CONTROL PLAN OPTIMIZATION

Charles Brett
IBM Research - Singapore
9 Changi Business Park Central 1
Singapore, 486048
c.brett@sg.ibm.com

Saif Jabari
New York University - Abu Dhabi
P.O. Box 129188
Abu Dhabi, UAE
sej7@nyu.edu

Sebastien Blandin
IBM Research - Singapore
9 Changi Business Park Central 1
Singapore, 486048
sblandin@sg.ibm.com

Laura Wynter
IBM Research - Singapore
9 Changi Business Park Central 1
Singapore, 486048
lwynter@sg.ibm.com

Word Count: 5704 words + 5 figure(s) + 1 table(s) = 7204 words

Submission Date: October 16, 2015

1 ABSTRACT

We propose an algorithm for real-time optimization of traffic lights in urban road networks. The algorithm is based on maximizing controller flow subject to a macroscopic traffic dynamics model (the cell transmission model). We make simplifications to this formulation that preserve its effectiveness at optimizing traffic lights while decreasing the computational cost by an order of magnitude. The resulting algorithm is based on alternately solving a number of continuous knapsack problems (which are computationally cheap) and simulating traffic dynamics (also cheap). Our algorithm is centralized, allowing coordination between traffic lights, however computationally cheap enough for use in real-time on road networks of moderate size. Numerical results are presented which support this claim and demonstrate the effectiveness of our algorithm at maximizing flow and minimizing delay.

2 INTRODUCTION

Congestion has negative social, economic and environmental consequences, so it is important to investigate if better use can be made of existing road infrastructure through improved traffic control. Many cities now have numerous traffic sensors (loop detectors and cameras) and traffic lights capable of communicating, but we lack algorithms capable of taking full advantage of this in real-time. This paper proposes a computationally efficient algorithm for coordinated optimization of traffic lights.

A building block of such optimization algorithms is a traffic model. The standard macroscopic traffic model is the LWR (Lighthill-Whitman-Richards) model proposed in Lighthill and Whitham (*1*) and Richards (*2*), which is based on a system of hyperbolic conservation laws. Theoretical aspects of this model and junction coupling were developed further in Lebacque (*3*) and Garavello and Piccoli (*4*). The cell transmission model (CTM) proposed in Daganzo, Daganzo (*5, 6*) turns out to be a finite difference discretization of the LWR model using a Godunov scheme (see LeVeque, Lebacque (*7, 8*)) under reasonable assumptions. This model is widely used due to its simplicity, computational efficiency, and ability to replicate traffic dynamics. Szeto et al. (*9*) demonstrates that the CTM can also replicate congested traffic dynamics on arterial networks containing traffic lights.

In order to address the traffic light optimization problem, one might like to use a macroscopic traffic model such as the CTM in a model predictive control (MPC) framework (see García et al. (*10*)). Some literature has explored this possibility (see e.g. Hegyi et al., Hegyi et al., Lin et al. (*11, 12, 13*)), however the CTM requires small time steps for stability (on the order of 1 second). Therefore the resulting nonlinear control problems have large state spaces, even for small time horizons, and generally cannot be solved in real time (see e.g. Gomes and Horowitz, Jacquet et al., Papamichail et al., Lu et al., Lo (*14, 15, 16, 17, 18*)). Recent work such as Tang et al. (*19*) attempts to decompose this nonlinear control problem in order to enable real-time traffic control.

Decentralized control approaches are also an active area of research. For example, back pressure control, which was independently proposed in Wongpiromsarn et al. (*20*) and Varaiya (*21*). This approach is scalable and has the theoretical guarantee of 'maximum network throughput' though it may not coordinate traffic lights optimally or fit with their common cycle based implementation.

The main contribution of this work is a fast algorithm suitable for real-time optimization of traffic lights over a network. It is related to the algorithm in Tang et al. (*19*), but two key modifications allow effective traffic light optimization while decreasing the computational cost by approximately an order of magnitude. As in Tang et al. (*19*) we decomposes the full control problem into two subproblems and alternate between solving each one. The first is the control subproblem of optimizing the traffic lights given fixed traffic states. We propose a new cycle averaging approximation which reduces this problem to a number of continuous knapsack problems, which can be solved computationally cheaply (in comparison to the linear programming control subproblems in Tang et al. (*19*)). Note that the back pressure algorithm also involves solving knapsack problems, but with differently defined weights, so there is a link between this and our algorithm. The second subproblem is simulating traffic states given a fixed control plan. We propose using a simple 'max-flow' junction model, which has an explicit solution. In comparison the more complicated junction model in Tang et al. (*19*) requires solving a linear programming problem, which would dominate the decreased computation time of the control subproblem.

3 NETWORK TRAFFIC FLOW MODELING AND OPTIMIZATION

We begin by introducing a continuous model for traffic flow on a network with traffic lights.

3.1 Link dynamics

First-order continuum traffic flow models such as the LWR model (see Lighthill and Whitham, Richards (*1, 2*)) describe the spatio-temporal evolution of three variables of traffic flow: (i) the traffic density, denoted $\rho(x,t)$, which represents the number of vehicles at time instant $t \in \mathbb{R}_+$ in a "small" section of road centered at $x \in \mathbb{R}$, $(x-\frac{1}{2}\mathrm{d}x, x+\frac{1}{2}\mathrm{d}x)$, divided by the section length $\mathrm{d}x$. (ii) The flow of vehicles, $q(x,t)$, crossing position $x \in \mathbb{R}$ over the short time interval $(t-\frac{1}{2}\mathrm{d}t, t+\frac{1}{2}\mathrm{d}t)$. (iii) The (space) mean speed of vehicles, $v(x,t)$ in $(x-\mathrm{d}x, x+\mathrm{d}x)$ during $(t-\mathrm{d}t, t+\mathrm{d}t)$. By definition, we have that $q=\rho v$ and, consequently, any two of the three macroscopic variables can be used to describe traffic conditions along the road. The conservation of vehicles (or traffic densities) along the road, in the absence of sources and sinks, is given by:

$$\frac{\partial \rho}{\partial t} + \frac{\partial q}{\partial x} = 0. \tag{1}$$

To close the conservation equation, an "equilibrium" flow-density relation is used: $q \equiv Q(\rho)$, which is typically a non-linear concave relation (also known as the "fundamental diagram" of traffic flow). Without loss of generality, we shall assume a unique critical density, denoted by ρ_{crit}, below which traffic conditions are in free-flow and above which traffic conditions are congested. An example of the fundamental diagram is shown in Figure 1. The resulting (first-order) traffic

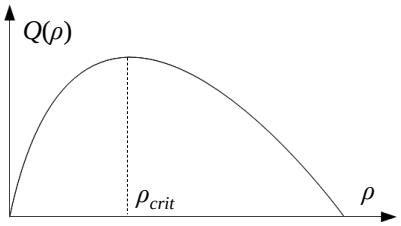


FIGURE 1 : Example fundamental flow-density relation

flow model is a non-linear scalar conservation law:

$$\frac{\partial \rho}{\partial t} + \frac{\partial Q(\rho)}{\partial x} = 0. \tag{2}$$

To solve (2), knowledge of initial traffic conditions is needed; this is given by a prescribed initial traffic density profile: $\rho(\cdot,0)$. The solution of (2) generally develops discontinuities known as shockwaves (see, for example, LeVeque (*7*) for further details). Shockwaves describe the spatio-temporal evolution of traffic jams (and their dissipation dynamics).

3.2 Boundary conditions and node dynamics

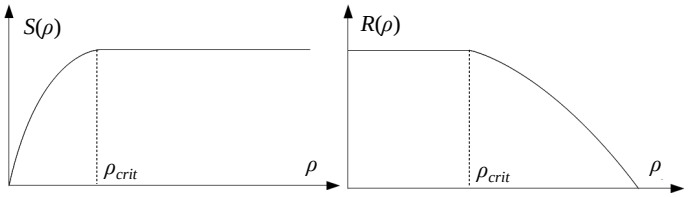
Consider a section of roadway of length L (i.e., $x \in [0,L]$) with fixed traffic sensors at the boundaries providing flow measurements at successive time intervals. The traffic densities, $\rho(0,t+)$ and $\rho(L,t+)$, at the boundaries, $x=0$ and $x=L$, respectively, are given by (see, for example, references Lebacque, Lebacque, Lebacque (*3, 8, 22*)):

$$\rho(0,t+) = \begin{cases} S^{-1}(q(0,t)), & q(0,t) < R\big(\rho(0,t)\big) \\ \rho(0,t), & q(0,t) \geq R\big(\rho(0,t)\big) \end{cases} \tag{3}$$

and

$$\rho(L,t+) = \begin{cases} R^{-1}(q(L,t)), & q(L,t) < S\big(\rho(L,t)\big) \\ \rho(L,t), & q(L,t) \geq S\big(\rho(L,t)\big), \end{cases} \tag{4}$$

where $q(0,t)$ and $q(L,t)$ are prescribed boundary flows and S and R are equilibrium sending and receiving functions (sometimes referred to as equilibrium demand and supply functions, respectively). Examples of S and R are shown in Figure 2.



(a) (b)

FIGURE 2 : Example sending and receiving functions from $Q(\rho)$ (a) sending function, (b) receiving function

Now consider a road network. Denote by $\rho_a(x,t)$ and $q_a(x,t) \equiv Q_a(t,\rho_a)$, respectively, the traffic density and flow at a distance $x \in [0, L_a]$ from the upstream end of road segment $a \in \mathscr{A}$, where Q_a is a segment-specific fundamental relation (allowed also to vary with time) and $\mathscr{A}$ is a set of network road segments. The traffic dynamics are now described by a system of non-linear conservation laws:

$$\frac{\partial \rho_a(x,t)}{\partial t} + \frac{\partial Q_a(t,\rho_a)}{\partial x} = 0, \; a \in \mathscr{A} \tag{5}$$

with prescribed initial conditions $\{\rho_a(\cdot,0)\}_{a\in\mathscr{A}}$. Likewise, the boundary traffic densities must honor the following boundary conditions for each $a \in \mathscr{A}$:

$$\rho_a(0,t+) = \begin{cases} S_a^{-1}\big(t, q_a(0,t)\big), & q_a(0,t) < R_a\big(t,\rho_a(0,t)\big), \\ \rho_a(0,t), & q_a(0,t) \geq R_a\big(t,\rho_a(0,t)\big). \end{cases} \tag{6}$$

and

$$\rho_a(L_a,t+) = \begin{cases} R_a^{-1}\big(t, q_a(L_a,t)\big), & q_a(L_a,t) < S_a\big(t,\rho_a(L_a,t)\big), \\ \rho_a(L_a,t), & q_a(L_a,t) \geq S_a\big(t,\rho_a(L_a,t)\big), \end{cases} \tag{7}$$

where $S_a(t,\rho_a)$ and $R_a(t,\rho_a)$ are local demand and local supply (sending and receiving) functions for road segment a.

The flows $q_a(0,\cdot)$ and $q_a(L_a,\cdot)$ are prescribed for a set of boundary road segments $a \in \mathscr{A}_B \subseteq\in \mathscr{A}$. The boundary flows (both prescribed and otherwise) must honor the following general conditions: (i) traffic is to be conserved at the interfaces between road segments at any instant t, (ii) flows out of the downstream boundary of a road segment are bounded from above by the local demand (i.e., the sending flow) for each time instant t, and (iii) flows into the upstream boundary of a road segment are bounded from above by the local supply (i.e., the receiving flow) for each time instant t. We also impose the condition that flow through each junction is to be maximized to avoid unrealistic flow holding. More stringent conditions have been proposed in Lebacque (*22*) (the invariance principle), Holden and Risebro (*23*) and Jabari (*24*) (non-simultaneity of conflicting flows), but are not considered in this work. Our junction model is reasonable in the situation that lanes do not share movements and lanes are the entire length of a road.

Let $\mathscr{J}$ denote the set of network junctions representing the interfaces between road segments in $\mathscr{A}$. For each $j \in \mathscr{J}$, let $\mathscr{P}(j) \subset \mathscr{A}$ denote the set of road segments immediately preceding (i.e., producing flows into) junction j and let $\mathscr{S}(j) \subset \mathscr{A}$ denote the set of road segments immediately succeeding (i.e., producing flows out of) junction j. We adopt the following formulation for boundary flows ($\{q_i(L_i,t)\}_{i\in\mathscr{P}(j)}$ and $\{q_k(0,t)\}_{k\in\mathscr{S}(j)}$) through a junction, which must be satisfied at each time instant t:

$$\begin{array}{ll} \max\limits_{\{q_i(L_i,t)\}_{i\in\mathscr{P}(j)}} & \sum_{i\in\mathscr{P}(j)} q_i(L_i,t) \\ \text{s.t.} & \sum_{i\in\mathscr{P}(j)} q_i(L_i,t) - \sum_{k\in\mathscr{S}(j)} q_k(0,t) = 0 \\ & 0 \le q_i(L_i,t) \le S_i\big(t,\rho_i(L_i,t-)\big) \qquad i\in\mathscr{P}(j) \\ & 0 \le q_k(0,t) \le R_k\big(t,\rho_k(0,t-)\big) \qquad k\in\mathscr{S}(j). \end{array} \tag{8}$$

In essence, the solution to (8) constitutes a coupling between the system of conservation laws in (5).

3.3 Junction movements

Let $q_{i,k}(t)$ denote the portion of flow departing road segment $i \in \mathscr{P}(j)$ towards road segment $k \in \mathscr{S}(j)$. That is, $q_i(L_i,t) = \sum_{k\in\mathscr{S}(j)} q_{i,k}(t)$ and $q_k(0,t) = \sum_{i\in\mathscr{P}(j)} q_{i,k}(t)$. Consequently, the equality (mass balance) constraint in (8) holds by definition. Using these new variables, (8) can be reformulated as

$$\begin{array}{ll} \max\limits_{\{q_{i,k}(t)\}_{i\in\mathscr{P}(j),k\in\mathscr{S}(j)}} & \sum_{i\in\mathscr{P}(j)}\sum_{k\in\mathscr{S}(j)} q_{i,k}(t) \\ \text{s.t.} & \sum_{k\in\mathscr{S}(j)} q_{i,k}(t) \le S_i\big(t,\rho_i(L_i,t-)\big) \quad i\in\mathscr{P}(j) \\ & \sum_{i\in\mathscr{P}(j)} q_{i,k}(t) \le R_k\big(t,\rho_k(0,t-)\big) \quad k\in\mathscr{S}(j) \\ & q_{i,k}(t) \ge 0 \qquad\qquad\qquad\quad i\in\mathscr{P}(j), k\in\mathscr{S}(j). \end{array} \tag{9}$$

This reformulation expresses the junction flows in terms of individual movements. This will offer the flexibility needed to allow for various signal phasing schemes. Before we present the impact of signal timing on the junction dynamics, we demonstrate that a closed form solution exists for (9). For each $i \in \mathscr{P}(j)$ and each $k \in \mathscr{S}(j)$, the solution is given by

$$q^*_{i,k}(t) \equiv \min\left\{ \frac{R_k(t)}{\sum_{k'\in\mathscr{S}(j)} R_{k'}(t)} S_i(t), \frac{S_i(t)}{\sum_{i'\in\mathscr{P}(j)} S_{i'}(t)} R_k(t) \right\} \tag{10}$$

where the arguments of the local demand and supply functions (as they appear in (8)) are omitted to simplify notation. To demonstrate that $\{q^*_{i,j}(t)\}_{i\in\mathscr{P}(j),k\in\mathscr{S}(j)}$ is an optimal solution to (9), first note that, for each $k \in \mathscr{S}(j)$,

$$\begin{aligned} \sum_{i\in\mathscr{P}(j)} q^*_{i,k}(t) &= \sum_{i\in\mathscr{P}(j)} \min\left\{ \frac{R_k(t)}{\sum_{k'\in\mathscr{S}(j)} R_{k'}(t)} S_i(t), \frac{S_i(t)}{\sum_{i'\in\mathscr{P}(j)} S_{i'}(t)} R_k(t) \right\} \\ &= R_k(t) \min\left\{ \frac{1}{\sum_{k'\in\mathscr{S}(j)} R_{k'}(t)}, \frac{1}{\sum_{i'\in\mathscr{P}(j)} S_{i'}(t)} \right\} \sum_{i\in\mathscr{P}(j)} S_i(t) \\ &\le R_k(t). \end{aligned} \tag{11}$$

Likewise, for each $i \in \mathscr{P}(j)$

$$\begin{aligned}\sum_{k\in\mathscr{S}(j)} q^*_{i,k}(t) &= \sum_{k\in\mathscr{S}(j)} \min\left\{\frac{R_k(t)}{\sum_{k'\in\mathscr{S}(j)} R_{k'}(t)} S_i(t), \frac{S_i(t)}{\sum_{i'\in\mathscr{P}(j)} S_{i'}(t)} R_k(t)\right\} \\ &= S_i(t)\min\left\{\frac{1}{\sum_{k'\in\mathscr{S}(j)} R_{k'}(t)}, \frac{1}{\sum_{i'\in\mathscr{P}(j)} S_{i'}(t)}\right\} \sum_{k\in\mathscr{S}(j)} R_k(t) \\ &\leq S_i(t). \end{aligned} \tag{12}$$

Since, by definition, $S_i(t)$ and $R_k(t)$ are positive (see Figure 2), $\{q^*_{i,j}(t)\}_{i\in\mathscr{P}(j),k\in\mathscr{S}(j)}$ is a feasible solution to (9). Summing the first and second sets of constraints in (9) over $k \in \mathscr{S}(j)$ and $i \in \mathscr{P}(j)$, respectively, we see that the objective function in (9) is bounded from above by

$$\min\left\{\sum_{k\in\mathscr{S}(j)} R_k(t), \sum_{i\in\mathscr{P}(j)} S_i(t)\right\}. \tag{13}$$

To show that this bound is always attained by $\{q^*_{i,j}(t)\}_{i\in\mathscr{P}(j),k\in\mathscr{S}(j)}$, sum the second equality in (11) over $k \in \mathscr{S}(j)$ (alternatively, sum the second equality in (12) over $i \in \mathscr{P}(j)$). We get

$$\sum_{k\in\mathscr{S}(j)}\sum_{i\in\mathscr{P}(j)} q^*_{i,k}(t) = \min\left\{\frac{1}{\sum_{k'\in\mathscr{S}(j)} R_{k'}(t)}, \frac{1}{\sum_{i'\in\mathscr{P}(j)} S_{i'}(t)}\right\} \sum_{i\in\mathscr{P}(j)} S_i(t) \sum_{k\in\mathscr{S}(j)} R_k(t),$$

so that if $\sum_{k'\in\mathscr{S}(j)} R_{k'}(t) \geq \sum_{i'\in\mathscr{P}(j)} S_{i'}(t)$, then

$$\sum_{k\in\mathscr{S}(j)}\sum_{i\in\mathscr{P}(j)} q^*_{i,k}(t) = \sum_{i\in\mathscr{P}(j)} S_i(t).$$

Likewise, if $\sum_{k'\in\mathscr{S}(j)} R_{k'}(t) \leq \sum_{i'\in\mathscr{P}(j)} S_{i'}(t)$, then

$$\sum_{k\in\mathscr{S}(j)}\sum_{i\in\mathscr{P}(j)} q^*_{i,k}(t) = \sum_{k\in\mathscr{S}(j)} R_k(t).$$

Hence, the bound given by (13) is always attained and $\{q^*_{i,j}(t)\}_{i\in\mathscr{P}(j),k\in\mathscr{S}(j)}$ is optimal. It then follows immediately that $q^*_i(L_i,t) \equiv \sum_{k\in\mathscr{S}(j)} q^*_{i,k}(t)$ and $q^*_k(0,t) \equiv \sum_{i\in\mathscr{P}(j)} q^*_{i,k}(t)$ is an optimal solution of (8). Note that this solution is holding free (see Jabari (*24*)) and so appropriate for traffic modeling.

As a special case, consider a junction with one predecessor, $|\mathscr{P}(j)| = 1$, and one successor, $|\mathscr{S}(j)| = 1$. Then, (10) simplifies to $q^*_i(L_i,t) = q^*_k(0,t) = \min\{S_i(t), R_k(t)\}$, which is consistent with numerical schemes used for modeling traffic dynamics within road segments (see Daganzo, Lebacque, Daganzo (*5, 8, 25*)).

3.4 Signal control

The movement-based solution formula (10) can be immediately extended to capture the impact of signal status on boundary flows. The local supply and demand functions, S and R, represent maximal flows under the prevailing traffic conditions (the traffic densities). The formula, (10), can then be interpreted as maximal flows at the junction boundaries. Traffic signal control only restricts

these flows. For junction $j \in \mathscr{J}$, let $g_{i,k}(t) \in [0,1]$ denote such capacity restrictions for the traffic movement between road segments $i \in \mathscr{P}(j)$ and $k \in \mathscr{S}(j)$. The restricted movement flow between $i \in \mathscr{P}(j)$ and $k \in \mathscr{S}(j)$ at time t is given by

$$q_{i,k}(t) = g_{i,k}(t) q^*_{i,k}(t) = g_{i,k}(t) \min \left\{ \frac{R_k(t)}{\sum_{k' \in \mathscr{S}(j)} R_{k'}(t)} S_i(t), \frac{S_i(t)}{\sum_{i' \in \mathscr{P}(j)} S_{i'}(t)} R_k(t) \right\}. \tag{14}$$

The restriction parameter $g_{i,k}(t)$ is allowed to take values in $[0,1]$. This is interpreted as follows: $g_{i,k}(t) = 0$ during a red signal indication, $g_{i,k}(t) = 1$ during a protected green phase, and $g_{i,k}(t) \in (0,1)$ during a permitted (turning) phase. By modifying the restriction variables, g, any signal timing plan can be accommodated in this framework. The restricted boundary flows are given by

$$q_i(L_i,t) = \sum_{k \in \mathscr{S}(j)} q_{i,k}(t) \tag{15}$$

and

$$q_k(0,t) = \sum_{i \in \mathscr{P}(j)} q_{i,k}(t). \tag{16}$$

4 SIGNAL GREEN SPLIT OPTIMIZATION: CONTINUOUS FORMULATION

In this section we formulate the problem of modifying green splits so as to maximize network-wide throughput across network junctions. Note that maximising this does not necessarily imply improvement in other performance measures, such as delay. However it is reasonable to hope for this. In the numerical results section we will see that delay decreased in all of our test cases.

Let $\mathscr{J}_c \subseteq \mathscr{J}$ denote the set of junctions to control. For each $j \in \mathscr{J}_c$, let $\mathscr{M}(j) \equiv \{(i,k) : i \in \mathscr{P}(j), k \in \mathscr{S}(j)\}$ denote the set of movements to be controlled and let $\mathscr{H}_p(j) \subset \mathscr{M}(j)$ denote the set of movements in phase p, which receive a green signal indication simultaneously. Note that $\mathscr{H}_p(j)$ may contain both protected and permitted movements, that is, the movements which belong to a phase need not be non-conflicting. The set of all phases shall be referred to here as a signal phasing plan. The phasing plan is denoted by $\mathscr{H}(j) = \{\mathscr{H}_1(j), \cdots, \mathscr{H}_{|\mathscr{H}(j)|}(j)\}$. Let $g_p(t) \in \{0,1\}$ denote whether phase p receives a green signal indication at time t. Then, for each $(i,k) \in \mathscr{M}(j)$

$$g_{i,k}(t) = \sum_{p=1}^{|\mathscr{H}(j)|} \alpha^p_{i,k} g_p(t), \tag{17}$$

where $\alpha^p_{i,k} = 1$ if movement (i,k) is protected in phase p, $\alpha^p_{i,k} \in (0,1)$ if (i,k) is permitted in phase p, and $\alpha^p_{i,k} = 0$ if (i,k) is prohibited in phase p.

Our optimization will not alter the phasing plans, only the green durations allocated to the different phases and, possibly, the order of the phases. Consequently, the variables $\{\alpha^p_{i,k}\}$ are constant and fixed (they could be calibrated for each junction based on data or experience). Then, from (17), the movement variables $g_{i,k}$ are uniquely determined by the phase status variables g_p. The latter shall serve as our decision variables. Throughout, we will assume a finite control/computational horizon, that is, $t \in [0,T]$, where $T < \infty$. We are now ready to state the control constraints.

Bounds on phase lengths. For $j \in \mathscr{J}_c$, let Ω_j denote a fixed cycle length. To prohibit aggressive changes in signal timing (compared to a baseline timing plan), we place upper and lower bound

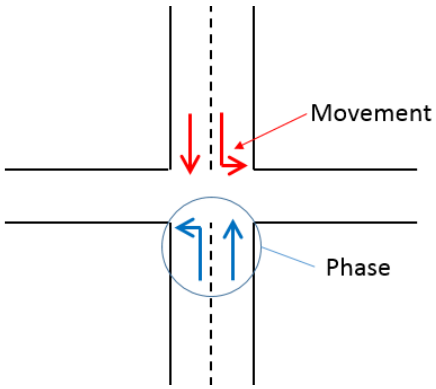


FIGURE 3 : A movement allows traffic to move from one incoming link to one outgoing link of a junction. A phase is a set of movements that are active simultaneously.

constraints, denoted l_p^j and u_p^j, on the length of each phase: for each $t \in [0,T]$, $p \in \{1,\cdots,|\mathscr{H}(j)|\}$, and $j \in \mathscr{J}_c$

$$l_p^j \leq \int_t^{t+\Omega_j} g_p(u)\mathrm{d}u \leq u_p^j. \tag{18}$$

One phase at a time. This is ensured via the following two sets of constraints: for each $t \in [0,T]$ and each $j \in \mathscr{J}_c$

$$\sum_{p=1}^{|\mathscr{H}(j)|} g_p(t) = 1, \tag{19}$$

and for each $p \in \{1,\cdots,|\mathscr{H}(j)|\}$

$$g_p(t) \in \{0,1\}. \tag{20}$$

Each phase is active at most once during a signal cycle. The following states that a switch in phase status from inactive to active occurs no more than once over the course of a cycle: for each $t \in (0,T]$,$p \in \{1,\cdots,|\mathscr{H}(j)|\}$, and $j \in \mathscr{J}_c$

$$\int_t^{t+\Omega_j} \delta\big(\Delta g_p(u) - 1\big)\mathrm{d}u \leq 1, \tag{21}$$

where δ is the Dirac delta function and $\Delta g_p(t) = g_p(t) - g_p(t-)$. $\Delta g_p(t) = 1$ if at time instant t phase p goes from inactive to active, $\Delta g_p(t) = 0$ if there is no change in the status of phase p at time instant t, and $\Delta g_p(t) = -1$ if the status of phase p changes from active to inactive.

When $l_p^j > 0$ for all p, (18) - (21) also ensure a constant cycle length, Ω_j, with each phase being active only once during the phase. Since $l_p^j > 0$ for all p, each phase must become active during any interval of length Ω_j, in accordance with (18). In this case, the constraints in (21) are binding. Hence, each phase occurs exactly once during any signal cycle. That the resulting cycle length is Ω_j follows immediately from (19) by integrating over intervals of length Ω_j. For each $t \in [0,T]$ and each $j \in \mathscr{J}_c$, we get

$$\int_t^{t+\Omega_j} \sum_{p=1}^{|\mathscr{H}(j)|} g_p(u)\mathrm{d}u = \Omega_j. \tag{22}$$

Hence, cycle lengths remain constant provided $l_p^j > 0$ for all p. In addition, the following condition is necessary for feasibility: for each $j \in \mathscr{J}_c$

$$\sum_{p=1}^{|\mathscr{H}(j)|} l_p^j \leq \Omega_j \leq \sum_{p=1}^{|\mathscr{H}(j)|} u_p^j. \tag{23}$$

Objective function. We wish to optimize throughput (i.e., maximize flow) through network junctions by modifying signal splits:

$$\max_{\{g_p\}} \sum_{j \in \mathscr{J}_c} \sum_{i \in \mathscr{P}(j)} \int_0^T q_i(L_i,t)\mathrm{d}t, \tag{24}$$

where, from (14), (15), and (17), we have that

$$q_i(L_i,t) = \sum_{k \in \mathscr{S}(j)} \sum_{p=1}^{|\mathscr{H}(j)|} \alpha_{i,k}^p q_{i,k}^*(t) g_p(t), \;\; t \in [0,T], j \in \mathscr{J}_c, i \in \mathscr{P}(j). \tag{25}$$

This captures the dependence of junction inflows on signal status.

5 COMPUTATIONAL SCHEME

We now state discretized versions of the traffic dynamics model and control problem which are suitable for solving computationally.

5.1 Network traffic state dynamics

We employ a discrete time and space (finite volume) computational technique similar to those described in Daganzo, Daganzo, Lebacque, Daganzo (*5, 6, 8, 25*), but generalized to include general (many-to-many) junctions. Each road segment is divided into small cells. Let $\mathscr{C}$ denote the set of network cells and let l_c denote the length of cell c; without loss of generality, l_c is constant across cells pertaining to the same road segment. We compute the traffic variables in discrete time steps of length Δt, which divides the time horizon into N steps (i.e., $T = N\Delta t$). Denote by ρ_c^n the (average) traffic density in cell c over the time interval $[(n-1)\Delta t, n\Delta t)$, where $n \in \{0,1,\cdots,N\}$. That is

$$\rho_c^n \equiv \frac{1}{l_c} \int_{\{x \in X_c\}} \rho_{a(c)}(x, n\Delta t)\mathrm{d}x, \tag{26}$$

where X_c defines the spatial extent of cell c and $a(c)$ is the index of the road segment to which cell c belongs. Likewise, q_{c-}^n and q_{c+}^n denote the average flow entering and departing cell c, respectively, over time interval n:

$$q_{c-}^n \equiv \frac{1}{\Delta t} \int_{[n\Delta t,(n+1)\Delta t)} q_{a(c)}\big(\inf X_c, t\big)\mathrm{d}t \tag{27}$$

and

$$q_{c+}^n \equiv \frac{1}{\Delta t} \int_{[n\Delta t,(n+1)\Delta t)} q_{a(c)}\big(\sup X_c, t\big)\mathrm{d}t. \tag{28}$$

The discrete time interval length, Δt, is global; that is, the traffic state throughout the entire network is calculated simultaneously in each time step. The choice of Δt is not arbitrary; it must honor the

Courant, Friedrichs, and Lewy (CFL) condition for all cells in the network. That is,

$$\Delta t \leq \min_{c\in\mathscr{C},t\in[0,T]} \left(\frac{l_c}{\sup\limits_{0\leq t\leq T} \sup\limits_{0\leq\kappa\leq\rho^t_{jam,c}} \left|\partial_\rho Q_c(t,\kappa)\right|} \right), \tag{29}$$

where $\partial_\rho Q_c(t,\rho) = \frac{\partial Q_c(t,\rho)}{\partial\rho}$ is the characteristic slope at density ρ for cell c and $\rho^t_{jam,c}$ is the jammed traffic density in cell c at time instant t. The traffic densities in the network are then computed recursively using

$$\rho^{n+1}_c - \rho^n_c = \frac{\Delta t}{l_c}\left(q^n_{c-} - q^n_{c+}\right),\ c\in\mathscr{C}. \tag{30}$$

The cell boundary flows q^n_{c-} and q^n_{c+} are calculated in accordance with the boundary conditions discussed in Subsection 3.3 and Subsection 3.4. That is, cell boundaries are treated as network junctions with predecessors and successors. With slight abuse of notation, let $\mathscr{P}(c-)$ ($\mathscr{S}(c-)$) denote the set of cells immediately preceding (succeeding) the upstream boundary of cell c and let $\mathscr{P}(c+)$ ($\mathscr{S}(c+)$) denote the same pertaining to the downstream boundary of cell c. Note that these boundaries may also be network junctions, i.e., they may also belong to $\mathscr{J}$. Also let S^n_c and R^n_c denote the sending and receiving (local demand and supply) functions for cell c. We shall adopt the following definitions: $S^n_c(\rho)\equiv S_{a(c)}(n\Delta t,\rho)$ and $R^n_c(\rho)\equiv R_{a(c)}(n\Delta t,\rho)$. The discrete movement flows (analogous to (14)) are given by

$$q^n_{c,d} = g^n_{c,d} q^{n,*}_{c,d} = g^n_{c,d}\min\left\{ \frac{R^n_d(\rho^n_d)}{\sum_{d'\in\mathscr{S}(c+)} R^n_{d'}(\rho^n_{d'})} S^n_c(\rho^n_c), \frac{S^n_c(\rho^n_c)}{\sum_{c'\in\mathscr{P}(d-)} S^n_{c'}(\rho^n_{c-})} R^n_d(\rho^n_d) \right\}, \tag{31}$$

where $g^n_{c,d}\equiv 1$ when flows between cell c and d do not cross a signalized intersection. When the boundary $c+\ (=d-)$ is a signalized junction $g^n_{c,d}\equiv g_{a(c),a(d)}(n\Delta t)$. (We are assuming, without loss of generality, that signal status changes coincide with the ticks of our discrete time interval clock.)

Finally, analogous to (16) and (15), respectively, we have that

$$q^n_{c-} = \sum_{b\in\mathscr{P}(c-)} q^n_{b,c} \tag{32}$$

and

$$q^n_{c+} = \sum_{d\in\mathscr{S}(c+)} q^n_{c,d}. \tag{33}$$

This, combined with equation (30) provides a recipe for iteratively calculating cell traffic densities over the course of the computational time horizon, when the signal timing plans are known. The steps are summarized in Algorithm 1.

5.2 Signal green splits: Discrete formulation

In this section, we formulate a discrete analogue of the continuous formulation presented in Section 4. For junction $j\in\mathscr{J}_c$, let g^n_p denote the status of phase $p\in\{1,\cdots,|\mathscr{H}(j)|\}$ during time step

Algorithm 1: Network traffic state computation scheme

Initialize:
initial conditions: set ρ_c^0 for each $c \in \mathscr{C}$
boundary conditions: set $q_{c-}^n = q_{a(c)}(0, n\Delta t)$ and $q_{c+}^n = q_{a(c)}(L_{a(c)}, n\Delta t)$ for $n = 1, \cdots, N$, $a \in \mathscr{A}_B$
Iterate:
for $n = 1 \cdots N$ **do**
  for $c \in \mathscr{C}$ **do**
    $S_c^{n-1} \leftarrow S_{a(c)}((n-1)\Delta t, \rho_c^{n-1})$
    $R_c^{n-1} \leftarrow R_{a(c)}((n-1)\Delta t, \rho_c^{n-1})$
  end for
  for $c \in \mathscr{C}$ **do**
    $q_{c,d}^{n-1} \leftarrow g_{c,d}^{n-1} \min\left\{ \frac{R_d^{n-1}}{\sum_{d' \in \mathscr{S}(c+)} R_{d'}^{n-1}} S_c^{n-1}, \frac{S_c^{n-1}}{\sum_{c' \in \mathscr{P}(c+)} S_{c'}^{n-1}} R_d^{n-1} \right\}$
    $q_{c-}^{n-1} = \sum_{b \in \mathscr{P}(c-)} q_{b,c}^{n-1}$
    $q_{c+}^{n-1} = \sum_{d \in \mathscr{S}(c+)} q_{c,d}^{n-1}$
    `/* if c is the upstream-most (boundary) cell of a(c) */`
    if $\inf X_c == 0$ **then**
      $\rho_c^n \leftarrow \begin{cases} S_{a(c)}^{-1}\left((n-1)\Delta t, q_{c-}^{n-1}\right), & q_{c-}^{n-1} < R_c^{n-1}, \\ \rho_c^{n-1}, & q_{c-}^{n-1} \geq R_c^{n-1} \end{cases}$
    else if $\sup X_c == L_a$ **then**
      `/* if c is the downstream-most (boundary) cell of a(c) */`
      $\rho_c^n \leftarrow \begin{cases} R_{a(c)}^{-1}\left((n-1)\Delta t, q_{c+}^{n-1}\right), & q_{c+}^{n-1} < S_c^{n-1}, \\ \rho_c^{n-1}, & q_{c+}^{n-1} \geq S_c^{n-1} \end{cases}$
    else
      `/* if c is an intermediate cell of a(c) */`
      $\rho_c^n \leftarrow \rho_c^{n-1} + \frac{\Delta t}{l_c}\left(q_{c-}^{n-1} - q_{c+}^{n-1}\right)$
    end if
  end for
end for

n and let ω_j denote the number of time steps in cycle j, that is, $\omega_j \Delta t = \Omega_j$. The discrete analogues of (18) - (20) are immediately given by:

$$l_p^j \leq \sum_{m=n}^{n+\omega_j} g_p^m \leq u_p^j \quad 0 \leq n \leq N - \omega_j, 1 \leq p \leq |\mathscr{H}(j)|, j \in \mathscr{J}_c \tag{34}$$

$$\sum_{p=1}^{|\mathscr{H}(j)|} g_p^n = 1 \quad 0 \leq n \leq N, j \in \mathscr{J}_c \tag{35}$$

$$g_p^n \in \{0,1\} \quad 0 \leq n \leq N, 1 \leq p \leq |\mathscr{H}(j)|, j \in \mathscr{J}_c. \tag{36}$$

Now, for any p and n, the discrete analogue of (21) is given by

$$\sum_{m=n}^{n+\omega_j} \mathbf{1}_{\{g_p^m - g_p^{m-1} = 1\}} \leq 1. \tag{37}$$

Since the decision variables g_p^n are binary, this is equivalent to

$$\sum_{m=n}^{n+\omega_j} \left(g_p^m - g_p^{m-1}\right)^+ \leq 1, \tag{38}$$

where $(\cdot)^+ = \max\{\cdot, 0\}$. This can be written as[1]

$$\sum_{m=n}^{n+\omega_j} \left(g_p^m - g_p^{m-1} + \left| g_p^m - g_p^{m-1} \right| \right) \leq 2. \tag{39}$$

For $n \in \{1, \cdots, N-\omega_j\}$, $p \in \{1, \cdots, |\mathscr{H}(j)|\}$, and $j \in \mathscr{J}_c$, introduce new decision variables $\{z_p^n\}$ and new constraints

$$\left| g_p^n - g_p^{n-1} \right| \leq z_p^n.$$

Then, (39) can be replaced with

$$\sum_{m=n}^{n+\omega_j} \left(g_p^m - g_p^{m-1} + z_p^m \right) \leq 2.$$

Thus, the following (linear!) constraints will serve as a discrete analogue of (21)

$$\begin{aligned}
\textstyle\sum_{m=n}^{n+\omega_j} \left(g_p^m - g_p^{m-1} + z_p^m \right) \leq 2 \quad & 1 \leq n \leq N-\omega_j, 1 \leq p \leq |\mathscr{H}(j)|, j \in \mathscr{J}_c && (40)\\
g_p^n - g_p^{n-1} - z_p^n \leq 0 \quad & 1 \leq n \leq N, 1 \leq p \leq |\mathscr{H}(j)|, j \in \mathscr{J}_c && (41)\\
g_p^n - g_p^{n-1} + z_p^n \geq 0 \quad & 1 \leq n \leq N, 1 \leq p \leq |\mathscr{H}(j)|, j \in \mathscr{J}_c. && (42)
\end{aligned}$$

5.3 Decomposition approach

Due to the size and complexity of the overall problem, solution algorithms typically used to solve non-linear programming problems are ineffective in our case. For this reason, a decomposition approach which utilizes the structure of the subproblems is used. The approach iterates between solving the network traffic dynamics problem and the control problem.

We start with a baseline timing plan $\{g_p^n\}$ and calculate the constants $\{g_{c,d}^n\}$ using (17) and $g_{c,d}^n = g_{a(c),a(d)}(n\Delta t)$. Next, we use these constants to compute the network traffic states using Algorithm 1. We then use the output to solve the following binary integer programming subproblem:

$$\begin{aligned}
\max_{\{g_p^n, z_p^n\}} \quad & \textstyle\sum_{j\in\mathscr{J}_c} \sum_{(i,k)\in\mathscr{M}(j)} \sum_{p=1}^{|\mathscr{H}(j)|} \sum_{n=0}^{N} \alpha_{i,k}^p q_{i,k}^{n,*} g_p^n && \\
\text{s.t.} \quad & l_p^j \leq \textstyle\sum_{m=n}^{n+\omega_j} g_p^m \leq u_p^j && 0 \leq n \leq N-\omega_j, 1 \leq p \leq |\mathscr{H}(j)|, j \in \mathscr{J}_c \\
& \textstyle\sum_{p=1}^{|\mathscr{H}(j)|} g_p^n = 1 && 0 \leq n \leq N, j \in \mathscr{J}_c \\
& \textstyle\sum_{m=n}^{n+\omega_j} \left(g_p^m - g_p^{m-1} + z_p^m \right) \leq 2 && 1 \leq n \leq N-\omega_j, 1 \leq p \leq |\mathscr{H}(j)|, j \in \mathscr{J}_c \\
& g_p^n - g_p^{n-1} - z_p^n \leq 0 && 1 \leq n \leq N, 1 \leq p \leq |\mathscr{H}(j)|, j \in \mathscr{J}_c \\
& g_p^n - g_p^{n-1} + z_p^n \geq 0 && 1 \leq n \leq N, 1 \leq p \leq |\mathscr{H}(j)|, j \in \mathscr{J}_c \\
& g_p^n \in \{0,1\} && 0 \leq n \leq N, 1 \leq p \leq |\mathscr{H}(j)|, j \in \mathscr{J}_c.
\end{aligned} \tag{43}$$

The only change to (43) from one iteration to the next is the $\{q_{i,k}^{n,*}\}$ variables obtained after calculating the network traffic states using Algorithm 1. In other words, we only change the slope

[1] Since $(z)^+ = \frac{1}{2}(z + |z|)$ for any $z \in \mathbb{R}$.

of the objective function from one iteration to the next. Since decision variables pertaining to different junctions do not interact (in the constraints), (43) can be decomposed into $|\mathscr{J}_c|$ subproblems that can be solved in parallel. Thus, for each $j \in \mathscr{J}_c$, we have a control subproblem given by

$$\begin{array}{lll} \max\limits_{\{g_p^n, z_p^n\}} & \sum_{(i,k)\in\mathscr{M}(j)} \sum_{p=1}^{|\mathscr{H}(j)|} \sum_{n=0}^{N} \alpha_{i,k}^p q_{i,k}^{n,*} g_p^n & \\ \text{s.t.} & l_p^j \leq \sum_{m=n}^{n+\omega_j} g_p^m \leq u_p^j & 0 \leq n \leq N-\omega_j, 1 \leq p \leq |\mathscr{H}(j)| \\ & \sum_{p=1}^{|\mathscr{H}(j)|} g_p^n = 1 & 0 \leq n \leq N \\ & \sum_{m=n}^{n+\omega_j} \left(g_p^m - g_p^{m-1} + z_p^m\right) \leq 2 & 1 \leq n \leq N-\omega_j, 1 \leq p \leq |\mathscr{H}(j)| \\ & g_p^n - g_p^{n-1} - z_p^n \leq 0 & 1 \leq n \leq N, 1 \leq p \leq |\mathscr{H}(j)| \\ & g_p^n - g_p^{n-1} + z_p^n \geq 0 & 1 \leq n \leq N, 1 \leq p \leq |\mathscr{H}(j)| \\ & g_p^n \in \{0,1\} & 0 \leq n \leq N, 1 \leq p \leq |\mathscr{H}(j)|. \end{array} \tag{44}$$

5.4 Simplification of binary programming problem

We now consider another simplification to the control subproblems (44) that results from (i) changing the objective function to one that maximizes the averaged throughputs over signal cycles and (ii) assuming that the order of the phases is not to be altered (only their duration).

Without loss of generality, we will assume that T can be divided evenly into Υ_j intervals of length Ω_j (ω_j in units of discrete time steps) for each junction j (typically, $\Upsilon_j \gg \Delta t$). Let w_p^τ denote the length of phase p in cycle τ. That is, for $\tau \geq 1$,

$$w_p^\tau \equiv \int_{(\tau-1)\Omega_j}^{\tau\Omega_j} g_p(u)\mathrm{d}u. \tag{45}$$

In terms of the discrete variables, we have that

$$w_p^\tau = \sum_{n=(\tau-1)\omega_j}^{\tau\omega_j} g_p^n. \tag{46}$$

Hereafter, we adopt the latter, in which case, phase lengths are in (integer) units of number of discrete time steps. The signal timing constraints are immediately given by

$$l_p^j \leq w_p^\tau \leq u_p^j \quad 1 \leq \tau \leq \Upsilon_j, 1 \leq p \leq |\mathscr{H}(j)| \tag{47}$$

$$\sum_{p=1}^{|\mathscr{H}(j)|} w_p^\tau = \omega_j \quad 1 \leq \tau \leq \Upsilon_j. \tag{48}$$

Binary constraints are not needed in this setting. Furthermore, that each phase is active once during a cycle is implicit in this new formulation. These consequences follow directly from the two simplifications considered above. For cycle τ and phase p, the average throughput for movement (i,k) is given by

$$\frac{w_p^\tau}{\omega_j} \sum_{n=(\tau-1)\omega_j}^{\tau\omega_j} \alpha_{i,k}^p q_{i,k}^{n,*},$$

which is the total throughput over the cycle scaled by the fraction of time allocated to phase p in cycle τ. For junction $j \in \mathscr{J}_c$, maximizing the total, cycle-averaged, throughput over all cycles, all

movements, and all phases is written as follows:

$$\max_{\{w_p^\tau\}} \sum_{p=1}^{|\mathscr{H}(j)|} \sum_{\tau=1}^{\Upsilon_j} \left(\sum_{(i,k)\in\mathscr{M}(j)} \sum_{n=(\tau-1)\omega_j}^{\tau\omega_j} \frac{\alpha_{i,k}^p q_{i,k}^{n,*}}{\omega_j} \right) w_p^\tau. \tag{49}$$

To simplify notation, define the decomposition constants

$$\beta_p^\tau \equiv \sum_{(i,k)\in\mathscr{M}(j)} \sum_{n=(\tau-1)\omega_j}^{\tau\omega_j} \frac{\alpha_{i,k}^p q_{i,k}^{n,*}}{\omega_j}.$$

which are obtained directly from the network traffic states calculated using Algorithm 1. The simplified junction control subproblems are given by

$$\begin{aligned} \max_{\{w_p^\tau\}} \quad & \textstyle\sum_{\tau=1}^{\Upsilon_j} \sum_{p=1}^{|\mathscr{H}(j)|} \beta_p^\tau w_p^\tau \\ \text{s.t.} \quad & \textstyle\sum_{p=1}^{|\mathscr{H}(j)|} w_p^\tau = \omega_j \qquad 1 \le \tau \le \Upsilon_j \\ & l_p^j \le w_p^\tau \le u_p^j \qquad 1 \le \tau \le \Upsilon_j, 1 \le p \le |\mathscr{H}(j)|. \end{aligned} \tag{50}$$

Notice that decision variables in (50) pertaining to different cycles do not appear in the same constraints. Hence, these subproblems can be further decomposed into one problem per signal cycle. The resulting problems are easy to solve using a greedy algorithm. To see this, let $\overline{w}_p^\tau \equiv w_p^\tau - l_p^j$, $\overline{u}_p^j \equiv u_p^j - l_p^j$, and $\overline{\omega}_j \equiv \omega_j - \sum_p^{|\mathscr{H}(j)|} l_p^j$. Then, the decomposed problems are equivalent to

$$\begin{aligned} \max_{\{\overline{w}_p^\tau\}} \quad & \textstyle\sum_{p=1}^{|\mathscr{H}(j)|} \beta_p^\tau \overline{w}_p^\tau \\ \text{s.t.} \quad & \textstyle\sum_{p=1}^{|\mathscr{H}(j)|} \overline{w}_p^\tau = \overline{\omega}_j \\ & 0 \le \overline{w}_p^\tau \le \overline{u}_p^j \qquad 1 \le p \le |\mathscr{H}(j)|. \end{aligned} \tag{51}$$

Solving (51) is equivalent to solving a continuous knapsack problem[2], provided the discrete analogue of the feasibility condition (23) holds for each j:

$$\sum_{p=1}^{|\mathscr{H}(j)|} l_p^j \le \omega_j \le \sum_{p=1}^{|\mathscr{H}(j)|} u_p^j. \tag{52}$$

When (52) holds, the greedy algorithm used to solve the knapsack problem is guaranteed to produce an optimal solution to (51), that is, one that also honors the equality constraint in (51). To summarize, the control subproblem can be solved using Algorithm 2.

Since the lower and upper bounds on the phase lengths, l_p^j and u_p^j, are in (integer) units of discrete time steps, Algorithm 2 produces integer solutions for w_p^τ. Assume the orders of the phases is the same as their indices, i.e., phase p is the pth phase of the cycle. Then, the output of the control subproblem is translated into input to the network traffic state computation scheme by setting $g_p^n = 1$ for each $n \in \mathscr{Q}_p^n = \{(\tau-1)\omega_j + \sum_{q=1}^{p-1} w_q^\tau, \cdots, (\tau-1)\omega_j + \sum_{q=1}^{p} w_q^\tau - 1\}$ and $g_p^n = 0$ for $n \in \{(\tau-1)\omega_j, \cdots, \tau\omega_j\}/\mathscr{Q}_p^n$, where $\sum_{q=1}^{p-1} w_q^\tau = 0$ for $p = 1$.

[2]The only difference between (51) and a continuous knapsack problem is the equality constraint in (51), which is an inequality ($\le$) in the knapsack problem formulation.

Algorithm 2: Control subproblem solution algorithm

Initialize:
For each junction j, each phase p, and each cycle τ, $\beta_p^\tau \leftarrow \sum_{(i,k)\in\mathscr{M}(j)} \sum_{n=(\tau-1)\omega_j}^{\tau\omega_j} \frac{\alpha_{i,k}^p q_{i,k}^{n,*}}{\omega_j}$
Iterate:
for $j \in \mathscr{J}_c$ **do**
  for $\tau = 1 \cdots \Upsilon_j$ **do**
    Sort the phases $p \in \mathscr{H}(j)$ in descending order of β_j^τ:
    $\{(1), \cdots, (|\mathscr{H}(j)|) : \beta_{(1)}^\tau \geq \beta_{(2)}^\tau \geq \cdots \geq \beta_{(|\mathscr{H}(j)|)}^\tau\}$
    $\overline{w}_{(1)}^\tau \leftarrow \min\{\overline{\omega}_j, \overline{u}_{(1)}\}$
    for $(p) = 2 \cdots (|\mathscr{H}(j)|)$ **do**
      $\overline{w}_{(p)}^\tau \leftarrow \min\{\overline{\omega}_j - \sum_{(q)=1}^{(p)-1} \overline{w}_{(q)}^\tau, \overline{u}_{(p)}^j\}$
    end for
    for $p \in \mathscr{H}(j)$ **do**
      $w_p^\tau \leftarrow \overline{w}_p^\tau + l_p^j$
    end for
  end for
end for

5.5 Full algorithm

Our full algorithm is formally summarized in Algorithm 3. We will discuss the stopping criterion in the numerical results section. Note that a related computationally cheaper algorithm (that may not optimize the control plan as effectively) is to optimize *all* junctions (rather than just one) at each cycle before simulating again.

Algorithm 3: Full algorithm

Set initial control plan
repeat
  for cycles c **do**
    for controllable junctions j **do**
      Simulate traffic dynamics using Algorithm 1
      Optimize phase splits for a junction j at cycle c using Algorithm 2
    end for
  end for
until stopping criterion achieved

We envisage using Algorithm 3 in a model predictive control type way: At each cycle we run the algorithm to generate an optimized control plan over a time horizon of multiple cycles, but only apply the first cycle of controls. This is desirable in a real-time system as it allows us to make use of incoming data.

The dependency between junctions (e.g. the phase split of an upstream junction influencing the optimal phase split of a downstream junction) is taken care of because we optimize over all junctions and cycles multiple times as a result of the 'repeat' part of the algorithm. The algorithm is not guaranteed to find a minimizer of the optimization problem, but we will see in the next section that in our test cases it finds a solution that improves the objective value over the initial

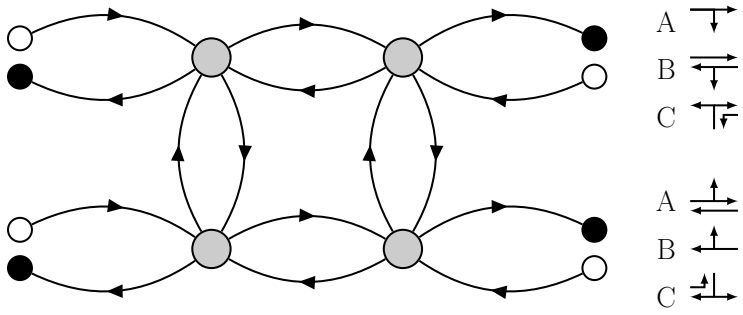


FIGURE 4 : The directed graph on the left illustrates the topology of our synthetic road network. White circles are source nodes, black circles are sink nodes, and gray circles are controllable junctions. In the top right we illustrate the phases for the top two junctions. In the bottom right we illustrate the phases for the bottom two junctions.

condition.

6 NUMERICAL RESULTS

In this section we will discuss the performance of Algorithm 3 in three key areas: Algorithmic performance (convergence properties of the iterative method), runtime performance (computational cost of the algorithm in comparison to the related algorithm from Tang et al. (*19*)), and control performance (the improvement in traffic flow compared to a base plan).

We will mainly demonstrate performance on a synthetic network containing 4 controllable junctions, 16 links, and 4 source nodes. Its network and phase topology are illustrated in Figure 4. Note that such junctions and phases can be observed in real road networks. The optimized control plan on a network of this size is easy to interpret and the low computational cost enables us to solve many different instances of the problem with a variety of boundary conditions. We will also mention preliminary results on a network inspired by a real road network with 20 controllable junctions and 297 links. One may want to optimize a network of this size when adjusting control plans in response to localized extreme conditions, such as an accident creating a traffic jam.

Unless otherwise stated, the key parameters that we will use in our tests are a cycle time of 30 seconds, an optimization horizon of 5 cycles, an initial control plan where phase time is split evenly between phases, upper and lower phase bounds of 80% and 20% of the cycle time, and link parameters that are physically reasonable for an urban road.

There are no permitted phases in our test network so the α variables take the value 0 or 1.

6.1 Algorithmic performance

The alternating directions based algorithm is not guaranteed to increase the objective function from one iteration to the next. In fact the nature of our flow averaging simplification (which was what allowed us to reduce to a knapsack problem) means that the algorithm might not converge to just one control plan, but rather to a periodic oscillation between a number of control plans. In this case we take the control plan that maximizes the objective function. In the examples tested convergence occurred within a small number of iterations (generally less than 5, even on the largest problems).

In order to circumvent this undesirable oscillation property we have investigated more complicated algorithms which undo changes to the control plan if they do not lead to an improvement

# junctions	# cycles	CPU time for our algorithm (s)	CPU time 'fast decomposition' (s)
4	1	0.58	7.89
4	2	0.95	17.62
4	4	1.74	27.88
4	6	2.30	76.23
1	6	0.63	9.17
2	6	1.35	32.40
4	6	2.30	76.23

TABLE 1 : The average CPU time for our algorithm and the 'fast decomposition' algorithm on problems of different dimensions. Observe that our algorithm is approximately an order of magnitude faster.

in the objective function. However such algorithms do not result in control plans with significantly better performance. We will demonstrate in Subsection 6.3 that the naive alternating directions approach nevertheless produces control plans that are effective at improving traffic conditions.

6.2 Runtime performance

In deriving our algorithm we make a number of simplifications with the intention of decreasing the computational cost without sacrificing control performance. We will demonstrate that these simplifications allow us to achieve a runtime that is an order of magnitude faster than the related 'fast decomposition' algorithm from Tang et al. (*19*).

This algorithm works in a similar way to ours, alternating between solving a control problem and simulating traffic. However instead of solving the simplified control problem (51) it solves a linear programming problem similar to (44), but continuous rather than binary. We implement the algorithm from Tang et al. (*19*) using our slightly different forward model (we use actual flow at controlled junctions rather than averaged flow) and compare runtime performance against our proposed algorithm.

The computational cost of the forward simulation (using the CTM) scales linearly in time and space, and at each iteration we simulate $O(|\mathscr{J}_c|)$ times. This cost is the same for both algorithms and appears to be dominated by the cost of the control subproblem, which is different for each algorithm. At each iteration of our algorithm we solve $|\mathscr{J}_c| \times \Upsilon$ knapsack problems each with computational cost independent of the number of junctions and cycles Υ (which we suppose here, for simplicity, is the same for all junctions). So the total computational cost of solving the control problems is $O(|\mathscr{J}_c| \times |\Upsilon|)$. In contrast the 'fast decomposition' algorithm requires solving a linear programming problem, which we would expect to have computational cost greater than $O(|\mathscr{J}_c| \times \Upsilon)$. We can see in Table 1 that the runtime of our algorithm is approximately an order of magnitude faster than the 'fast decomposition' algorithm. This comparison was made on networks consisting of different numbers of controllable junctions that are otherwise similar to our previously mentioned synthetic example.

Note that this performance scales to networks of a size big enough for traffic management applications. Preliminary tests on a realistic test example with 20 controllable junctions and 297 links achieved an average CPU time of 12.18 seconds, which is fast enough for real-time use.

In order to ensure a fair comparison we used the same implementation apart from the optimization routine for the control problem. The linear program in the 'fast decomposition' algorithm

was solved using the IBM ILOG CPLEX Optimizer CPL (*26*). An average of the computation time was taken for 50 different optimization runs with randomly sampled boundary conditions, though the boundary conditions did not significantly affect the computation time. The computations were performed on a standard laptop.

6.3 Control performance (non-incident situation)

We compute two statistics to demonstrate the performance of our optimized control plans. The total link outflow (over all timesteps) and the total delay, which we define as

$$\sum_{\text{timesteps}} \sum_{\text{cells}} \#\text{ vehicles that cannot move to next cell (due to red lights, congestion etc)}\ .$$

We can compute the above statistics for different control plans in order to compare their performance. Note that the relative performance of different control plans likely varies for different networks and boundary and initial conditions. We will consider the synthetic network from Figure 4 and generate different instances of the problem by randomly sampling different boundary conditions (i.e. randomly and independently sample each source flow uniformly from $(0, j)$, where j is the capacity of the link). The affect of different initial conditions is negligible over our 30 minute simulation, so we initialize to zero traffic in the network.

The base control plan for our comparison is one that evenly splits the cycle time between phases. This is also the initial control plan for our optimization algorithm. We compare the base control plan against the optimized control plan on 200 different randomly generated instances. The percentage improvements can be seen in the histograms in Figure 5. In particular, we observe an improvement in both statistics for all instances, with a median flow increase of 6.6% and a median delay decrease of 26%. Preliminary results show comparable performance on our realistic network with 20 controllable junctions and 297 links.

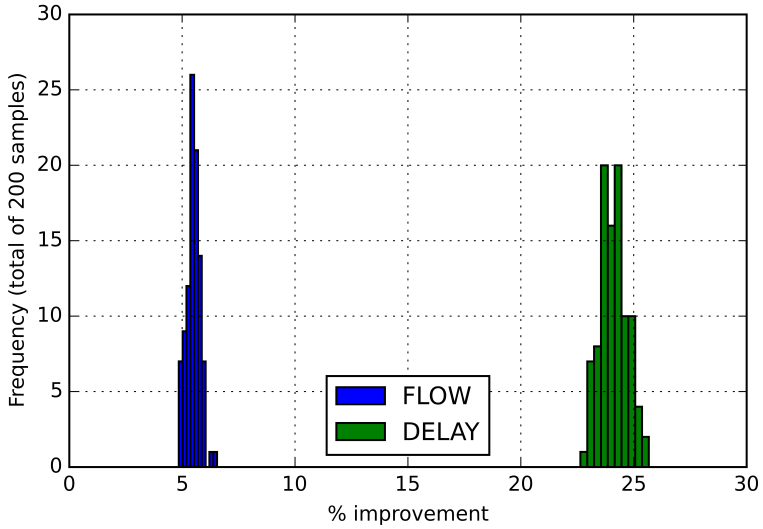


FIGURE 5 : Percentage improvement in flow and delay for 200 instances of the synthetic example in Figure 4 with 4 controllable junctions.

One might claim the junction model we use in this paper does not provide a realistic enough replication of traffic dynamics for validation purposes. We tested a junction flow model than has fixed splitting rates, similar to that in Tang et al. (*19*), however this did not significantly change the improvement statistics. In this case median flow increased by 5.5% (in comparison to 6.6%) and median delay decreased by 24% percent (compared to 26%).

Note that taking different values for the upper and lower bounds on phase time can alter the performance of the optimized control plans. In particular, if we have no upper and lower bounds the algorithm will assign all cycle time to just a single phase. In addition to being undesirable from a network management point of view (drivers may get frustrated), this can lead to a decrease in performance compared to the even split base control plan. However, taking nonzero lower bounds on phase times was generally sufficient to ensure improved performance. In practice these bounds could be calibrated using historical data or operator experience.

6.4 Control performance (incident situation)

In the previous section we saw that our optimized control plans allow the road network to react to varying boundary conditions (source flows), leading to improved traffic management compared to a fixed base control plan. Similarly, the road network parameters can vary when incidents reduce the capacity of links.

We suppose an incident reduces the capacity of a central link in the synthetic network to 20% of its original capacity half way though our simulation. We perform our optimization with 100 randomly sampled but fixed in time source flows. The median time averaged drop in total flow before and after the incident is 14.0 % using the base plan compared to 11.6% using the optimized control plan. Similarly we the median time averaged increase in delay before and after the incident is 55.8 % using the base plan compared to 32.5% using the optimized control plan.

7 CONCLUSION

We started with the complicated nonlinear control problem of optimizing traffic lights in a network in order to maximize controller flow. Some simplifying assumptions led us to a fast algorithm for achieving this, which can be used in a real-time model predictive control type way. We have demonstrated that our algorithm is an order of magnitude faster than a related algorithm from the literature. In all tested cases the algorithm produced a control plan that performed better than a naive even split control plan in terms of flow and delay improvement. Our next step is to verify that our results hold on larger networks derived from real world data.

REFERENCES

[1] Lighthill, M. J. and G. B. Whitham, On kinematic waves. II. A theory of traffic flow on long crowded roads. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, Vol. 229, No. 1178, 1955, pp. 317–345.

[2] Richards, P. I., Shock waves on the highway. *Operations Research*, Vol. 4, No. 1, 1956, pp. 42–51.

[3] Lebacque, J.-P., Intersection modeling, application to macroscopic network traffic flow models and traffic management. In *Traffic and Granular Flow 03*, 2005, pp. 261–278.

[4] Garavello, M. and B. Piccoli, *Traffic Flow on Networks*. Applied Mathematics, American Institute of Mathematical Sciences, 2006.

[5] Daganzo, C. F., The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B: Methodological*, Vol. 28, No. 4, 1994, pp. 269–287.

[6] Daganzo, C. F., The cell transmission model, part II: Network traffic. *Transportation Research Part B: Methodological*, Vol. 29, No. 2, 1995, pp. 79–93.

[7] LeVeque, R. J., *Numerical Methods for Conservation Laws*. Lectures in Mathematics, Birkhäuser, 1992.

[8] Lebacque, J.-P., The Godunov scheme and what it means for first order traffic flow models. *Internaional symposium on transportation and traffic theory*, 1996, pp. 647–677.

[9] Szeto, W. Y., B. Ghosh, B. Basu, and M. O. Mahony, Cell Transmission Model and SARIMA Model. *Journal of Transportational Engineering*, Vol. 135, No. 9, 2009, pp. 658–667.

[10] García, C. E., D. M. Prett, and M. Morari, Model predictive control: Theory and practice - A survey. *Automatica*, Vol. 25, No. 3, 1989, pp. 335–348.

[11] Hegyi, A., B. De Schutter, H. Hellendoorn, and T. van den Boom, Optimal coordination of ramp metering and variable speed control - An MPC approach. In *Proceedings of the 2002 American Control Conference*, IEEE, 2002, Vol. 5, pp. 3600–3605.

[12] Hegyi, A., B. De Schutter, and H. Hellendoorn, Model predictive control for optimal coordination of ramp metering and variable speed limits. *Transportation Research Part C: Emerging Technologies*, Vol. 13, No. 3, 2005, pp. 185–209.

[13] Lin, S., B. De Schutter, Y. Xi, and H. Hellendoorn, Efficient network-wide model-based predictive control for urban traffic networks. *Transportation Research Part C: Emerging Technologies*, Vol. 24, 2012, pp. 122–140.

[14] Gomes, G. and R. Horowitz, Optimal freeway ramp metering using the asymmetric cell transmission model. *Transportation Research Part C: Emerging Technologies*, Vol. 14, No. 4, 2006, pp. 244–262.

[15] Jacquet, D., J. Jaglin, D. Koenig, and C. Canudas De Wit, Non-local feedback ramp metering controller design. In *Proceedings of the 11th IFAC symposium on control in transportation system*, 2006.

[16] Papamichail, I., A. Kotsialos, I. Margonis, and M. Papageorgiou, Coordinated ramp metering for freeway networks - A model-predictive hierarchical control approach. *Transportation Research Part C: Emerging Technologies*, Vol. 18, No. 3, 2010, pp. 311–331.

[17] Lu, X.-Y., P. Varaiya, R. Horowitz, D. Su, and S. Shladover, A new approach for combined freeway Variable Speed Limits and Coordinated Ramp Metering. In *13th International IEEE Conference on Intellligent Transportation Systems*, 2010, pp. 491–498.

[18] Lo, H. K., A novel traffic signal control formulation. *Transportation Research Part A: Policy and Practice*, Vol. 33, No. 6, 1999, pp. 433–448.
[19] Tang, X., S. Blandin, and L. Wynter, A fast decomposition approach for traffic control. In *Preprints of the 19th World Congress The International Federation of Automatic Control*, 2014, pp. 5109–5144.
[20] Wongpiromsarn, T., T. Uthaicharoenpong, Y. Wang, E. Frazzoli, and D. Wang, Distributed traffic signal control for maximum network throughput. *IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2012, pp. 588–595.
[21] Varaiya, P., The max-pressure controller for arbitrary networks of signalized intersections. *Advances in Dynamic Network Modeling in Complex Transportation Systems Complex Networks and Dynamic Systems*, Vol. 2, 2013, pp. 27–66.
[22] Lebacque, J.-P., First-order macroscopic traffic flow models: Intersection modeling, network modeling. In *Transportation and Traffic Theory. Flow, Dynamics and Human Interaction. 16th International Symposium on Transportation and Traffic Theory*, 2005.
[23] Holden, H. and N. H. Risebro, A mathematical model of traffic flow on a network of unidirectional roads. *SIAM Journal on Mathematical Analysis*, Vol. 26, No. 4, 1995, pp. 999–1017.
[24] Jabari, S. E., Some remarks on first-order network flow models. *Preprint*, 2015.
[25] Daganzo, C. F., A finite difference approximation of the kinematic wave model of traffic flow. *Transportation Research Part B: Methodological*, Vol. 29, No. 4, 1995, pp. 261–276.
[26] *IBM ILOG CPLEX Optimization Studio website (www-01.ibm.com/software/commerce/optimization/cplex-cp-optimizer/index.html)*, 2015.