}(k)$ is the average speed of link $i$ in interval $k$, which can be approximated by the ratio of the exit flow rate $S_l(k)$ and the average link density $C_l(k)/L_l$, where $L_l$ is the length of the link $i$. Here, we assume that vehicles in link $i$ have a speed identical to the exit speed. In reality, this assumption usually does not hold due to the delay involved in the acceleration process. But for the scheduling purpose, this is a simple yet sufficiently representative performance index, which specifies the minimum delay time that a given traffic light schedule can achieve, when there is no acceleration delay and all vehicles are uniformly distributed in the link. Since
\[
s_i(k) = \sum_{j \in \mathcal{L}(i,j) \in \mathcal{J}} f_{ij}(k)
\]
we have our cost function as
\[
\min \sum_{i \in \mathcal{L}} \sum_{k=1}^{N} \left[ C_i(k) - \frac{L_i}{\psi_i \max} \sum_{j \in \mathcal{J}} f_{ij}(k) \right] \Delta
\]  
(7)

**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 \( M \) be chosen to be sufficient big, e.g., \( M > \max_{i \in \mathcal{L}} \hat{C}_i \). Then (1a) can be converted into
\[
(\forall w \in \Omega_j) \left( \forall (i,j) \in h_j(w) \right) f_{ij}(k) \leq M \theta_w(k)
\]  
(8)

**Proposition 1**: Replacing Condition (1a) with Condition (8) in the urban network traffic signal scheduling formulation leads to the same solution. □

**Proof**: To see this conversion is valid, let \( \theta_w(k) = 0 \), then \( f_{ij}(k) \leq 0 \). But since \( f_{ij}(k) \geq 0 \), we have \( f_{ij}(k) = 0 \). If \( \theta_w(k) = 1 \), it is trivially true that \( f_{ij}(k) \leq M \) because by (5a) we have \( f_{ij}(k) \leq \max_{i \in \mathcal{L}} \hat{C}_i < M \).

Condition (4c) and (4d) can be converted into the following conditions:
\[
\begin{align*}
(\forall q = 0 \leq q \leq r - 1 - (1 - \theta_w(k - q - 1)) + \delta_{ij}^{r-q}(k) \leq 0 \quad & \text{(9a)} \\
(\forall q = 0 \leq q \leq r - 1)(\forall p: 0 \leq p \leq q) - \theta_w(k - p) + \delta_{ij}^{r-q}(k) \leq 0 \quad & \text{(9b)} \\
(\forall q = 0 \leq q \leq r - 1 - (1 - \theta_w(k - q - 1)) + \sum_{p=0}^{q} \theta_w(k - p) - \delta_{ij}^{r-q}(k) \leq q + 1 \quad & \text{(9c)} \\
(\forall p: 0 \leq p \leq r) - \theta_w(k - p) + \delta_{ij}^{0}(k) \leq 0 \quad & \text{(9d)} \\
\sum_{p=0}^{r} \theta_w(k - p) - \delta_{ij}^{0}(k) \leq r \quad & \text{(9e)}
\end{align*}
\]

**Proposition 2**: Replacing Condition (4c) and (4d) with Condition (9a)-(9e) in the urban network traffic signal scheduling formulation leads to the same solution. □

**Proof**: We can verify that Conditions (9a)-(9c) are equivalent to Condition (4c), and Conditions (9d)-(9e) are equivalent to Condition (4d).

Condition (6b) can be rewritten in the following mixed logical constraint:

![Figure 2 A simplified urban traffic network](image-url)
\[ (\forall q = 0 \leq q \leq r) \delta_{ij}^q(k) = 1 \iff f_{ij}(k) \leq l_{ij}^p \left( \hat{C}_j - C_j(k) \right) \] (10)

which can be converted into the following mixed integer linear constraints (Bemporad et al., 1999):

\[
\begin{align*}
(\forall q = 0 \leq q \leq r) f_{ij}(k) \Delta - l_{ij}^p \left( \hat{C}_j - C_j(k) \right) & \leq M_q \left( 1 - \delta_{ij}^q(k) \right) \quad (11a) \\
(\forall q = 0 \leq q \leq r) f_{ij}(k) \Delta - l_{ij}^p \left( \hat{C}_j - C_j(k) \right) & \geq \epsilon + \left( m_q - \epsilon \right) \delta_{ij}^q(k) \quad (11b)
\end{align*}
\]

where \( M_q \) is chosen to be sufficiently big, e.g., \( M_q \geq \max_k \left( f_{ij}(k) - l_{ij}^p \left( \hat{C}_j - C_j(k) \right) \right) \), and \( m_q \) is chosen to be sufficiently small, e.g., \( m_q \leq \min_k \left( f_{ij}(k) - l_{ij}^p \left( \hat{C}_j - C_j(k) \right) \right) \).

Let \([0, r] := \{0, \ldots, r\}\). Condition (6c) is equivalent to the mixed integer constraints below:

\[
\begin{align*}
(\forall q \in [0, r]) m_1(1 - \eta_1(k)) - f_{ij}(k) \Delta - 1 + \lambda_{ij}(k) C_i(k) + \epsilon & \leq 0 \quad (12a) \\
(\forall q \in [0, r]) m_2(1 - \eta_2(k)) - f_{ij}(k) \Delta - 1 + l_{ij}^p \left( \hat{C}_j - C_j(k) \right) + \epsilon & \leq \bar{M}_q(1 - \delta_{ij}^q(k)) \quad (12b) \\
(\forall q \in [0, r]) m_2(1 - \eta_2(k)) - f_{ij}(k) \Delta - 1 + l_{ij}^p \left( \hat{C}_j - C_j(k) \right) & \geq \epsilon + \left( \bar{m}_q - 2\epsilon \right) \delta_{ij}^q(k) \quad (12c) \\
\left[ \eta_1(k) + \eta_2(k) \right] & \geq 1 \quad (12d) \\
\eta_1(k), \eta_2(k) & \in \{0, 1\} \quad (12e)
\end{align*}
\]

where \( m_1 \) is sufficiently small, e.g., \( m_1 \leq \min_k f_{ij}(k) + 1 - \lambda_{ij}(k) C_i(k) - \epsilon \), and \( m_2 \) is sufficiently small, e.g., \( m_2 \leq \min_k l_{ij}^p \left( \hat{C}_j - C_j(k) \right) - \epsilon, \bar{M}_q \) is sufficient big, e.g., \( \bar{M}_q \geq \max_k \left( m_2(1 - \eta_2(k)) - f_{ij}(k) \Delta - 1 + l_{ij}^p \left( \hat{C}_j - C_j(k) \right) \right) + \epsilon \), and \( \bar{m}_q \) is sufficiently small, e.g., \( \bar{m}_q \leq \min_k \left( m_2(1 - \eta_2(k)) - f_{ij}(k) \Delta - 1 + l_{ij}^p \left( \hat{C}_j - C_j(k) \right) \right) + \epsilon \).

**Proposition 3:** Replacing Conditions (6b)-(6c) with Conditions (11a)-(11b), (12a)-(12e) in the urban network traffic signal scheduling formulation leads to the same solution. □

**Proof:** Compare feasible sets of (6b)-(6c), (11a)-(11b), (12a)-(12e), and the proposition holds. ■

We summarize our Urban Traffic Signal Scheduling Problem (UTSSP) as follows:

\[
\min \sum_{i \in E} \sum_{k=1}^N \left[ C_i(k) - \frac{L_i}{v_{\text{max}}} \sum_{j \in \mathcal{J}_i} f_{ij}(k) \right] \Delta
\]

subject to

\[
(\forall j \in \mathcal{J})(\forall w \in \Omega_j) \left( \forall (i, j) \in h_j(w) \right) f_{ij}(k) \leq M \theta_w(k)
\]

\[
(\forall j \in \mathcal{J}) \sum_{w \in h_j} \theta_w(k) = 1
\]

\[
(\forall j \in \mathcal{J})(\forall w \in \Omega_j) \theta_w(k) \in \{0, 1\}
\]

\[
C_j(k + 1) = C_j(k) + \Delta(d_j(k) - s_j(k))
\]

\[
d_j(k) = \sum_{i \in \mathcal{I}_j} f_{ij}(k)
\]

\[
s_j(k) = \sum_{i \in \mathcal{I}_j} f_{ij}(k)
\]

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

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

\[
(\forall q = 0 \leq q \leq r - 1) \left( 1 - \theta_w(k - q - 1) \right) + \delta_{ij}^{r-q}(k) \leq 0
\]

\[
(\forall q = 0 \leq q \leq r - 1)(\forall p: 0 \leq p \leq q) - \theta_w(k - p) + \delta_{ij}^{r-p}(k) \leq 0
\]
\( (\forall q \leq q \leq r - 1)(1 - \theta_w(k - q - 1)) + \sum_{p=0}^{q} \theta_w(k - p) - \delta_{ij}(k) \leq q + 1 \)

\( (\forall p: 0 \leq p \leq r) - \theta_w(k - p) + \delta_{ij}(k) \leq 0 \)

\( \sum_{p=0}^{r} \theta_w(k - p) - \delta_{ij}(k) \leq r \)

\( (\forall q: 0 \leq q \leq r) f_{ij}(k) \Delta \leq \lambda_{ij}(k) C_j(k) \)

\( (\forall q: 0 \leq q \leq r) f_{ij}(k) \Delta - t_{ij}^q (\hat{C}_j - C_j(k)) \leq M_q \left(1 - \delta_{ij}^q(k)\right) \)

\( (\forall q: 0 \leq q \leq r) f_{ij}(k) \Delta - t_{ij}^q (\hat{C}_j - C_j(k)) \geq \epsilon + (m_q - \epsilon) \delta_{ij}^q(k) \)

\( m_1(1 - \eta_2(k)) - f_{ij}(k) \Delta - 1 + \lambda_{ij}(k) C_j(k) + \epsilon \leq 0 \)

\( (\forall q: 0, r) m_2(1 - \eta_2(k)) - f_{ij}(k) \Delta - 1 + t_{ij}^q (\hat{C}_j - C_j(k)) + \epsilon \leq M_q \left(1 - \delta_{ij}^q(k)\right) \)

\( (\forall q: 0, r) m_2(1 - \eta_2(k)) - f_{ij}(k) \Delta - 1 + t_{ij}^q (\hat{C}_j - C_j(k)) \geq \epsilon + (m_q - \epsilon) \delta_{ij}^q(k) \)

\( \eta_1(k) + \eta_2(k) \geq 1 \)

\( \eta_1(k), \eta_2(k) \in [0, 1] \)

### III. A CASE STUDY WITH A RECEIVING 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: \( \text{fast} \) denoted by \( t_{ij}^q \) and \( \text{slow} \) denoted by \( t_{ij}^l \), i.e., \( r = 1 \) in the general problem formulation. 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 dynamic of \( C_i \) for \( i \in B \) obeys the following update:

\[
C_i(k + 1) = C_i(k) + b_i(k) - f_{ij}(k)
\]

where the entrance flow rate \( b_i(k) \) is assumed to be given in advance. The final mixed integer linear programming formulation for the UTSSP in this case study is described as follows:

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

subject to

\( (\forall J \in J) \left( \forall w \in \Omega_j \right) \left( \forall (i,j) \in h_j(w) \right) f_{ij}(k) \leq M \theta_w(k) \)

\( (\forall J \in J) \sum_{w \in \Omega_j} \theta_w(k) = 1 \)

\( (\forall J \in J) \left( \forall w \in \Omega_j \right) \theta_{w}(k) \in \{0, 1\} \)

\( C_j(k + 1) = C_j(k) + \Delta(d_j(k) - s_{-j}(k)) \)

\( d_j(k) = \sum_{i \in I \setminus \{j\} \cup \mathcal{F}_j} f_{ij}(k) \)

\( s_{-j}(k) = \sum_{i \in I \setminus \{j\} \cup \mathcal{F}_j} f_{ji}(k) \)

\( l_{ij}(k) = \delta_{ij}(k) t_{ij}^q + \delta_{ij}(k) t_{ij}^l \)

\( \delta_{ij}(k) + \delta_{ij}(k) - \theta_w(k) = 0 \)

\( -\theta_w(k) + \delta_{ij}(k) \leq 0 \)

\( -\theta_w(k - 1) + \delta_{ij}(k) \leq 0 \)

\( \theta_w(k) + \theta_w(k - 1) - \delta_{ij}(k) \leq 1 \)
\[\delta_{ij}, \delta'_{ij} \in \{0,1\}\]
\[f_{ij}(k)\Delta \leq c_i(k)\]
\[(\forall q = 0 \leq q \leq 1) f_{ij}(k)\Delta - t_{ij}^q \left(\hat{c_j} - c_j(k)\right) \leq M_q \left(1 - \delta_{ij}^q(k)\right)\]
\[(\forall q = 0 \leq q \leq 1) f_{ij}(k)\Delta - t_{ij}^q \left(\hat{c_j} - c_j(k)\right) \geq \epsilon + (m_q - \epsilon)\delta_{ij}^q(k)\]
\[f_{ij}(k)\Delta c_i(k) \in \mathbb{N}\]
\[m_i(1 - \eta_i(k)) - f_{ij}(k)\Delta - 1 + c_i(k) + \epsilon \leq 0\]
\[(\forall q \in [0,1]) m_q(1 - \eta_2(k)) - f_{ij}(k)\Delta - 1 + t_{ij}^q \left(\hat{c_j} - c_j(k)\right) + \epsilon \leq M_q \left(1 - \delta_{ij}^q(k)\right)\]
\[(\forall q \in [0,1]) m_q(1 - \eta_2(k)) - f_{ij}(k)\Delta - 1 + t_{ij}^q \left(\hat{c_j} - c_j(k)\right) \geq \epsilon + (m_q - 2\epsilon)\delta_{ij}^q(k)\]
\[\eta_1(k) + \eta_2(k) \geq 1\]
\[\eta_1(k), \eta_2(k) \in [0,1]\]

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.

### IV. SIMULATION RESULTS

Here we consider a traffic network consisting of different numbers of horizontal and vertical roads. The objective is to illustrate how fast the centralized mixed integer linear programming scheduling problem can be solved, compared with the scheduling horizon \(H_p = N\Delta\). The incoming vehicle flow rate \(b_i(k)\) is given and \(L_i/v_i,max = 1\) is used in the simulation. The maximal volume \(\hat{c}_j\) for each link \(j\) is chosen as 30. Fast and low speed ratios are chosen as \(t_{ij}^q = 0.4\) and \(t_{ij}^q = 0.2\), respectively. The sampling interval is 6 seconds. The optimization problem is solved by CPLEX (IBM, 2015) on a TOSHIBA Portege R700 with an Intel Core(TM) i7 2.67GHz CPU and 4GB (2.68 usable) memory. We have tested cases of \(n_p = n_h = 2, 5, 7, 10\) and \(H_p = 6, 12, 18, 30, 48, 60\) seconds, respectively. Table II summarizes the corresponding computation time in seconds. Each computation time followed by (*) means that the corresponding solution to the optimal scheduling problem is not globally optimal but a sub-optimal one due to our configurations for CPLEX. Each entry with a dash line means that MATLAB fails to run CPLEX due to memory insufficiency.

### Table II Computation time (in seconds) for different cases

<table>
<thead>
<tr>
<th>(n_p = n_h)</th>
<th>(H_p = 6(s))</th>
<th>(H_p = 12(s))</th>
<th>(H_p = 18(s))</th>
<th>(H_p = 30(s))</th>
<th>(H_p = 48(s))</th>
<th>(H_p = 60(s))</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>~0</td>
<td>~0</td>
<td>0.1560</td>
<td>0.2340</td>
<td>0.8890</td>
<td>0.9670</td>
</tr>
<tr>
<td>5</td>
<td>~0</td>
<td>0.1410</td>
<td>0.4060</td>
<td>9.391</td>
<td>78.58</td>
<td>205.8(*)</td>
</tr>
<tr>
<td>7</td>
<td>~0</td>
<td>0.2340</td>
<td>0.9050</td>
<td>76.8460(*)</td>
<td>~</td>
<td>~</td>
</tr>
<tr>
<td>10</td>
<td>0.0160</td>
<td>0.2340</td>
<td>~</td>
<td>~</td>
<td>~</td>
<td>~</td>
</tr>
</tbody>
</table>

In Table II we can see that CPLEX can solve the 10 × 10 intersections with 12-second scheduling horizon in just 0.2340 second, which is equivalent to solving a mixed integer linear
programming problem with 3060 integer and binary decision variables and 7560 constraints. In a real-time scheduling environment, this means that within each 12-second scheduling horizon, only $0.2340/12 = 1.95\%$ 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. Currently, we are exploring distributed scheduling strategies such as parallel computing environments (MathWorks, 2014) and distributed optimization algorithms (Bertsekas et al., 1997) to deal with a large-scale urban traffic network with multiple stages.

To show the effectiveness of the proposed scheduling strategy, we provide a comparison of the traffic delay time between the proposed strategy and the fixed time strategy, which reverses the traffic signal after a fixed time interval. Here the sampling period is chosen as 5 seconds and scheduling prediction horizon is 10 seconds. As presented in Table III, the results show that the proposed scheduling strategy reduces the delay time significantly.

<table>
<thead>
<tr>
<th>$n_c = n_h$</th>
<th>Fixed time (s)</th>
<th>Proposed Strategy (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>38.74</td>
<td>12.80</td>
</tr>
<tr>
<td>5</td>
<td>231.7</td>
<td>45.12</td>
</tr>
<tr>
<td>7</td>
<td>466.1</td>
<td>64.01</td>
</tr>
<tr>
<td>10</td>
<td>955.9</td>
<td>125.1</td>
</tr>
</tbody>
</table>

V. CONCLUSIONS

In this paper we have proposed a novel centralized scheduling formulation for minimizing the total delay time via urban traffic signal control, which is based on a cell transmission model incorporated with novel link flow rate functions over the upstream volume, the downstream available capacity and the past stage GREEN assignments. This modeling feature allows users to describe nonlinear acceleration (or deceleration) effects on traffic flow rates via mixed logical constraints, which can be effectively converted into mixed integer linear constraints. Our simulation results have indicated that this 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 larger urban traffic network with more traffic stages, we have been working on distributed scheduling strategies based on the same network modeling formalism. The current approach assumes that the turning ratios and the network boundary flow rates are known in advance, which in practice are usually stochastic in practise. These issues will be addressed in our future work on real-time system model identification.

REFERENCES


