K.U.Leuven

Department of Electrical Engineering (ESAT) **SISTA**

Technical report 96-90

Optimal traffic light control for a single intersection*

B. De Schutter and B. De Moor

If you want to cite this report, please use the following reference instead:
B. De Schutter and B. De Moor, "Optimal traffic light control for a single intersection," *European Journal of Control*, vol. 4, no. 3, pp. 260–276, 1998.

ESAT-SISTA
K.U. Leuven
Kardinaal Mercierlaan 94
B-3001 Leuven (Heverlee)
Belgium
phone: +32-16-32.17.09 (secretary)
fax: +32-16-32.19.70
URL: `http://www.esat.kuleuven.ac.be/sista`

*This report can also be downloaded via `http://lcewww.et.tudelft.nl/~deschutt/pub/abstracts/96_90_abstr.html`

Optimal traffic light control for a single intersection

Bart De Schutter* Bart De Moor†

ESAT-SISTA, K.U.Leuven, Kardinaal Mercierlaan 94,
B-3001 Leuven (Heverlee), Belgium.
email: bart.deschutter@esat.kuleuven.ac.be,
bart.demoor@esat.kuleuven.ac.be

Abstract

In this paper we consider an intersection of two two-way streets with controllable traffic lights on each corner. We construct a model that describes the evolution of the queue lengths (as continuous variables) in each lane as a function of time. We discuss how optimal and suboptimal traffic light switching schemes (with possibly variable cycle lengths) for this system can be determined. We show that for a special class of objective functions, an optimal traffic light switching scheme can be computed very efficiently. The main difference of the approach presented in this paper and most other existing methods is that we allow the green-amber-red cycle time to vary from one cycle to another.

Keywords: traffic, optimal control, optimal traffic light switching scheme

1 Introduction

1.1 Overview

As the number of vehicles and the need for transportation grow, cities around the world face serious road traffic congestion problems. On some major roads this can lead to speeds below 10 km/h, slower than a bicycle. Costs include lost work and leisure time, increased fuel consumption, air pollution, health problems, stress and discomfort. Furthermore, congestion slows the movement of goods and services, thereby increasing the price of products and reducing the competitiveness of business[1]. In general there exist different methods to tackle the traffic congestion problem:

- construction of new roads (to eliminate the most important bottle-necks or to realize missing links),

---

*Senior research assistant with the F.W.O. (Fund for Scientific Research-Flanders)

†Research associate with the F.W.O.

[1] In Hong Kong, the Hang Seng Bank computed that a 10-minute delay each day for every worker in Hong Kong would lead to a loss in output of about US$ 2.28 billion a year, or 2% of the Gross Domestic Product [10].

- stimulating alternatives by promoting public transportation and greater vehicle occupancy or by appropriate pricing and taxing,
- reducing demand by raising tolls or other taxes, or by developing a high-speed communication network, which for many purposes could replace the need for travel,
- better use of existing infrastructure by pricing (time-of-the-day dependent tolls), influencing behavior (promoting car-pooling and measures to organize joint transportation for the employees of a single company or of a set of neighboring companies) and by better control of traffic in order to augment the capacity.

In the short term the most effective measures in the battle against traffic congestion seem to be a selective construction of new roads and a better control of traffic through traffic management.

One approach in traffic management is the use of an automated highway system in order to maximize the throughput of the system by using a platooning approach (see, e.g., [8, 22]). This approach has been studied extensively in the framework of the PATH[2] project. In this approach cars travel on the highway in platoons with small distances (e.g., 1 m) between vehicles within the platoon, and much larger distances (e.g., 30–60 m) between different platoons. Due to the very short intra-platoon distance this approach requires automated distance keeping since a human driver cannot react fast enough to guarantee adequate safety. The vehicle should contain an automatic system that can take over the driver's responsibilities in steering, braking and throttle control. As a consequence, this approach requires major investments to be made both by the government or the traffic authority body (for the roadside control infrastructure that controls the composition and movements of platoons, and the interaction between the different platoons) and by the constructors and owners of the vehicles (for the hardware and software on-board: sensors (to measure the distance to the other vehicles), telecommunication systems (to communicate with the other vehicles) and control systems (to maintain the right distance to the other vehicles)).

Another approach in traffic management is based on a system where most of the intelligence is not located in the individual cars but along the roads. As a consequence, this approach is feasible in the short term and can be implemented by the local traffic authorities. In this approach we try to regulate and redirect the traffic flow using measures such as variable message signs or dynamic route information panels (that show appropriate speeds, preferred directions or information on the length and the duration of traffic jams), ramp metering (i.e., putting traffic lights at highway access-roads to control the incoming traffic flow on the highway), etc. One of the methods that can be used in this traffic management approach is traffic light control to augment the flow of traffic in urban environments by providing a smooth circulation of the traffic or by using "green" waves, or to regulate the access to highways or main roads.

In this paper we study the optimal traffic light control problem for an intersection of two two-way streets. We shall derive a model that describes the evolution of the queue lengths as a function of the time and the switching time instants. The input data for our model are the arrival and departure rates of the vehicles at the intersection. These quantities can be determined by electric induction loop detectors, ultrasonic sensors or vision systems. It is

[2] Partners for Advanced Transit and Highways (see `http://www-path.eecs.berkeley.edu` for more information)

obvious that in practice this sensor infrastructure should be low-cost, high-performance and reliable.

In general, when we are making models of systems, a very important issue is the trade-off between the accuracy of the model and the (computational) complexity of the analysis of the given model. As a rule of thumb we could say that the more accurate the model is, the less we can analytically say about its properties. There are already many possible models for modeling traffic systems: some are mathematical (see, e.g., [9, 14] and the references therein), others are computer models (such as, e.g., SATURN[3] or RONETS[4]). In this paper we shall derive a model that belongs to the former category. First we derive an exact model for the evolution of the queue lengths, but in order to obtain a simplified model that will be the basis of efficient methods to compute optimal traffic light schemes, we make some simplifying assumptions. We shall explain why and when most of these assumptions are reasonable and how deviations of these assumptions can be taken into account. Starting from our simplified model we can then compute the traffic light switching scheme that minimizes a criterion such as average queue length, worst case queue length, average waiting time, ..., thereby augmenting the flow of traffic and diminishing the effects of traffic congestion. The main difference of the approach presented in this paper as compared to other approaches is the use of an optimization over a fixed number of switch-overs instead of an optimization over a fixed number of time steps. This allows to optimize not only the split but also the cycle time with continuous optimization variables (usually the optimization of split and cycle time is performed using boolean variables at each time step, each variable corresponding to the decision of switching or not the traffic lights as in PRODYN, UTOPIA and OPAC).

This paper is organized as follows. In Section 2 we describe the set-up of the intersection and we derive a model that describes the queue lengths, i.e., the number of cars that are waiting, as a function of time, the average arrival and departure rates, and the traffic light switching time instants. First we give the exact model. Next we make some assumptions in order to obtain a model that is more amenable to mathematical analysis. In Section 3 we discuss how an optimal traffic light switching scheme for this system can be determined. We show that for a special class of objective functions (i.e., objective functions that depend strictly monotonously on the queue lengths at the traffic light switching time instants), the optimal traffic light switching scheme can be computed very efficiently. Moreover, if the objective function is linear, the problem reduces to a linear programming problem. In Section 4 we discuss some extensions of the basic model. We conclude with an illustrative example in Section 5.

1.2 Notation

In this paper we use vector as a synonym for "column vector" or "matrix with one column". Vectors and matrices will be represented by boldface symbols. Let $\boldsymbol{a}$ and $\boldsymbol{b}$ vectors with n components. We use a_i or $(\boldsymbol{a})_i$ to denote the ith component of $\boldsymbol{a}$. We use $\boldsymbol{a} \geqslant \boldsymbol{b}$ to indicate that $a_i \geqslant b_i$ for all i. The maximum operator on vectors is defined as follows: $\big(\max(\boldsymbol{a}, \boldsymbol{b})\big)_i = \max(a_i, b_i)$ for all i. A zero column vector with n components is represented by $\mathbf{0}_n$. If the dimension of the zero vector is omitted, it should be clear from the context.

The set of the real numbers is denoted by $\mathbb{R}$, the set of the nonnegative real numbers is denoted by $\mathbb{R}^+$, and the set of the positive real numbers is denoted by $\mathbb{R}_0^+$.

[3]Simulation and Assignment of Traffic to Urban Road Networks

[4]Road Network Simulator

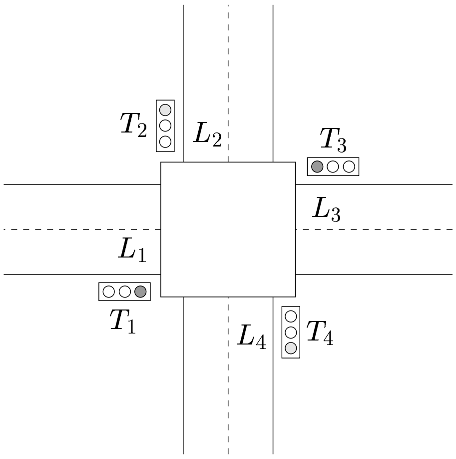

Figure 1: A traffic light controlled intersection of two two-way streets.

2 The set-up and the model of the system

2.1 The set-up

We consider an intersection of two two-way streets (see Figure 1). There are four lanes L_1, L_2, L_3 and L_4, and on each corner of the intersection there is a traffic light (T_1, T_2, T_3 and T_4). For each traffic light there are three subsequent phases: green, amber, and red. The arrival rate of cars in lane L_i at time t is $\lambda_i(t)$ for $i = 1, 2, 3, 4$. So the function λ_i consists of a series of Dirac impulses where a Dirac impulse at time t_i corresponds to the arrival of a car at time t_i and where each Dirac impulse has area 1. When the traffic light is green (respectively amber), the departure rate in lane L_i at time t is $\mu_i(t)$ (respectively $\kappa_i(t)$) for $i = 1, 2, 3, 4$. We assume that the duration of the amber phase is fixed[5] and equal to δ_{amb}.

Let t_0, t_2, t_4, ... be the time instants at which the traffic lights T_1 and T_3 switch from amber to red (and T_2 and T_4 switch from red to green). Let $t_1, t_3, t_5, \ldots$ be the time instants at which the traffic lights T_1 and T_3 switch from red to green (and T_2 and T_4 switch from amber to red). The traffic light switching scheme is shown in Table 1. Let $\mathcal{G}_1 = \mathcal{G}_3 = \{1, 3, \ldots\}$ and $\mathcal{G}_2 = \mathcal{G}_4 = \{0, 2, \ldots\}$. So if $k \in \mathcal{G}_i$ then traffic light T_i switches from red to green at time t_k and is green in the period $(t_k, t_{k+1} - \delta_{\text{amb}})$. Define $\delta_k = t_{k+1} - t_k$ for $k \in \mathbb{N}$. It is obvious that we should have $0 \leqslant \delta_{\text{amb}} < \delta_k$ for all k. Furthermore, $\lambda_i(t), \mu_i(t), \kappa_i(t) \geqslant 0$ for all i and all t, and $t_k < t_{k+1}$ for all k.

Let $l_i(t)$ be the queue length (i.e., the number of cars waiting) in lane L_i at time instant t. Clearly, $l_i(t) \geqslant 0$ for all i and all t.

2.2 The exact model

Consider lane L_1. When the traffic light T_1 is red, there are arrivals at lane L_1 (characterized by the arrival rate function λ_1) and no departures. When the traffic light T_1 is green (or amber), there are arrivals and departures at lane L_1. In that case the net queue growth rate at time t is $\lambda_1(t) - \mu_1(t)$ (respectively $\lambda_1(t) - \kappa_1(t)$). Hence, the evolution of the queue length

[5] In many countries the amber time is fixed by regulation (e.g., to 3 s in France). However, our model can also accommodate varying amber durations (see Section 4).

Period	T_1	T_2	T_3	T_4
t_0 – $t_1-\delta_{\text{amb}}$	red	green	red	green
$t_1-\delta_{\text{amb}}$ – t_1	red	amber	red	amber
t_1 – $t_2-\delta_{\text{amb}}$	green	red	green	red
$t_2-\delta_{\text{amb}}$ – t_2	amber	red	amber	red
t_2 – $t_3-\delta_{\text{amb}}$	red	green	red	green
$t_3-\delta_{\text{amb}}$ – t_3	red	amber	red	amber
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

Table 1: The traffic light switching scheme.

in lane L_1 is given by

$$\frac{dl_1(t)}{dt} = \begin{cases} \lambda_1(t) & \text{if } t \in (t_{2k}, t_{2k+1}) \\ \lambda_1(t) - \mu_1(t) & \text{if } t \in (t_{2k+1}, t_{2k+2}-\delta_{\text{amb}}) \\ \lambda_1(t) - \kappa_1(t) & \text{if } t \in (t_{2k+2}-\delta_{\text{amb}}, t_{2k+2}) \end{cases} \tag{1}$$

for $k = 0, 1, 2, \ldots$ This implies that the relation between the queue lengths at the switching time instants is given by:

$$l_1(t_{2k+1}) = l_1(t_{2k}) + \int_{t_{2k}}^{t_{2k+1}} \lambda_1(t)\, dt \tag{2}$$

$$l_1(t_{2k+2}) = l_1(t_{2k+1}) + \int_{t_{2k+1}}^{t_{2k+2}-\delta_{\text{amb}}} (\lambda_1(t) - \mu_1(t))\, dt + \int_{t_{2k+2}-\delta_{\text{amb}}}^{t_{2k+2}} (\lambda_1(t) - \kappa_1(t))\, dt \tag{3}$$

for $k = 0, 1, 2, \ldots$ We can write down similar equations for the evolution of the queue lengths in the other lanes.
Note that since the functions λ_i, μ_i and κ_i are series of Dirac impulses, the functions l_i are piecewise-constant functions.

2.3 A simplified model

The model derived above is not really amenable to mathematical analysis. Therefore, we shall now introduce some assumptions that will result in a much simpler (approximate) model that can be analyzed very easily and for which we can efficiently compute optimal traffic light switching schemes (see Section 3).

From now on we make the following assumptions:

- the queue lengths are continuous variables,
- the average arrival and departure rates of the cars are constant (or slowly time-varying),
- for each lane the average departure rate during the green phase is greater than or equal to the average departure rate during the amber phase.

The first two assumptions deserve a few remarks:

- The main purpose of the model we shall derive is to compute optimal traffic light control schemes. Designing optimal traffic light switching schemes is only useful if the arrival and departure rates of vehicles at the intersection are high. In that case, approximating the queue lengths by continuous variables only introduces small errors. Furthermore, there is also some uncertainty and variation in time of the arrival and departure rates, which makes that in general computing the exact optimal traffic light switching scheme is utopian. Moreover, in practice we are more interested in quickly obtaining a good approximation of the optimal traffic light switching scheme than in spending a large amount of time to obtain the exact optimal traffic light switching scheme.

- If we keep in mind that one of the main purposes of the model that we shall derive, is the design of optimal traffic light switching schemes, then assuming that the average arrival and departure rates are constant is not a serious restriction, provided that we use a *moving horizon strategy*: we compute the optimal traffic light switching scheme for, say, the next 10 cycles, based on a prediction of the average arrival and departure rates (using data measured during the previous cycles) and we apply this scheme during the first of the 10 cycles, meanwhile we update our estimates of the arrival and departure rates and compute a new optimal scheme for the next 10 cycles, and so on. This will be discussed more extensively in Section 3.

Let $\bar{\lambda}_i$ be the average arrival rate of vehicles in lane L_i. Let $\bar{\mu}_i$ (respectively $\bar{\kappa}_i$) be the average departure rate in lane L_i when the traffic light is green (respectively amber) and the queue length is larger than 0 (i.e., when there are cars waiting or arriving at lane L_i). So from now on we assume that

$$
\begin{aligned}
\lambda_i(t) &= \bar{\lambda}_i \qquad \text{for all } t, \\
\mu_i(t) &= \begin{cases} \bar{\mu}_i & \text{if } t \in \bigcup\limits_{k \in \mathcal{G}_i} (t_k, t_{k+1} - \delta_{\text{amb}}) \text{ and } l_i(t) > 0, \\ 0 & \text{otherwise,} \end{cases} \\
\kappa_i(t) &= \begin{cases} \bar{\kappa}_i & \text{if } t \in \bigcup\limits_{k \in \mathcal{G}_i} (t_{k+1} - \delta_{\text{amb}}, t_{k+1}) \text{ and } l_i(t) > 0, \\ 0 & \text{otherwise.} \end{cases}
\end{aligned}
$$

Note that $\bar{\lambda}_i$, $\bar{\mu}_i$ and $\bar{\kappa}_i$ are constants (possibly over a limited period of time if we apply a moving horizon strategy (see Section 3)). We have $\bar{\lambda}_i, \bar{\mu}_i, \bar{\kappa}_i \geqslant 0$ and $\bar{\kappa}_i \leqslant \bar{\mu}_i$.

Note that if we consider (1) and (2) – (3), then the queue length can never become negative. However, if we use average arrival and departure rates in the description of the evolution of the queue lengths then we have to include the nonnegativity condition explicitly when writing down the (approximate) evolution equations for the queue lengths. As a consequence, we have

$$
\begin{aligned}
l_1(t_{2k+1}) &= l_1(t_{2k}) + \bar{\lambda}_1 \delta_{2k} \\
l_1(t_{2k+2} - \delta_{\text{amb}}) &= \max\big(l_1(t_{2k+1}) + (\bar{\lambda}_1 - \bar{\mu}_1)\,(\delta_{2k+1} - \delta_{\text{amb}}), 0\big) && (4) \\
l_1(t_{2k+2}) &= \max\big(l_1(t_{2k+2} - \delta_{\text{amb}}) + (\bar{\lambda}_1 - \bar{\kappa}_1)\,\delta_{\text{amb}}, 0\big) && (5) \\
&= \max\big(l_1(t_{2k+1}) + (\bar{\lambda}_1 - \bar{\mu}_1)\,\delta_{2k+1} + (\bar{\mu}_1 - \bar{\kappa}_1)\,\delta_{\text{amb}}, (\bar{\lambda}_1 - \bar{\kappa}_1)\,\delta_{\text{amb}}, 0\big)
\end{aligned}
$$

for $k = 0, 1, 2, \ldots$ Note that we also have

$$
l_1(t_{2k+1}) = \max\big(l_1(t_{2k}) + \bar{\lambda}_1 \delta_{2k}, 0\big) \qquad \text{for } k = 0, 1, 2, \ldots,
$$

since $l_1(t) \geqslant 0$ for all t.
We can write down similar equations for $l_2(t_k)$, $l_3(t_k)$ and $l_4(t_k)$. So if we define

$$\boldsymbol{x}_k = \begin{bmatrix} l_1(t_k) \\ l_2(t_k) \\ l_3(t_k) \\ l_4(t_k) \end{bmatrix}, \quad \boldsymbol{b}_1 = \begin{bmatrix} \bar{\lambda}_1 \\ \bar{\lambda}_2 - \bar{\mu}_2 \\ \bar{\lambda}_3 \\ \bar{\lambda}_4 - \bar{\mu}_4 \end{bmatrix}, \quad \boldsymbol{b}_2 = \begin{bmatrix} \bar{\lambda}_1 - \bar{\mu}_1 \\ \bar{\lambda}_2 \\ \bar{\lambda}_3 - \bar{\mu}_3 \\ \bar{\lambda}_4 \end{bmatrix},$$

$$\boldsymbol{b}_3 = \begin{bmatrix} 0 \\ (\bar{\mu}_2 - \bar{\kappa}_2)\,\delta_{\mathrm{amb}} \\ 0 \\ (\bar{\mu}_4 - \bar{\kappa}_4)\,\delta_{\mathrm{amb}} \end{bmatrix}, \quad \boldsymbol{b}_4 = \begin{bmatrix} (\bar{\mu}_1 - \bar{\kappa}_1)\,\delta_{\mathrm{amb}} \\ 0 \\ (\bar{\mu}_3 - \bar{\kappa}_3)\,\delta_{\mathrm{amb}} \\ 0 \end{bmatrix},$$

$$\boldsymbol{b}_5 = \begin{bmatrix} 0 \\ \max\big((\bar{\lambda}_2 - \bar{\kappa}_2)\,\delta_{\mathrm{amb}},\, 0\big) \\ 0 \\ \max\big((\bar{\lambda}_4 - \bar{\kappa}_4)\,\delta_{\mathrm{amb}},\, 0\big) \end{bmatrix}, \quad \boldsymbol{b}_6 = \begin{bmatrix} \max\big((\bar{\lambda}_1 - \bar{\kappa}_1)\,\delta_{\mathrm{amb}},\, 0\big) \\ 0 \\ \max\big((\bar{\lambda}_3 - \bar{\kappa}_3)\,\delta_{\mathrm{amb}},\, 0\big) \\ 0 \end{bmatrix},$$

then we have

$$\boldsymbol{x}_{2k+1} = \max\left(\boldsymbol{x}_{2k} + \boldsymbol{b}_1 \delta_{2k} + \boldsymbol{b}_3,\, \boldsymbol{b}_5\right) \tag{6}$$

$$\boldsymbol{x}_{2k+2} = \max\left(\boldsymbol{x}_{2k+1} + \boldsymbol{b}_2 \delta_{2k+1} + \boldsymbol{b}_4,\, \boldsymbol{b}_6\right) \tag{7}$$

for $k = 0, 1, 2, \ldots$ We say that the sequences $\{\boldsymbol{x}_k\}_{k=0}^{N}$ and $\{\delta_k\}_{k=0}^{N-1}$ are *compatible* (for given δ_{amb}, $\bar{\lambda}_i$'s, $\bar{\mu}_i$'s and $\bar{\kappa}_i$'s) if they satisfy the recurrence equations (6) – (7) for all k.

Remark 2.1 The model we have derived above is different from the models used by most other researchers due to the fact that we consider green-amber-red cycle lengths that may vary from cycle to cycle. Furthermore, we consider non-saturated intersections, i.e., we allow queue lengths to become equal to 0 during the green phase. Some authors (see, e.g., [12, 19]) consider models for oversaturated intersections, i.e., they do not allow queue lengths to become equal to 0 during the green phase. In that case the maximum operator that appears in (4) and (5) — and in (6) – (7) — is not necessary any more, which leads to a simpler description of the behavior of the system. However, it can be shown that applying a model for oversaturated intersections to a non-saturated intersection in general does not lead to an optimal traffic light switching scheme (see [4]). Furthermore, in Section 3 we shall show that by considering some approximate objective functions we can find suboptimal traffic light switching schemes using a simplified description of the system in which the maximum operator does not appear. Nevertheless, the suboptimal solutions obtained in this way will satisfy the model (6) – (7). Furthermore, for a non-saturated intersection their performance will be considerably better than the performance of the solutions obtained from a model for an oversaturated intersection (see [4]). ◇

3 Optimal control

3.1 Problem statement

From now on we assume that the arrival and departure rates are known. For a given integer N we want to compute an optimal sequence $t_0,\ t_1,\ \ldots t_N$ of switching time instants that minimizes a given criterion J. Define $\mathcal{G}_i(N) = \mathcal{G}_i \cap \{0, 1, 2, \ldots, N-1\}$. Possible objective functions are:

- (weighted) average queue length over all queues:

$$J_1 = \sum_{i=1}^{4} w_i \frac{\displaystyle\int_{t_0}^{t_N} l_i(t)\, dt}{t_N - t_0} \quad , \tag{8}$$

- (weighted) average queue length over the worst queue:

$$J_2 = \max_i \left(w_i \frac{\displaystyle\int_{t_0}^{t_N} l_i(t)\, dt}{t_N - t_0} \right) \quad , \tag{9}$$

- (weighted) worst case queue length:

$$J_3 = \max_{i,\, t} \big(w_i \, l_i(t) \big) \quad , \tag{10}$$

- (weighted) average waiting time[6] over all queues:

$$J_4 = \sum_{i=1}^{4} w_i \frac{\displaystyle\int_{t_0}^{t_N} l_i(t)\, dt}{\bar{\lambda}_i (t_N - t_0)} \quad , \tag{11}$$

- (weighted) average waiting time over the worst queue:

$$J_5 = \max_i \left(w_i \frac{\displaystyle\int_{t_0}^{t_N} l_i(t)\, dt}{\bar{\lambda}_i (t_N - t_0)} \right) \quad , \tag{12}$$

where $w_i > 0$ for all i. The weight factors w_i can be used to give a higher importance or weight to some lanes: we could, e.g., choose $w_1 = w_3 = 2$ and $w_2 = w_4 = 1$ to give a 100% higher importance to cars on the street consisting of lanes L_1 and L_3.
The factor $t_N - t_0$ in J_1, J_2, J_4 and J_5 seems to be original with respect to the literature. The reason for introducing this factor is that in our approach the time horizon is not fixed. Using criteria based on time averaged values has the advantage that we maintain finite values for the objective functions even if N or t_N go to ∞ (provided that the queue lengths remain finite). Note that J_1 and J_4 are in fact equivalent in the sense that for any weight vector $\boldsymbol{w}$ for J_1 there exists a weight vector $\tilde{\boldsymbol{w}}$ for J_4 such that J_1 and J_4 are equal. This also holds for J_2 and J_5.

[6] The average waiting time is equal to the average queue length divided by the average arrival rate.

We can impose some extra conditions such as minimum and maximum durations for the red and green times[7], maximum queue lengths[8], and so on. This leads to the following problem:

$$\text{minimize } J \tag{13}$$

subject to

$$\delta_{\text{min,green},1} \leqslant \delta_{2k+1} - \delta_{\text{amb}} \leqslant \delta_{\text{max,green},1} \qquad \text{for } k = 0, 1, \ldots, \left\lfloor \frac{N}{2} \right\rfloor - 1, \tag{14}$$

$$\delta_{\text{min,green},2} \leqslant \delta_{2k} - \delta_{\text{amb}} \leqslant \delta_{\text{max,green},2} \qquad \text{for } k = 0, 1, \ldots, \left\lfloor \frac{N-1}{2} \right\rfloor, \tag{15}$$

$$\boldsymbol{x}_k \leqslant \boldsymbol{x}_{\max} \qquad \text{for } k = 1, 2, \ldots, N, \tag{16}$$

$$\boldsymbol{x}_{2k+1} = \max\left(\boldsymbol{x}_{2k} + \boldsymbol{b}_1 \delta_{2k} + \boldsymbol{b}_3, \, \boldsymbol{b}_5\right) \qquad \text{for } k = 0, 1, \ldots, \left\lfloor \frac{N-1}{2} \right\rfloor, \tag{17}$$

$$\boldsymbol{x}_{2k+2} = \max\left(\boldsymbol{x}_{2k+1} + \boldsymbol{b}_2 \delta_{2k+1} + \boldsymbol{b}_4, \, \boldsymbol{b}_6\right) \qquad \text{for } k = 0, 1, \ldots, \left\lfloor \frac{N}{2} \right\rfloor - 1 \ , \tag{18}$$

where $\delta_{\text{min,green},i}$ (respectively $\delta_{\text{max,green},i}$) is the minimum (respectively maximum) green time in lane L_i, and $(\boldsymbol{x}_{\max})_i$ is the maximum queue length in lane L_i. Note that we could also let $\delta_{\text{min,green},i}$, $\delta_{\text{max,green},i}$, and $\boldsymbol{x}_{\max}$ depend on k. In the remainder of this paper, we shall assume that $\boldsymbol{x}_0 \leqslant \boldsymbol{x}_{\max}$.

Now we discuss some methods to solve problem (13) – (18).
First consider (17) for an arbitrary index k. This equation can be rewritten as follows:

$$\boldsymbol{x}_{2k+1} \geqslant \boldsymbol{x}_{2k} + \boldsymbol{b}_1 \delta_{2k} + \boldsymbol{b}_3 \tag{19}$$

$$\boldsymbol{x}_{2k+1} \geqslant \boldsymbol{b}_5 \tag{20}$$

$$(\boldsymbol{x}_{2k+1})_i = (\boldsymbol{x}_{2k} + \boldsymbol{b}_1 \delta_{2k} + \boldsymbol{b}_3)_i \quad \text{or} \quad (\boldsymbol{x}_{2k+1})_i = (\boldsymbol{b}_5)_i \qquad \text{for } i = 1, 2, 3, 4\,, \tag{21}$$

or equivalently

$$\boldsymbol{x}_{2k+1} - \boldsymbol{x}_{2k} - \boldsymbol{b}_1 \delta_{2k} - \boldsymbol{b}_3 \geqslant \boldsymbol{0}$$
$$\boldsymbol{x}_{2k+1} - \boldsymbol{b}_5 \geqslant \boldsymbol{0}$$
$$(\boldsymbol{x}_{2k+1} - \boldsymbol{x}_{2k} - \boldsymbol{b}_1 \delta_{2k} - \boldsymbol{b}_3)_i \, (\boldsymbol{x}_{2k+1} - \boldsymbol{b}_5)_i = 0 \qquad \text{for } i = 1, 2, 3, 4\,.$$

Since a sum of nonnegative numbers is equal to 0 if and only if all the numbers are equal to 0, this system of equations is equivalent to:

$$\boldsymbol{x}_{2k+1} - \boldsymbol{x}_{2k} - \boldsymbol{b}_1 \delta_{2k} - \boldsymbol{b}_3 \geqslant \boldsymbol{0} \tag{22}$$

$$\boldsymbol{x}_{2k+1} - \boldsymbol{b}_5 \geqslant \boldsymbol{0} \tag{23}$$

[7] A green time that is too short is wasteful. If the red time is too long, drivers tend to believe that the signals have broken down.

[8] This could correspond to an upper bound on the available storage space due to the distance to the preceding junction or to the layout of the intersection.

$$\sum_{i=1}^{4} (\boldsymbol{x}_{2k+1} - \boldsymbol{x}_{2k} - \boldsymbol{b}_1 \delta_{2k} - \boldsymbol{b}_3)_i \, (\boldsymbol{x}_{2k+1} - \boldsymbol{b}_5)_i = 0 \ . \tag{24}$$

We can repeat this reasoning for (18) and for each k.
So if we define

$$\boldsymbol{x}^* = \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_N \end{bmatrix} \quad \text{and} \quad \boldsymbol{\delta}^* = \begin{bmatrix} \delta_0 \\ \delta_1 \\ \vdots \\ \delta_{N-1} \end{bmatrix} \ ,$$

it is easy to verify that we finally get a problem of the form

$$\text{minimize } J \tag{25}$$

subject to

$$\boldsymbol{A}\boldsymbol{x}^* + \boldsymbol{B}\boldsymbol{\delta}^* + \boldsymbol{c} \geqslant \boldsymbol{0} \tag{26}$$

$$\boldsymbol{x}^* + \boldsymbol{d} \geqslant \boldsymbol{0} \tag{27}$$

$$\boldsymbol{E}\boldsymbol{x}^* + \boldsymbol{F}\boldsymbol{\delta}^* + \boldsymbol{g} \geqslant \boldsymbol{0} \tag{28}$$

$$(\boldsymbol{A}\boldsymbol{x}^* + \boldsymbol{B}\boldsymbol{\delta}^* + \boldsymbol{c})^T \, (\boldsymbol{x}^* + \boldsymbol{d}) = 0 \ , \tag{29}$$

with appropriately defined matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{E}$, $\boldsymbol{F}$ and vectors $\boldsymbol{c}$, $\boldsymbol{d}$, $\boldsymbol{g}$. Equations (26), (27) and (29) correspond to (22), (23) and (24) respectively, and the system of linear inequalities (28) contains the conditions (14) – (16).

We say that the vectors $\boldsymbol{x}^*$ and $\boldsymbol{\delta}^*$ are *compatible* for a given $\boldsymbol{x}_0$ if the corresponding sequences $\{\boldsymbol{x}_k\}_{k=0}^{N}$ and $\{\delta_k\}_{k=0}^{N-1}$ are compatible. In the remainder of this paper a vector $\boldsymbol{x}^* \in (\mathbb{R}^+)^{4N}$ will always correspond to a sequence $\{\boldsymbol{x}_k\}_{k=1}^{N}$ and vice versa, with $(\boldsymbol{x}^*)_{4(k-1)+j} = (\boldsymbol{x}_k)_j$ for $k = 1, 2, \ldots, N$ and $j = 1, 2, 3, 4$. A vector $\boldsymbol{\delta}^* \in (\mathbb{R}_0^+)^N$ will always correspond to a sequence $\{\delta_k\}_{k=0}^{N-1}$ and vice versa, with $(\boldsymbol{\delta}^*)_k = \delta_{k-1}$ for $k = 1, 2, \ldots, N$.

The system (26) – (29) is (a special case of) an Extended Linear Complementarity Problem (ELCP) [2, 3]. In [2, 3] we have developed an algorithm to compute a parametric description of the complete solution set of an ELCP. If we assume that $\boldsymbol{x}^*$ and $\boldsymbol{\delta}^*$ are bounded from above (i.e., $\delta_{\max,\text{green},i}$, $\delta_{\max,\text{red},i}$ and the components of $\boldsymbol{x}_{\max}$ are defined and different from $+\infty$), then the solution set of the system (26) – (29) is characterized by a set of vectors

$$V = \left\{ \begin{bmatrix} \boldsymbol{x}_i^* \\ \boldsymbol{\delta}_i^* \end{bmatrix} \,\middle|\, i = 1, 2, \ldots, r \right\}$$

and a set of index sets $\Lambda = \{\phi_j \,|\, j = 1, 2, \ldots, p\}$ with $\phi_j \subseteq \{1, 2, \ldots, r\}$ for all j, such that for any j any convex combination of the form

$$\sum_{i \in \phi_j} \nu_i \begin{bmatrix} \boldsymbol{x}_i^* \\ \boldsymbol{\delta}_i^* \end{bmatrix}$$

with $\nu_i \geqslant 0$ and $\sum_{i \in \phi_j} \nu_i = 1$, is a solution of (26) – (29). The vectors of V correspond to vertices of the polyhedron P defined by the system of linear equations and inequalities (26) – (28), and each index set ϕ_j corresponds to a face of this polyhedron.

We could for each index set ϕ_j determine for which combination of the ν_i's the objective function J reaches a global minimum and afterwards select the overall minimum. Our computational experiments have shown that in most cases the determination of the minimum value of the objective functions J_1, J_2, J_4 and J_5 over a face of P is a well-behaved problem in the sense that using a local minimization routine (that uses, e.g., sequential quadratic programming) starting from different initial points almost always yields the same numerical result (within a certain tolerance). Furthermore, it can be shown that J_3 is a convex function of the ν_i's (see [4]).

However, the general ELCP is an NP-hard problem [2, 3]. Furthermore, the vectors of V correspond to vertices of the polyhedron defined by (26) – (28) and it can be shown that in the worst case the number of vertices of a polyhedron defined by p (non-redundant) inequality constraints in an n-dimensional space is $O\left(p^{\lfloor \frac{n}{2} \rfloor}\right)$ if $p \gg n \gg 1$. This implies that the approach sketched above is not feasible if the number of switching cycles N is large (see also Example 5.1).

If N is large, we could consider a small number N_{s} of switching cycles, compute the optimal switching strategy with the method given above, implement the first step(s) of this strategy, and recompute the optimal switching strategy for the next N_{s} switching cycles. Note that although the latter approach is feasible, it will only give a suboptimal solution! In the remainder of this paper we shall call this approach the *suboptimal multi-ELCP approach.*
We could also apply this moving horizon strategy if the average arrival rates $\bar{\lambda}_i$ and the average departure rates $\bar{\mu}_i$ and $\bar{\kappa}_i$ are slowly time-varying: we estimate the average arrival and departure rates over a certain period (using measurements obtained from cameras or loop detectors), compute the optimal control strategy and apply it for a certain number of switching cycles. Meanwhile, we update our estimates of the $\bar{\lambda}_i$'s, $\bar{\mu}_i$'s and the $\bar{\kappa}_i$'s, and then we compute a new optimal control strategy for the updated values of the average arrival and departure rates, and so on. This approach is called *Model-Based Predictive Control* or *Generalized Predictive Control* (see, e.g., [1, 6]).

The objective functions J_1, J_2, J_3, J_4 and J_5 do not explicitly depend on $\boldsymbol{x}^*$ since for given $\boldsymbol{x}_0$, $\bar{\lambda}_i$'s, $\bar{\mu}_i$'s, $\bar{\kappa}_i$'s and δ_{amb}, the components of $\boldsymbol{x}^*$ (i.e., the $\boldsymbol{x}_k$'s) are uniquely determined by $\boldsymbol{\delta}^*$. So if there is no upper bound on $\boldsymbol{x}_k$, then problem (13) – (18) reduces to a constrained optimization problem in $\boldsymbol{\delta}^*$, which could be solved using a constrained minimization procedure. If $\boldsymbol{x}_{\mathrm{max}}$ is finite we could still use this approach by adding an extra penalty term to J if $\boldsymbol{x}_k \geqslant \boldsymbol{x}_{\mathrm{max}}$ (see Example 5.1). The major disadvantage of this approach is that in general[9] the resulting objective function is neither convex nor concave so that the minimization routine will only return a local minimum. Our computational experiments have shown that several initial starting points are necessary to obtain the global minimum.

3.2 The relaxed problem and suboptimal solutions

In this section we shall make some extra approximations that will result in suboptimal traffic light switching schemes that can be computed very efficiently but that approximate the exact

[9] Except for the objective function J_3 with a convex penalty term.

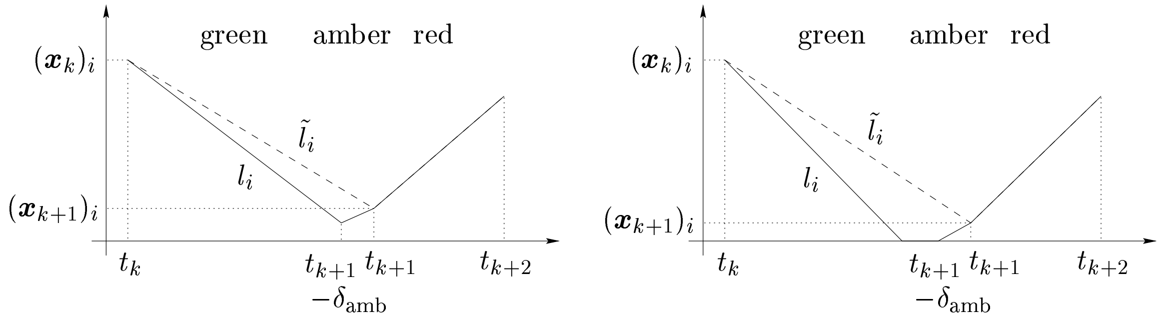

Figure 2: The functions l_i (full line) and $\tilde{l}_i$ (dashed line). During the red phase the functions l_i and $\tilde{l}_i$ coincide. The left plot shows a situation in which the queue length does not become 0 during the green phase, whereas the right plot shows a situation where the queue length becomes 0 during the green phase.

optimum rather well. We use the notation $l_i(\cdot, \boldsymbol{\delta}^*)$ to indicate that the queue length function $l_i(\cdot)$ corresponds to the switching interval vector $\boldsymbol{\delta}^*$.

Now we define some objective functions that can be considered as approximations to the objective functions J_1, J_2, J_3, J_4 and J_5. Although these approximations will lead to suboptimal solutions with respect to J_1, J_2, J_3, J_4 or J_5, the corresponding optimization problems can be solved much more efficiently. As has already been mentioned, we have already introduced an extra approximation by considering continuous queue lengths. Furthermore, in practice there is also some uncertainty and variation in time of the arrival and departure rates, which makes that in general computing the exact optimal traffic light switching scheme is utopian. Moreover, in practice we are more interested in quickly obtaining a good approximation of the optimal traffic light switching scheme than in spending a large amount of time to obtain the exact optimal switching scheme.

Recall that the objective functions J_1, J_2, J_3, J_4 and J_5 do not explicitly depend on $\boldsymbol{x}^*$ since the components of $\boldsymbol{x}^*$ are uniquely determined by $\boldsymbol{\delta}^*$. The approximate objective functions that we introduce now will depend explicitly on $\boldsymbol{x}^*$ and $\boldsymbol{\delta}^*$. For a given $\boldsymbol{x}_0$ and t_0, we define the function $\tilde{l}_i(\cdot, \boldsymbol{x}^*, \boldsymbol{\delta}^*)$ — or $\tilde{l}_i(\cdot)$ for short — as the piecewise-linear function that interpolates in the points $(t_k, l_i(t_k))$ for $k = 0, 1, \ldots, N$.
The approximate objective functions $\tilde{J}_l$ for $l = 1, 2, 3, 4, 5$, are defined as in (8) – (12) but with l_i replaced by $\tilde{l}_i$.
We shall show that the values of J_3 and $\tilde{J}_3$ coincide. Now let $l \in \{1, 2, 4, 5\}$. Note that the value of the objective functions J_l and $\tilde{J}_l$ is determined by the surface under the functions l_i and $\tilde{l}_i$ respectively. If the duration of the amber phase is zero and if the queue lengths never become zero during the green phases, then the functions l_i and $\tilde{l}_i$ and the values of J_l and $\tilde{J}_l$ coincide (cf. Figure 2). In practice, the length of the amber phase will be small compared to the length of the green or the red phase. Furthermore, an optimal traffic light switching scheme implies the absence of long periods in which no cars wait in one lane while in the other lanes the queue lengths increase. So if we have an optimal traffic light switching scheme, then the periods during which the queue length in some lane is equal to 0 are in general short. As a consequence, for traffic light switching schemes in the neighborhood of the optimal scheme $\tilde{J}_l$ will be a good approximation of J_l.

Proposition 3.1 *Let $\boldsymbol{x}_0 \in (\mathbb{R}^+)^4$, $\boldsymbol{x}^* \in (\mathbb{R}^+)^{4N}$ and $\boldsymbol{\delta}^* \in (\mathbb{R}_0^+)^N$. If $\boldsymbol{x}^*$ and $\boldsymbol{\delta}^*$ are compat-*

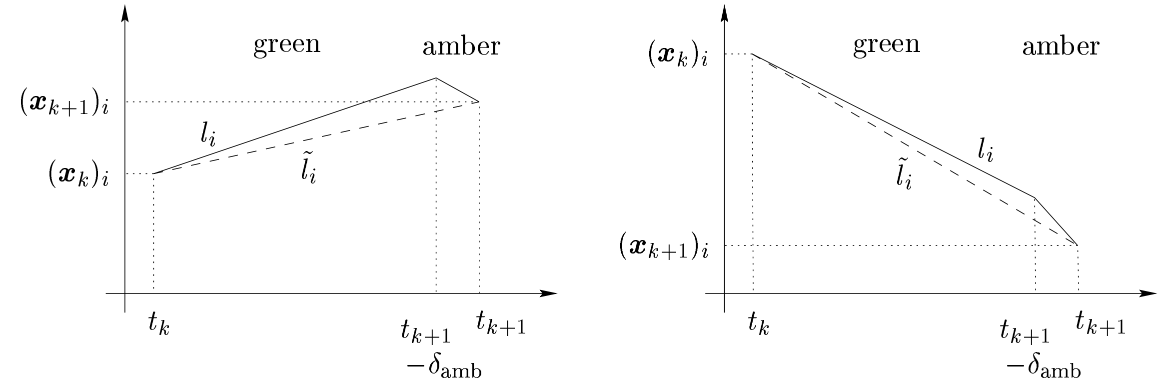

Figure 3: The evolution of the queue length l_i (full line) during a green phase and a subsequent amber phase with $\bar{\lambda}_i - \bar{\kappa}_i < 0$ and $\bar{\lambda}_i - \bar{\mu}_i > 0$ (on the left), and $\bar{\lambda}_i - \bar{\kappa}_i < \bar{\lambda}_i - \bar{\mu}_i < 0$ (on the right). Note that the assumption $\bar{\kappa}_i \leqslant \bar{\mu}_i$ implies that these situations cannot occur. The function $\tilde{l}_i$ is represented by the dashed line.

ible for $\boldsymbol{x}_0$ *then we have* $J_3(\boldsymbol{\delta}^*) = \tilde{J}_3(\boldsymbol{x}^*, \boldsymbol{\delta}^*)$*, and* $J_l(\boldsymbol{\delta}^*) \leqslant \tilde{J}_l(\boldsymbol{x}^*, \boldsymbol{\delta}^*)$ *for* $l = 1, 2, 4, 5$.

Proof: Let $\boldsymbol{x}^*$ and $\boldsymbol{\delta}^*$ be compatible for $\boldsymbol{x}_0$.

We have assumed that $\bar{\kappa}_i \leqslant \bar{\mu}_i$. Hence, $\bar{\lambda}_i - \bar{\kappa}_i \geqslant \bar{\lambda}_i - \bar{\mu}_i$ for all i. This implies that a situation such as in the left plot of Figure 3 where $\bar{\lambda}_i - \bar{\mu}_i > 0$ and $\bar{\lambda}_i - \bar{\kappa}_i < 0$ is not possible. As a consequence, the maxima of the piecewise-linear functions $l_i(\cdot, \boldsymbol{\delta}^*)$ and $\tilde{l}_i(\cdot, \boldsymbol{x}^*, \boldsymbol{\delta}^*)$ coincide. This implies that $J_3(\boldsymbol{\delta}^*) = \tilde{J}_3(\boldsymbol{x}^*, \boldsymbol{\delta}^*)$.

Let $k \in \mathcal{G}_i(N)$. So T_i is green in the interval $(t_k, t_{k+1} - \delta_{\rm amb})$ and amber in $(t_{k+1} - \delta_{\rm amb}, t_{k+1})$. It is easy to verify that $l_i(t, \boldsymbol{\delta}^*) \leqslant \tilde{l}_i(t, \boldsymbol{x}^*, \boldsymbol{\delta}^*)$ for all $t \in (t_k, t_{k+1})$ since $\bar{\kappa}_i \leqslant \bar{\mu}_i$ (cf. Figures 2 and 3).
If $k \leqslant N - 2$ then T_i is red in the interval (t_{k+1}, t_{k+2}) and in this interval l_i and $\tilde{l}_i$ coincide. This implies that $l_i(t, \boldsymbol{\delta}^*) \leqslant \tilde{l}_i(t, \boldsymbol{x}^*, \boldsymbol{\delta}^*)$ for all $t \in [t_0, t_N]$. Hence,

$$\int_{t_0}^{t_N} l_i(t, \boldsymbol{\delta}^*)\, dt \;\leqslant\; \int_{t_0}^{t_N} \tilde{l}_i(t, \boldsymbol{x}^*, \boldsymbol{\delta}^*)\, dt \;\; .$$

As a consequence, we have $J_l(\boldsymbol{\delta}^*) \leqslant \tilde{J}_l(\boldsymbol{x}^*, \boldsymbol{\delta}^*)$ for $l = 1, 2, 4, 5$. □

Now we show that the use of the approximate objective function $\tilde{J}_1$ or $\tilde{J}_4$ leads to an optimization problem that can be solved more efficiently than the original problem in which J_1 or J_4 is used.

Let $\mathcal{P}$ be the problem (13) – (18). We define the "relaxed" problem $\tilde{\mathcal{P}}$ corresponding to the original problem $\mathcal{P}$ as:

$$\text{minimize } J \tag{30}$$

subject to

$$\delta_{\rm min,green,1} \leqslant \delta_{2k+1} - \delta_{\rm amb} \leqslant \delta_{\rm max,green,1} \qquad \text{for } k = 0, 1, \ldots, \left\lfloor \frac{N}{2} \right\rfloor - 1, \tag{31}$$

$$\delta_{\text{min,green},2} \leqslant \delta_{2k} - \delta_{\text{amb}} \leqslant \delta_{\text{max,green},2} \qquad \text{for } k = 0, 1, \ldots, \left\lfloor \frac{N-1}{2} \right\rfloor, \tag{32}$$

$$\boldsymbol{x}_k \leqslant \boldsymbol{x}_{\max} \qquad \text{for } k = 1, 2, \ldots, N, \tag{33}$$

$$\boldsymbol{x}_{2k+1} \geqslant \boldsymbol{x}_{2k} + \boldsymbol{b}_1 \delta_{2k} + \boldsymbol{b}_3 \qquad \text{for } k = 0, 1, \ldots, \left\lfloor \frac{N-1}{2} \right\rfloor, \tag{34}$$

$$\boldsymbol{x}_{2k+1} \geqslant \boldsymbol{b}_5 \qquad \text{for } k = 0, 1, \ldots, \left\lfloor \frac{N-1}{2} \right\rfloor, \tag{35}$$

$$\boldsymbol{x}_{2k+2} \geqslant \boldsymbol{x}_{2k+1} + \boldsymbol{b}_2 \delta_{2k+1} + \boldsymbol{b}_4 \qquad \text{for } k = 0, 1, \ldots, \left\lfloor \frac{N}{2} \right\rfloor - 1, \tag{36}$$

$$\boldsymbol{x}_{2k+2} \geqslant \boldsymbol{b}_6 \qquad \text{for } k = 0, 1, \ldots, \left\lfloor \frac{N}{2} \right\rfloor - 1. \tag{37}$$

Note that compared to the original problem we have replaced (17) – (18) by relaxed equations of the form (19) – (20) without taking (21) into account.

Proposition 3.2 *If J is a strictly monotonous function of $\boldsymbol{x}^*$ — i.e., if for any $\boldsymbol{\delta}^*$ with positive components and for all $\tilde{\boldsymbol{x}}^*, \hat{\boldsymbol{x}}^*$ with $\mathbf{0} \leqslant \tilde{\boldsymbol{x}}^* \leqslant \hat{\boldsymbol{x}}^*$ and with $\tilde{x}^*_j < \hat{x}^*_j$ for at least one index j, we have $J(\tilde{\boldsymbol{x}}^*, \boldsymbol{\delta}^*) < J(\hat{\boldsymbol{x}}^*, \boldsymbol{\delta}^*)$ — then any solution of the relaxed problem $\tilde{\mathcal{P}}$ is also a solution of the original problem $\mathcal{P}$.*

Proof: Let $\tilde{\boldsymbol{x}}^*$, $\tilde{\boldsymbol{\delta}}^*$ be an optimal solution of $\tilde{\mathcal{P}}$. Now we show by contradiction that if J depends strictly monotonously on $\boldsymbol{x}^*$, then $\tilde{\boldsymbol{x}}^*$ also satisfies (17) – (18).
Assume that $\tilde{\boldsymbol{x}}^*$ does not satisfy (17) – (18).

- First we assume that for some index k the vector $\tilde{\boldsymbol{x}}_{2k+1}$ does not satisfy (17). Let l be the smallest index such that

  $$\tilde{\boldsymbol{x}}_{2l+1} \geqslant \max\left(\tilde{\boldsymbol{x}}_{2l} + \boldsymbol{b}_1 \tilde{\delta}_{2l} + \boldsymbol{b}_3,\, \boldsymbol{b}_5\right) \text{ and } (\tilde{\boldsymbol{x}}_{2l+1})_i \neq \max\left((\tilde{\boldsymbol{x}}_{2l} + \boldsymbol{b}_1 \tilde{\delta}_{2l} + \boldsymbol{b}_3)_i,\, (\boldsymbol{b}_5)_i\right)$$

  for some index i.
  Now define $\hat{\boldsymbol{\delta}}^* = \tilde{\boldsymbol{\delta}}^*$ and

  $$\hat{\boldsymbol{x}}^* = \begin{bmatrix} \hat{\boldsymbol{x}}_1 \\ \hat{\boldsymbol{x}}_2 \\ \vdots \\ \hat{\boldsymbol{x}}_N \end{bmatrix}$$

  with

  $$\hat{\boldsymbol{x}}_k = \tilde{\boldsymbol{x}}_k \qquad \text{for } k = 0, 1, \ldots, 2l,$$
  $$\hat{\boldsymbol{x}}_{2k+1} = \max\left(\hat{\boldsymbol{x}}_{2k} + \boldsymbol{b}_1 \hat{\delta}_{2k} + \boldsymbol{b}_3,\, \boldsymbol{b}_5\right) \qquad \text{for } k = l, l+1, \ldots, \left\lfloor \frac{N-1}{2} \right\rfloor,$$
  $$\hat{\boldsymbol{x}}_{2k+2} = \max(\hat{\boldsymbol{x}}_{2k+1} + \boldsymbol{b}_2 \hat{\delta}_{2k+1} + \boldsymbol{b}_4,\, \boldsymbol{b}_6) \qquad \text{for } k = l, l+1, \ldots, \left\lfloor \frac{N}{2} \right\rfloor - 1 \; .$$

  Note that $\hat{\boldsymbol{x}}_{2l+1} \leqslant \tilde{\boldsymbol{x}}_{2l+1}$. Let us now show that $\hat{\boldsymbol{x}}_{2l+2} \leqslant \tilde{\boldsymbol{x}}_{2l+2}$. We have

  $$\hat{\boldsymbol{x}}_{2l+2} = \max\left(\hat{\boldsymbol{x}}_{2l+1} + \boldsymbol{b}_2 \hat{\delta}_{2l+1} + \boldsymbol{b}_4,\, \boldsymbol{b}_6\right)$$

$$
\begin{aligned}
&= \max(\hat{\boldsymbol{x}}_{2l+1} + \boldsymbol{b}_2 \tilde{\delta}_{2l+1} + \boldsymbol{b}_4, \boldsymbol{b}_6) && (\text{since } \hat{\boldsymbol{\delta}}^* = \tilde{\boldsymbol{\delta}}^*) \\
&\leqslant \max(\tilde{\boldsymbol{x}}_{2l+1} + \boldsymbol{b}_2 \tilde{\delta}_{2l+1} + \boldsymbol{b}_4, \boldsymbol{b}_6) && (\text{since } \hat{\boldsymbol{x}}_{2l+1} \leqslant \tilde{\boldsymbol{x}}_{2l+1}) \\
&\leqslant \tilde{\boldsymbol{x}}_{2l+2} && (\text{by (36) and (37)}).
\end{aligned}
$$

In a similar way we can prove that $\hat{\boldsymbol{x}}_k \leqslant \tilde{\boldsymbol{x}}_k$ for $k = 2l+3, 2l+4, \ldots, N$. Hence, $\hat{\boldsymbol{x}}^* \leqslant \tilde{\boldsymbol{x}}^*$. Since $\tilde{\boldsymbol{x}}_k \leqslant \boldsymbol{x}_{\max}$ for all k, this implies that $\hat{\boldsymbol{x}}_k$ also satisfies (33) for all k. Furthermore, since $\hat{\boldsymbol{\delta}}^* = \tilde{\boldsymbol{\delta}}^*$, this implies that $\hat{\boldsymbol{x}}_k$ and $\hat{\delta}_k$ satisfy the inequalities of the system (31) – (37) for all k.

- If $\tilde{\boldsymbol{x}}_{2k+1}$ satisfies (17) for all k, then there has to exist an index l such that (18) does not hold for $\tilde{\boldsymbol{x}}_{2l+2}$ since we have assumed that $\tilde{\boldsymbol{x}}^*$ does not satisfy (17) – (18). Now we can use a reasoning that is similar to the one given above to construct $\hat{\boldsymbol{x}}^*$ and $\hat{\boldsymbol{\delta}}^*$ such that $\hat{\boldsymbol{x}}^* \leqslant \tilde{\boldsymbol{x}}^*$, $\hat{x}_j^* < \tilde{x}_j^*$ for some j and such that $\hat{\boldsymbol{x}}_k$ and $\hat{\delta}_k$ satisfy the inequalities of the system (31) – (37) for all k.

Since $\hat{\boldsymbol{x}}^* \leqslant \tilde{\boldsymbol{x}}^*$ and $\hat{x}_j^* < \tilde{x}_j^*$ for at least one index j, we have $J(\hat{\boldsymbol{x}}^*, \hat{\boldsymbol{\delta}}^*) < J(\tilde{\boldsymbol{x}}^*, \tilde{\boldsymbol{\delta}}^*)$ which would imply that $\tilde{\boldsymbol{x}}^*, \tilde{\boldsymbol{\delta}}^*$ is not an optimal solution of problem $\tilde{\mathcal{P}}$. Since this is a contradiction, our initial assumption that $\tilde{\boldsymbol{x}}^*$ does not satisfy (17) – (18) is wrong. The solution set of (14) – (18) is a subset of the solution set of (31) – (37). This implies that $\tilde{\boldsymbol{x}}^*$, $\tilde{\boldsymbol{\delta}}^*$ also is an optimal solution of the original problem $\mathcal{P}$. So now we have proved that every optimal solution of the relaxed problem $\tilde{\mathcal{P}}$ is also an optimal solution of the original problem $\mathcal{P}$. □

Note that in general it is easier to solve the relaxed problem $\tilde{\mathcal{P}}$ since the set of feasible solutions of $\tilde{\mathcal{P}}$ is a convex set, whereas the set of feasible solutions of $\mathcal{P}$ is in general not convex since it consists of a union of faces of the polyhedron defined by the system of inequalities (31) – (37).

Let us now show that $\tilde{J}_1$ and $\tilde{J}_4$ are strictly monotonous functions of $\boldsymbol{x}^*$, i.e., that they satisfy the conditions of Proposition 3.2.

Proposition 3.3 *For given $\boldsymbol{x}_0$, δ_{amb}, $\bar{\lambda}_i$'s, $\bar{\mu}_i$'s, $\bar{\kappa}_i$'s and a given $\boldsymbol{\delta}^*$ the functions $\tilde{J}_1$ and $\tilde{J}_4$ are strictly monotonous functions of $\boldsymbol{x}^*$.*

Proof: Let $\boldsymbol{\delta}^* \in (\mathbb{R}_0^+)^N$ and let $\tilde{\boldsymbol{x}}^*, \hat{\boldsymbol{x}}^* \in (\mathbb{R}^+)^{4N}$. If $\tilde{\boldsymbol{x}}^* \leqslant \hat{\boldsymbol{x}}^*$ and $\tilde{x}_j^* < \hat{x}_j^*$ for at least one index $j = 4(k-1)+i$ then we have $\tilde{l}_i(t, \tilde{\boldsymbol{x}}^*, \boldsymbol{\delta}^*) \leqslant \tilde{l}_i(t, \hat{\boldsymbol{x}}^*, \boldsymbol{\delta}^*)$ for all $t \in [t_0, t_N]$, and $\tilde{l}_i(t, \tilde{\boldsymbol{x}}^*, \boldsymbol{\delta}^*) < \tilde{l}_i(t, \hat{\boldsymbol{x}}^*, \boldsymbol{\delta}^*)$ in some non-empty interval $(t_k - \eta, t_k + \eta) \cap [t_0, t_N]$ with $\eta > 0$. Hence,

$$
\int_{t_0}^{t_N} \tilde{l}_i(t, \tilde{\boldsymbol{x}}^*, \boldsymbol{\delta}^*)\, dt \;<\; \int_{t_0}^{t_N} \tilde{l}_i(t, \hat{\boldsymbol{x}}^*, \boldsymbol{\delta}^*)\, dt \ ,
$$

which implies that $\tilde{J}_1(\tilde{\boldsymbol{x}}^*, \boldsymbol{\delta}^*) < \tilde{J}_1(\hat{\boldsymbol{x}}^*, \boldsymbol{\delta}^*)$ and $\tilde{J}_4(\tilde{\boldsymbol{x}}^*, \boldsymbol{\delta}^*) < \tilde{J}_4(\hat{\boldsymbol{x}}^*, \boldsymbol{\delta}^*)$. □

Consider $\boldsymbol{\delta}^*$ and $\tilde{\boldsymbol{x}}^*, \hat{\boldsymbol{x}}^*$ with $\tilde{\boldsymbol{x}}^* \leqslant \hat{\boldsymbol{x}}^*$ and $\tilde{x}_j^* < \hat{x}_j^*$ for at least one index j. Note that although $\tilde{J}_l(\tilde{\boldsymbol{x}}^*, \boldsymbol{\delta}^*) \leqslant \tilde{J}_l(\hat{\boldsymbol{x}}^*, \boldsymbol{\delta}^*)$ for $l = 2, 3, 5$, in general we do not have $\tilde{J}_l(\tilde{\boldsymbol{x}}^*, \boldsymbol{\delta}^*) < \tilde{J}_l(\hat{\boldsymbol{x}}^*, \boldsymbol{\delta}^*)$ for $l = 2, 3, 5$. So $\tilde{J}_2$, $\tilde{J}_3$ and $\tilde{J}_5$ are not *strictly* monotonous functions of $\boldsymbol{x}^*$ for a given $\boldsymbol{\delta}^*$.

In general $\tilde{J}_1$ and $\tilde{J}_4$ are not convex functions of $\boldsymbol{x}^*$ and $\boldsymbol{\delta}^*$ [4]. However, computational experiments have shown that the objective functions $\tilde{J}_1$ and $\tilde{J}_4$ are smooth enough, so that selecting different starting points for the optimization routine almost always leads to more or less the same numerical result (cf. Example 5.1).

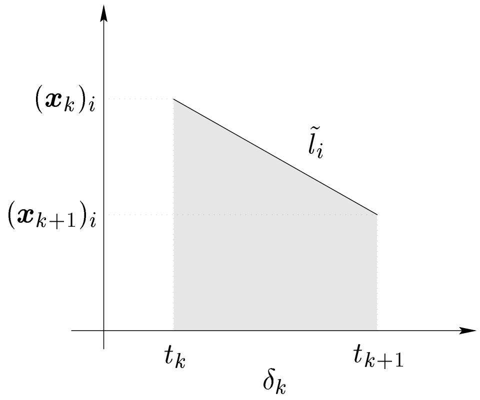

Figure 4: The function $\tilde{l}_i$ in the interval (t_k, t_{k+1}).

We shall now discuss a further approximation of $\tilde{J}_1$ (and $\tilde{J}_4$) that will lead to a problem that can be solved very efficiently. Let us first derive a formula to compute $\tilde{J}_1(\boldsymbol{x}^*, \boldsymbol{\delta}^*)$. The value of $\displaystyle\int_{t_k}^{t_{k+1}} \tilde{l}_i(t, \boldsymbol{x}^*, \boldsymbol{\delta}^*)\, dt$ is equal to the surface of the trapezium determined by the points $(t_k, 0)$, $(t_k, (\boldsymbol{x}_k)_i)$, $(t_{k+1}, (\boldsymbol{x}_{k+1})_i)$ and $(t_{k+1}, 0)$ in the t–$\tilde{l}_i$ plane (see Figure 4). Hence,

$$\int_{t_k}^{t_{k+1}} \tilde{l}_i(t, \boldsymbol{x}^*, \boldsymbol{\delta}^*) \;=\; \frac{\delta_k}{2}\big((\boldsymbol{x}_k)_i + (\boldsymbol{x}_{k+1})_i\big) \quad ,$$

and thus

$$\tilde{J}_1(\boldsymbol{x}^*, \boldsymbol{\delta}^*) \;=\; \sum_{i=1}^{4} w_i \, \frac{\displaystyle\sum_{k=0}^{N-1} \delta_k \big((\boldsymbol{x}_k)_i + (\boldsymbol{x}_{k+1})_i\big)}{2(\delta_0 + \delta_1 + \ldots + \delta_{N-1})} \quad . \tag{38}$$

If we assume that $\delta_k \approx \dfrac{t_N - t_0}{N}$ for all k then (38) leads to:

$$\tilde{J}_1(\boldsymbol{x}^*, \boldsymbol{\delta}^*) \;\approx\; \sum_{i=1}^{4} w_i \left(\frac{1}{2N}(\boldsymbol{x}_0)_i \;+\; \sum_{k=1}^{N-1} \frac{1}{N}(\boldsymbol{x}_k)_i \;+\; \frac{1}{2N}(\boldsymbol{x}_N)_i \right) \;\stackrel{\text{def}}{=}\; \hat{J}_1(\boldsymbol{x}^*) \quad .$$

Note that the right-hand side of this expression is a linear function of the components of $\boldsymbol{x}^*$. Since scaling of a linear objective function and subtracting a constant term $\left(\displaystyle\sum_{i=1}^{4} \frac{1}{2N}(\boldsymbol{x}_0)_i\right)$ does not change the location of the minimum, this implies that we can minimize $J_{\text{lin}} = \boldsymbol{\omega}^T \boldsymbol{x}^*$ with

$$\boldsymbol{\omega} = \begin{bmatrix} w_1 & w_2 & w_3 & w_4 & w_1 & \ldots & w_4 & \dfrac{w_1}{2} & \dfrac{w_2}{2} & \dfrac{w_3}{2} & \dfrac{w_4}{2} \end{bmatrix}^T$$

in order to find a minimum of $\hat{J}_1$. Since $w_i > 0$ for all i, J_{lin} is a strictly monotonous function of $\boldsymbol{x}^*$. As a consequence, Proposition 3.2 implies that for $J = J_{\text{lin}}$ problem (13) – (18) reduces to a linear programming problem, which can be solved efficiently using (variants of) the simplex method or using an interior point method (see, e.g., [15]). Note that the solution of this linear programming problem could be used as a good initial starting point for an iterative algorithm to compute a minimum of J_1 or $\tilde{J}_1$ using constrained optimization.

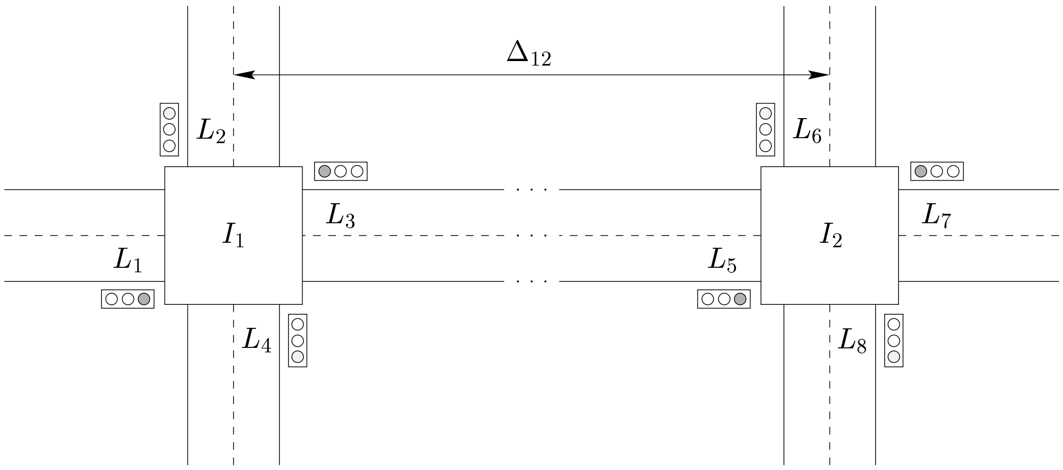

Figure 5: Two neighboring intersections I_1 and I_2. The distance between the two intersections is equal to Δ_{12}.

4 Extensions of the basic model

Let us now discuss some extensions of the basic model that has been introduced in Section 2.

In the previous sections we have assumed that δ_{amb} was a known constant. However, it is easy to verify that if δ_{amb} is a variable (or even if it depends on k) the model of the system can still be recast as an ELCP and all the techniques developed in Section 3 can still be applied.

The simplified model of Section 2 can easily be adapted to a situation in which some streets consist of more than 2 lanes. Furthermore, our model can also be adapted to a situation in which there is a difference in the departure rates for vehicles that turn left, right or go straight ahead (provided that the ratios between these three types of traffic are constant or slowly time-varying). This ratios can be measured using cameras or can be predicted using estimation techniques for intersection origin-destination matrices (see, e.g., [16, 18]).

If we want to extend our approach to intersections of networks, then the main difficulty for a centralized solution is the fact that the cycle time may vary from one cycle to another. One possible solution is to add an extra equation of the form $\delta_{2k} + \delta_{2k+1} = T$ where T is the fixed cycle time for the main road. To determine T we can use the conventional approaches. Note that in order to obtain synchronization the travel time between two neighboring intersections on the main road should be an integer multiple of T. It is easy to verify that once T has been determined, we still have an ELCP.

Note that if we use a moving horizon strategy in combination with a *decentralized control* solution, we can apply the approach given in this paper and use measurements from one intersection to better predict the arrival rates at the other intersections as follows. Consider a two neighboring intersections I_1 and I_2 on a main road (see Figure 5). Suppose that the distance between intersection I_1 and intersection I_2 is Δ_{12} and that the average speed on the trajectory from I_1 to I_2 is v_{12}. The travel time from I_1 to I_2 is then approximately $t_{12} = \dfrac{\Delta_{12}}{v_{12}}$. Now assume that t_{12} is larger than the average cycle time at I_2. Let r_{ij} be the ratio of vehicles that enter intersection I_1 at lane L_i and leave it via the lane next to lane L_j (see, e.g., [16, 18] for methods to estimate these ratios). If we measure the number of cars that leave intersection I_1 in the direction of I_2 at time t, or if we measure the arrival rates in lanes L_1, L_2 and L_4 at time t and if we have an accurate estimation of the ratios r_{13}, r_{23}, and r_{43} at time t, we can get a good estimate or prediction of the arrival rate of cars arriving at intersection I_2 from the direction of I_1 at time $t + t_{12}$. These predictions can then be used by a local controller to determine (sub)optimal traffic light switching schemes for I_2 using a moving horizon strategy. A similar approach can be used for the traffic arriving at I_1 and coming from I_2.

For more information on other models that describe the evolution of the queue lengths at a traffic-light-controlled intersection and on optimal traffic light control the interested reader is referred to [5, 7, 11, 12, 13, 14, 17, 19, 23] and the references given therein.

5 Example

With this example we want to illustrate that using the approximations introduced in the Section 3.2 for the objective functions J_1 (or equivalently J_4) leads to good suboptimal solutions that can be computed very efficiently. All times will be expressed in seconds and all rates in vehicles per second.

Example 5.1 Consider the intersection of Figure 1 with the following data: $\bar{\lambda}_1 = 0.25$, $\bar{\lambda}_2 = 0.12$, $\bar{\lambda}_3 = 0.20$, $\bar{\lambda}_4 = 0.1$, $\bar{\mu}_1 = \bar{\mu}_3 = 0.5$, $\bar{\mu}_2 = \bar{\mu}_4 = 0.4$, $\bar{\kappa}_1 = \bar{\kappa}_3 = 0.05$, $\bar{\kappa}_2 = \bar{\kappa}_4 = 0.03$, $\boldsymbol{x}_0 = [\,20\ \ 19\ \ 14\ \ 12\,]^T$, $\delta_{\text{amb}} = 3$, $\delta_{\text{min,green,1}} = \delta_{\text{min,green,2}} = 6$, $\delta_{\text{max,green,1}} = \delta_{\text{max,green,2}} = 60$, $\boldsymbol{x}_{\max} = [\,25\ \ 20\ \ 25\ \ 20\,]^T$. Let $\boldsymbol{w} = [\,2\ \ 1\ \ 2\ \ 1\,]^T$. Suppose that we want to compute a traffic light switching sequence $t_0, t_1, \ldots, t_7$ that minimizes J_1 — the weighted average queue length over all queues[10].
First we compute the solution set of the ELCP defined by (26) – (29) using the ELCP algorithm of [2, 3]. This yields a set V consisting of 818 vertices and a set Λ consisting of 21 index sets. Optimizing J_1 over the solution set of the ELCP yields the following optimal switching interval vector[11]:

$$\boldsymbol{\delta}^*_{\text{ELCP}} = [\,20.000\ \ 45.750\ \ \ 30.964\ \ \ 63.000\ \ \ 30.964\ \ \ 63.000\ \ \ 58.980\,]^T$$

(see Table 2 for the values of the objective functions and Figure 6 for the evolution of the queue lengths).
Next we use the `e04ucf` routine[12] of the NAG library [21] to compute a solution $\boldsymbol{\delta}^*_{\text{pen}}$ that minimizes the objective function

$$F(\boldsymbol{\delta}) = J_1(\boldsymbol{\delta}) + \alpha \min\left(5000N \sum_{(i,k)\in\Gamma} \big((\boldsymbol{x}_k(\boldsymbol{\delta}))_i - (\boldsymbol{x}_{\max})_i\big)\,,\ \sum_{(i,k)\in\Gamma} e^{\beta\left((\boldsymbol{x}_k(\boldsymbol{\delta}))_i - (\boldsymbol{x}_{\max})_i\right)} \right)$$

[10]We limit ourselves to a traffic light switching sequence with $N = 7$, since for higher values of N the ELCP approach is not feasible any longer since it takes too much CPU time. Nevertheless, in this example we also use the ELCP approach since we want to compare the approximate suboptimal solutions with the exact optimal solution, which is obtained using the ELCP approach.

[11]In this example we give all numerical results up to 3 decimal places.

[12]This function uses a Sequential Quadratic Programming method to find a local minimum of a constrained multivariable function. We have listed the best solution over 20 runs with random initial points. The mean of the objective values of the local minima returned by the minimization routine was 65.905 with a standard deviation of 3.437.

$\boldsymbol{\delta}^*, \boldsymbol{x}^*$	$J_1(\boldsymbol{\delta}^*)$	$\tilde{J}_1(\boldsymbol{x}^*, \boldsymbol{\delta}^*)$	$\hat{J}_1(\boldsymbol{x}^*, \boldsymbol{\delta}^*)$	$J_{\text{lin}}(\boldsymbol{x}^*)$	CPU time
$\boldsymbol{\delta}^*_{\text{ELCP}}, \boldsymbol{x}^*_{\text{ELCP}}$	60.657	64.267	69.190	434.827	404.83
$\boldsymbol{\delta}^*_{\text{pen}}, \boldsymbol{x}^*_{\text{pen}}$	61.150	64.740	69.916	439.909	78.69
$\boldsymbol{\delta}^*_{\text{mul}}, \boldsymbol{x}^*_{\text{mul}}$	61.613	65.118	67.881	425.664	13.83
$\tilde{\boldsymbol{\delta}}^*, \tilde{\boldsymbol{x}}^*$	60.659	64.264	69.117	434.319	2.49
$\boldsymbol{\delta}^*_{\text{lin}}, \boldsymbol{x}^*_{\text{lin}}$	64.551	67.905	67.199	420.895	0.94
$\boldsymbol{\delta}^*_{\text{con}}, \boldsymbol{x}^*_{\text{con}}$	63.101	66.363	67.565	423.455	96.40

Table 2: The values of the objective functions J_1, $\tilde{J}_1$, $\hat{J}_1$ and J_{lin} (up to 3 decimal places) and the CPU time (up to 2 decimal places) needed to compute respectively an optimal solution of the ELCP problem ($\boldsymbol{\delta}^*_{\text{ELCP}}$), the original switching problem $\mathcal{P}$ using a penalty function ($\boldsymbol{\delta}^*_{\text{pen}}$), the suboptimal multi-ELCP problem ($\boldsymbol{\delta}^*_{\text{mul}}$), the relaxed problem $\tilde{\mathcal{P}}$ ($\tilde{\boldsymbol{\delta}}^*$), the linear programming problem ($\boldsymbol{\delta}^*_{\text{lin}}$) and the constant cycle length problem ($\boldsymbol{\delta}^*_{\text{con}}$) of Example 5.1. The queue length vectors $\boldsymbol{x}^*$ are compatible with the switching interval vectors $\boldsymbol{\delta}^*$ for $\boldsymbol{x}_0$.

with

$$\alpha = 10, \;\; \beta = 5 \;\text{ and }\; \Gamma = \{(i,k) \,|\, (\boldsymbol{x}_k(\boldsymbol{\delta}))_i > (\boldsymbol{x}_{\max})_i\},$$

where $\{\boldsymbol{x}_k(\boldsymbol{\delta})\}_{k=1}^N$ is the sequence of 4-component queue length vectors that is compatible with $\boldsymbol{\delta}$. The second term in the function F is a penalty term that is used to force the $\boldsymbol{x}_k$'s to satisfy the condition $\boldsymbol{x}_k \leqslant \boldsymbol{x}_{\max}$. This leads to

$$\boldsymbol{\delta}^*_{\text{pen}} = [\, 19.388 \quad 44.323 \quad 31.029 \quad 63.000 \quad 36.044 \quad 63.000 \quad 57.835 \,]^T \;\; .$$

Now we compute a suboptimal multi-ELCP solution $\boldsymbol{\delta}^*_{\text{mul}}$ by solving a number of smaller ELCPs with $N_{\text{s}} = 3$ and each time (except for the last) selecting the first switching interval. This results in

$$\boldsymbol{\delta}^*_{\text{mul}} = [\, 20.000 \quad 45.750 \quad 30.964 \quad 63.000 \quad 30.964 \quad 63.000 \quad 29.421 \,]^T \;\; .$$

Next we use the `e04ucf` routine to compute a solution $\tilde{\boldsymbol{x}}^*, \tilde{\boldsymbol{\delta}}^*$ that minimizes the approximate objective function $\tilde{J}_1$ (using the relaxed problem $\tilde{\mathcal{P}}$). This yields[13]:

$$\tilde{\boldsymbol{\delta}}^* = [\, 20.000 \quad 45.750 \quad 30.964 \quad 63.000 \quad 30.964 \quad 63.000 \quad 57.342 \,]^T \;\; .$$

Then we use the `e04mbf` function[14] of the NAG library to compute a solution $\boldsymbol{x}^*_{\text{lin}}, \boldsymbol{\delta}^*_{\text{lin}}$ to minimize the linear objective function $J_{\text{lin}}(\boldsymbol{x}^*) = [\, 2 \;\; 1 \;\; 2 \;\; 1 \;\; \ldots \;\; 1 \;\; 1 \;\; \frac{1}{2} \;\; 1 \;\; \frac{1}{2} \,]\, \boldsymbol{x}^*$ using the relaxed problem $\tilde{\mathcal{P}}$. We obtain:

$$\boldsymbol{\delta}^*_{\text{lin}} = [\, 20.000 \quad 45.750 \quad 40.350 \quad 63.000 \quad 21.579 \quad 63.000 \quad 9.000 \,]^T \;\; .$$

[13]In this case using different starting points always leads to more or less the same numerical value of the optimal objective function: in an experiment with 20 random starting points the first 10 decimal places of the final objective function always had the same value. Therefore, we have only performed one run with an arbitrary random initial point here.

[14]This function solves linear programming problems.

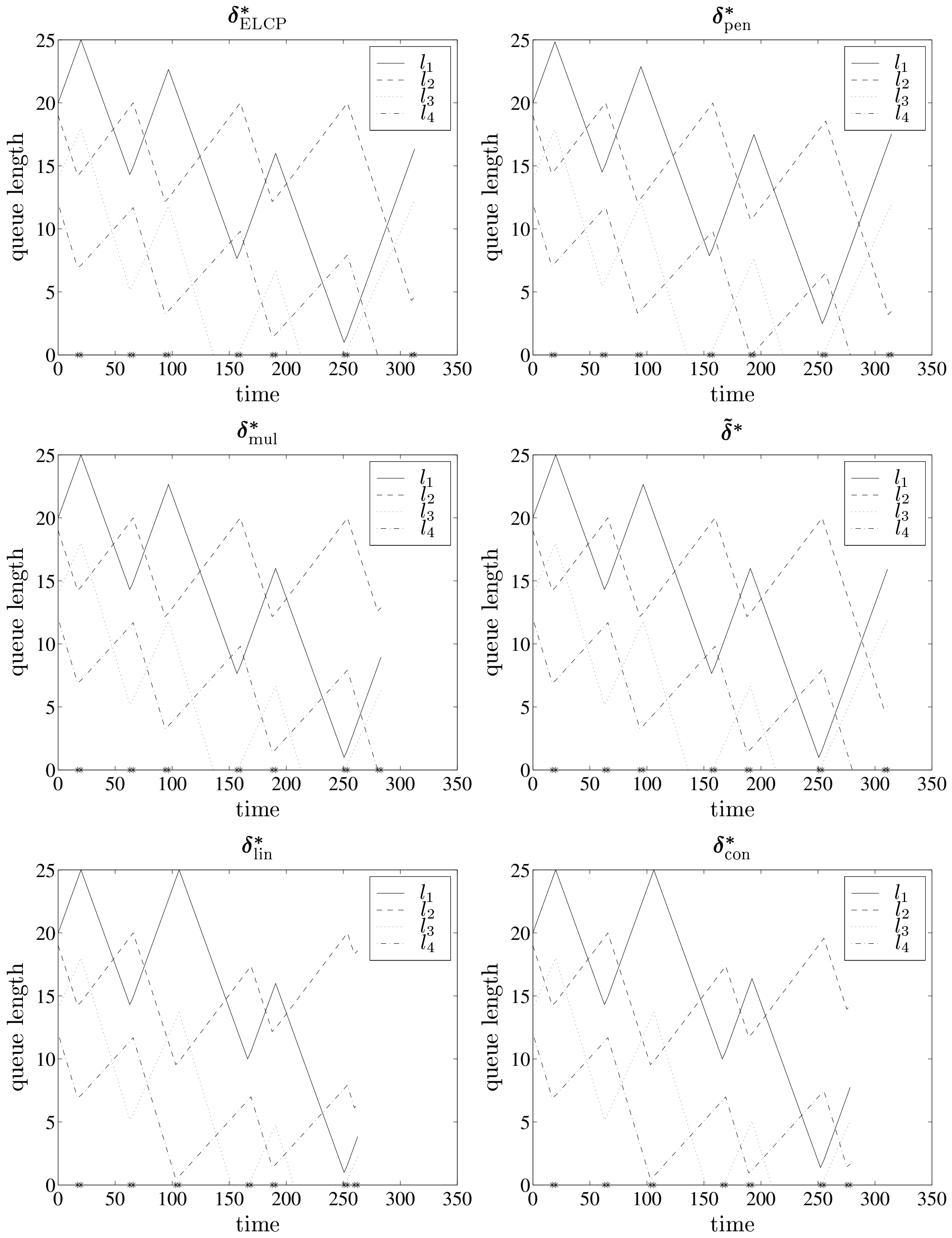

Figure 6: The queue lengths in the various lanes as a function of time for the traffic light switching sequences that correspond to the switching interval vectors of Example 5.1. The * signs on the time axis correspond to the switching time instants.

In order to compare our approach with a fixed cycle length approach we compute an optimal switching interval vector $\boldsymbol{\delta}^*_{\mathrm{con}}$ that also satisfies the following extra constraints: $\delta_1 + \delta_2 = \delta_3 + \delta_4 = \delta_5 + \delta_6$ (i.e., the length of the green-amber-red cycle for the main lanes L_1 and L_3 is constant). Using the ELCP approach[15] we obtain:

$$\boldsymbol{\delta}^*_{\mathrm{con}} = [\, 20.000 \quad 45.750 \quad 40.350 \quad 63.000 \quad 21.300 \quad 63.000 \quad 21.300 \,]^T \enspace .$$

Many conventional methods for determining optimal traffic light switching schemes use a two level approach: first the cycle length is determined and then the lengths of the green and red phases in each cycle are determined. Note that we can use a one level approach to determine $\boldsymbol{\delta}^*_{\mathrm{con}}$ since we optimize both the cycle length and the phase lengths in one step.
The evolution of the queue lengths for the various control strategies is represented in Figure 6. In Table 2 we have listed the values of the various objective functions for the various switching interval vectors and the CPU time[16] needed to compute the switching interval vectors. Note that in this example the ELCP solution is only given as a reference since the CPU time needed to compute the optimal switching interval vector using the ELCP algorithm of [2, 3] increases exponentially as N increases, which implies that the ELCP approach should never be used in practice, but one of the approximate objective functions should be used instead. If we look at Table 2 then we see that the $\tilde{\boldsymbol{\delta}}^*$ solution outperforms all other solutions except for the ELCP solution. Although this will in general not always be the case, we have noticed in our experiments that the objective value for this solution will always be close to the optimal value. Moreover, if we take the trade-off between optimality and efficiency into account, then the $\tilde{\boldsymbol{\delta}}^*$ solution is clearly the most interesting.

Let us now use the optimal switching interval vector $\boldsymbol{\delta}^*_{\mathrm{ELCP}}$ in an integer queue length simulation. In Figure 7 we have plotted the results of this simulation. The effective average queue length over all lanes $J_{1,\mathrm{sim}}$ for this simulation is 61.770[17]. □

6 Conclusions and further research

We have derived a model that describes the evolution of the queue lengths at an intersection of two two-way streets with controllable traffic lights on each corner. We have shown how an optimal traffic light switching scheme for the given system can be determined. In general this leads to a minimization problem over the solution set of an Extended Linear Complementarity Problem. We have shown that for objective functions that depend strictly monotonously on the queue lengths at the traffic light switching time instants, the optimal traffic light switching scheme can be computed very efficiently. We have derived some approximate objective functions for which this property holds. Moreover, if the objective function is linear, the problem reduces to a linear programming problem.

[15] It easy to verify that in this case we can also use the approximations introduced in Section 3.2 to obtain suboptimal solutions.

[16] CPU time on a SUN Ultra 2 Creator 2200 workstation with 128 M RAM and with the optimization routines called from MATLAB. The ELCP algorithm of [2, 3] has been implemented in C and for the NAG optimization routines (implemented in Fortran) the NAG Foundation Toolbox [20] of MATLAB has been used. The CPU time values listed in this table are average values over 10 experiments. For $\boldsymbol{\delta}^*_{\mathrm{pen}}$ we have given the total CPU time for 20 runs, each with a different random starting point.

[17] For 20 different simulation experiments, all with the switching interval vector $\boldsymbol{\delta}^*_{\mathrm{ELCP}}$, the mean value of $J_{1,\mathrm{sim}}$ was 61.178 with a standard deviation of 2.209.

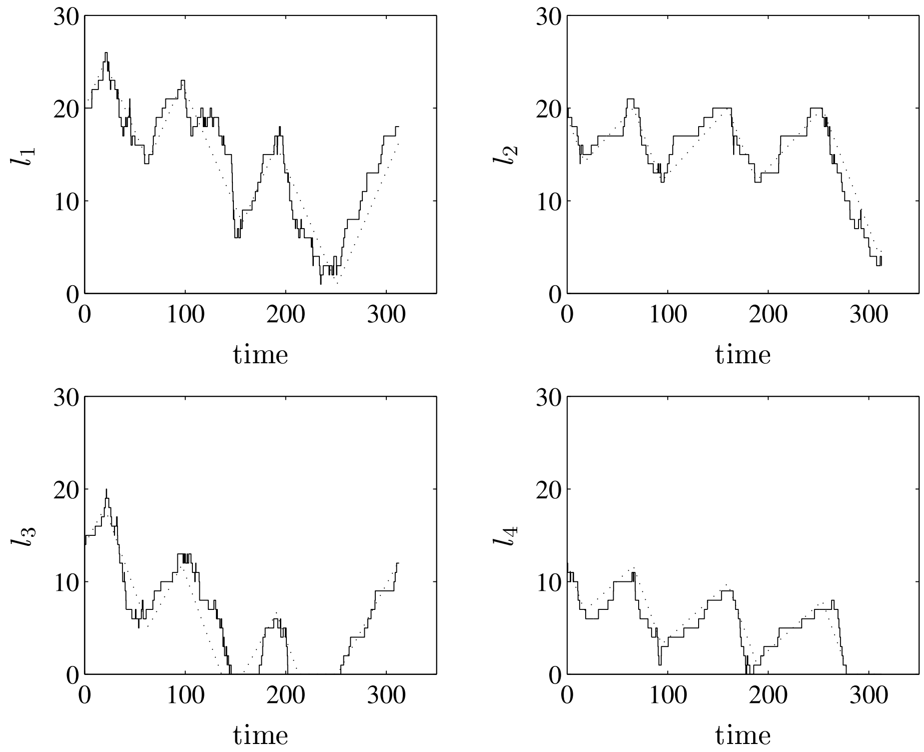

Figure 7: The queue lengths in the various lanes as a function of time for an integer queue length simulation for the traffic light switching sequence that corresponds to the switching interval vector $\boldsymbol{\delta}^*_{\mathrm{ELCP}}$. The integer queue length functions are plotted in full lines and their continuous approximations in dotted lines.

Topics for further research include: extension to models with integer queue lengths, further extension or modification of our approach to switching schemes with a fixed cycle time (such that it might also be used for networks of intersections); development of efficient algorithms to compute optimal traffic light switching schemes for the objective functions discussed in this paper; for other objective functions (such as intersection output or number of stops), and for more general (combined) objective functions; and development and design of neural and fuzzy traffic light controllers for single intersections and for networks of intersections.

Acknowledgments

We would like to thank Prof. J.L. Farges of ONERA/CERT (DERA, France) for his interesting and useful comments and suggestions on the first version of this paper.
This research was sponsored by the Concerted Action Project of the Flemish Community, entitled "Model-based Information Processing Systems" (GOA-MIPS), by the Belgian program on interuniversity attraction poles (IUAP P4-02 and IUAP P4-24), by the Belgian sustainable mobility program "Traffic Congestion Problems in Belgium: Mathematical Models, Simulation, Control and Actions" (MD/01/24), by the ALAPEDES project of the European Community Training and Mobility of Researchers Program, and by the European Commission Human Capital and Mobility Network SIMONET ("System Identification and Modelling Network").

References

[1] D.W. Clarke, C. Mohtadi, and P.S. Tuffs, "Generalized predictive control – Part I. The basic algorithm," *Automatica*, vol. 23, no. 2, pp. 137–148, Mar. 1987.

[2] B. De Schutter, *Max-Algebraic System Theory for Discrete Event Systems*. PhD thesis, Faculty of Applied Sciences, K.U.Leuven, Leuven, Belgium, Feb. 1996.

[3] B. De Schutter and B. De Moor, "The extended linear complementarity problem," *Mathematical Programming*, vol. 71, no. 3, pp. 289–325, Dec. 1995.

[4] B. De Schutter and B. De Moor, "Optimal traffic light control for a single intersection: Addendum," Tech. rep. 96-90a, ESAT-SISTA, K.U.Leuven, Leuven, Belgium, Jan. 1998.

[5] J. Favilla, A. Machion, and F. Gomide, "Fuzzy traffic control: Adaptive strategies," in *Proceedings of the 2nd IEEE International Conference on Fuzzy Systems*, San Francisco, California, pp. 506–511, Mar.–Apr. 1993.

[6] C.E. García, D.M. Prett, and M. Morari, "Model predictive control: Theory and practice — A survey," *Automatica*, vol. 25, no. 3, pp. 335–348, May 1989.

[7] N.H. Gartner, J.D.C. Little, and H. Gabbay, "Simultaneous optimization of offsets, splits, and cycle time," *Transportation Research Record*, no. 596, pp. 6–15, 1976.

[8] J.K. Hedrick, M. Tomizuka, and P. Varaiya, "Control issues in automated highway systems," *IEEE Control Systems Magazine*, vol. 14, no. 6, pp. 21–32, Dec. 1994.

[9] H.J.C. Huijberts and J.H. van Schuppen, "Routing control of a motorway network: A summary," in *Proceedings of the 3rd European Control Conference*, Rome, Italy, pp. 781–784, Sept. 1995.

[10] D.K. Kahaner, "Singapore's road traffic control and management," ATIP Report 96.060, Asian Technology Information Program, Tokyo, Japan, July 1996.

[11] H.R. Kashani and G.N. Saridis, "Intelligent control for urban traffic systems," *Automatica*, vol. 19, no. 2, pp. 191–197, Mar. 1983.

[12] J.H. Lim, S.H. Hwang, I.H. Suh, and Z. Bien, "Hierarchical optimal control of oversaturated urban traffic networks," *International Journal of Control*, vol. 33, no. 4, pp. 727–737, Apr. 1981.

[13] D. Lin, F. Ulrich, L.L. Kinney, and K.S.P. Kumar, "Hierarchical techniques in traffic control," in *Control in Transportation Systems, Proceedings of the IFAC/IFIP/IFORS 3rd International Symposium*, Columbus, Ohio, pp. 163–171, 1976.

[14] A.D. May, *Traffic Flow Fundamentals*. Englewood Cliffs, New Jersey: Prentice-Hall, 1990.

[15] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*. Philadelphia, Pennsylvania: SIAM, 1994.

[16] N.L. Nihan and G.A. Davis, “Application of prediction-error minimization and maximum likelihood to estimate intersection O-D matrices from traffic counts,” *Transportation Science*, vol. 23, no. 2, pp. 77–90, May 1989.

[17] E.S. Park, J.H. Lim, I.H. Suh, and Z. Bien, “Hierarchical optimal control of urban traffic networks,” *International Journal of Control*, vol. 40, no. 4, pp. 813–829, Oct. 1984.

[18] H.D. Sherali, N. Arora, and A.G. Hobeika, “Parameter optimization methods for estimating dynamic origin-destination trip-tables,” *Transportation Research Part B*, vol. 31, no. 2, pp. 141–157, Apr. 1997.

[19] M.G. Singh and H. Tamura, “Modelling and hierarchical optimization for oversaturated urban road traffic networks,” *International Journal of Control*, vol. 20, no. 6, pp. 913–934, 1974.

[20] The Mathworks and The Numerical Algorithms Group, Natick, Massachusetts, *NAG Foundation Toolbox User's Guide*, Apr. 1995.

[21] The Numerical Algorithms Group, Oxford, UK, *NAG Fortran Library Manual, Mark 16*, Sept. 1993.

[22] P. Varaiya, “Smart cars on smart roads: Problems of control,” *IEEE Transactions on Automatic Control*, vol. 38, no. 2, pp. 195–207, Feb. 1993.

[23] M. Wirth, “Traffic light control using branch & bound,” in *Proceedings of the International Workshop on Discrete Event Systems (WODES'96)*, Edinburgh, UK, pp. 244–249, Aug. 1996.

Optimal traffic light control for a single intersection: Addendum

Bart De Schutter and Bart De Moor

In this addendum we give some extra propositions, proofs and examples in connection with the model for the evolution of the queue lengths at the switching time instants that we have derived in Section 2, and in connection with the design of optimal traffic light switching schemes for this model.

In Section A we derive necessary and sufficient conditions for stability of this system under a periodic traffic light switching policy. In Section B we consider an oversaturated intersection, i.e., an intersection for which the queue lengths never become equal to 0 (except possibly at the end of a green or an amber phase). In Section C we derive expressions that give the value of the objective functions J_1, J_2, J_3, J_4, J_5 as a function of the switching time instants and the queue lengths at the switching time instants. In Section D we discuss the convexity and the concavity of the objective functions J_1, J_2, J_3, J_4 and J_5. Finally, in Section E we discuss another approximation of J_1 and J_4 that is also strictly monotonous as a function of the queue length vector.

A Stability

Now we discuss the conditions under which the system is "stable", i.e., has queue lengths that remain bounded as k goes to infinity. We assume that after a finite number, say $2K$, of switching cycles the switching scheme reaches a periodic regime, i.e., $\delta_{2k} = \delta_{\mathrm{e}}$ and $\delta_{2k+1} = \delta_{\mathrm{o}}$ for all $k \geqslant K$.

If we consider lane L_1, then there arrive $\bar{\lambda}_1(\delta_{\mathrm{e}} + \delta_{\mathrm{o}})$ vehicles during one complete green-amber-red cycle and (at most) $\bar{\mu}_1(\delta_{\mathrm{o}} - \delta_{\mathrm{amb}}) + \bar{\kappa}_1\delta_{\mathrm{amb}}$ vehicles leave lane L_1. So in order to prevent an unlimited growth of the queue length the following condition should hold:

$$\bar{\mu}_1(\delta_{\mathrm{o}} - \delta_{\mathrm{amb}}) + \bar{\kappa}_1\delta_{\mathrm{amb}} \geqslant \bar{\lambda}_1(\delta_{\mathrm{e}} + \delta_{\mathrm{o}}) \ .$$

If we write down similar conditions for the other lanes, we obtain the following necessary and sufficient conditions for stability:

$$(\bar{\mu}_1 - \bar{\lambda}_1)\delta_{\mathrm{o}} - \bar{\lambda}_1\delta_{\mathrm{e}} \geqslant (\bar{\mu}_1 - \bar{\kappa}_1)\delta_{\mathrm{amb}} \tag{39}$$

$$-\bar{\lambda}_2\delta_{\mathrm{o}} + (\bar{\mu}_2 - \bar{\lambda}_2)\delta_{\mathrm{e}} \geqslant (\bar{\mu}_2 - \bar{\kappa}_2)\delta_{\mathrm{amb}} \tag{40}$$

$$(\bar{\mu}_3 - \bar{\lambda}_3)\delta_{\mathrm{o}} - \bar{\lambda}_3\delta_{\mathrm{e}} \geqslant (\bar{\mu}_3 - \bar{\kappa}_3)\delta_{\mathrm{amb}} \tag{41}$$

$$-\bar{\lambda}_4\delta_{\mathrm{o}} + (\bar{\mu}_4 - \bar{\lambda}_4)\delta_{\mathrm{e}} \geqslant (\bar{\mu}_4 - \bar{\kappa}_4)\delta_{\mathrm{amb}} \ . \tag{42}$$

Together with the conditions $\delta_{\mathrm{min,green},1} \leqslant \delta_{\mathrm{o}} - \delta_{\mathrm{amb}} \leqslant \delta_{\mathrm{max,green},1}$ and $\delta_{\mathrm{min,green},2} \leqslant \delta_{\mathrm{e}} - \delta_{\mathrm{amb}} \leqslant \delta_{\mathrm{max,green},2}$, the conditions (39) – (42) define a convex region in the δ_{o}–δ_{e} plane. Note that adding conditions of this form to the conditions (14) - (18) of the optimal traffic light control problem still leads to an ELCP.

B Non-saturated versus oversaturated intersections

If we consider an oversaturated network then the queue lengths never become 0 during the green cycle (except possibly at the end of a green or an amber phase). In that case the maximum operator in (4) and (5), and thus also in (6) – (7) is not needed any more, and then we get the following model:

$$\boldsymbol{x}_{2k+1} = \boldsymbol{x}_{2k} + \boldsymbol{b}_1\delta_{2k} + \boldsymbol{b}_3 \tag{43}$$

$$\boldsymbol{x}_{2k+2} = \boldsymbol{x}_{2k+1} + \boldsymbol{b}_2\delta_{2k+1} + \boldsymbol{b}_4 \tag{44}$$

for $k = 0, 1, 2, \ldots$ with the extra constraints

$$\boldsymbol{x}_{2k+1} \geqslant \boldsymbol{b}_5$$
$$\boldsymbol{x}_{2k+2} \geqslant \boldsymbol{b}_6$$

for $k = 0, 1, 2, \ldots$ to ensure that the queue lengths are nonnegative at the end of the green and the amber phase. It is easy to verify that (43) and (44) lead to

$$\boldsymbol{x}_{2k+1} = \boldsymbol{x}_0 + \sum_{j=0}^{k} \boldsymbol{b}_1\delta_{2j} + \sum_{j=0}^{k-1} \boldsymbol{b}_2\delta_{2j+1} + (k+1)\boldsymbol{b}_3 + k\boldsymbol{b}_4$$
$$\boldsymbol{x}_{2k+2} = \boldsymbol{x}_0 + \sum_{j=0}^{k} \boldsymbol{b}_1\delta_{2j} + \sum_{j=0}^{k} \boldsymbol{b}_2\delta_{2j+1} + (k+1)\boldsymbol{b}_3 + (k+1)\boldsymbol{b}_4 \ .$$

As a consequence, the optimal traffic light control problem now becomes

$$\text{minimize } J$$

subject to

$$\delta_{\text{min,green,1}} \leqslant \delta_{2k+1} - \delta_{\text{amb}} \leqslant \delta_{\text{max,green,1}} \qquad \text{for } k = 0, 1, \ldots, \left\lfloor \frac{N}{2} \right\rfloor - 1,$$

$$\delta_{\text{min,green,2}} \leqslant \delta_{2k} - \delta_{\text{amb}} \leqslant \delta_{\text{max,green,2}} \qquad \text{for } k = 0, 1, \ldots, \left\lfloor \frac{N-1}{2} \right\rfloor,$$

$$\boldsymbol{b}_5 \leqslant \boldsymbol{x}_0 + \sum_{j=0}^{k} \boldsymbol{b}_1\delta_{2j} + \sum_{j=0}^{k-1} \boldsymbol{b}_2\delta_{2j+1} + (k+1)\boldsymbol{b}_3 + k\boldsymbol{b}_4 \leqslant \boldsymbol{x}_{\text{max}} \qquad \text{for } k = 0, 1, \ldots, \left\lfloor \frac{N-1}{2} \right\rfloor,$$

$$\boldsymbol{b}_6 \leqslant \boldsymbol{x}_0 + \sum_{j=0}^{k} \boldsymbol{b}_1\delta_{2j} + \sum_{j=0}^{k} \boldsymbol{b}_2\delta_{2j+1} + (k+1)\boldsymbol{b}_3 + (k+1)\boldsymbol{b}_4 \leqslant \boldsymbol{x}_{\text{max}} \qquad \text{for } k = 0, 1, \ldots, \left\lfloor \frac{N}{2} \right\rfloor - 1.$$

$\tilde{\boldsymbol{\delta}}^*, \tilde{\boldsymbol{x}}^*$	$J_1(\tilde{\boldsymbol{\delta}}^*)$	$\tilde{J}_1(\tilde{\boldsymbol{x}}^*, \tilde{\boldsymbol{\delta}}^*)$
$\boldsymbol{\delta}^*_{\mathrm{ELCP}}, \boldsymbol{x}^*_{\mathrm{ELCP}}$	60.657	64.267
$\tilde{\boldsymbol{\delta}}^*, \tilde{\boldsymbol{x}}^*$	60.659	64.264
$\boldsymbol{\delta}^*_{\mathrm{lin}}, \boldsymbol{x}^*_{\mathrm{lin}}$	64.551	67.905
$\boldsymbol{\delta}^*_{\mathrm{os}}, \boldsymbol{x}^*_{\mathrm{os}}$	72.658	74.452

Table 3: The values of the objective functions J_1 and $\tilde{J}_1$ (up to 3 decimal places) for the traffic light switching sequences defined by the switching interval vectors $\boldsymbol{\delta}^*_{\mathrm{ELCP}}$, $\tilde{\boldsymbol{\delta}}^*$, $\boldsymbol{\delta}^*_{\mathrm{lin}}$ of Example 5.1 and the switching interval vector $\boldsymbol{\delta}^*_{\mathrm{os}}$ of Example B.1. The queue length vectors $\boldsymbol{x}^*$ are compatible with the switching interval vectors $\boldsymbol{\delta}^*$ for $\boldsymbol{x}_0$.

The feasible region of this optimization problem is convex, which implies that it can be solved more efficiently than the optimization problem (13) – (18). However, the following example shows that in general applying the oversaturated model (43) – (44) to non-saturated intersections does not lead to an optimal solution. Therefore we could say that the use of the maximum operator will in the first instance lead to complex optimization problems — but for which some approximations lead to good suboptimal solutions that can be computed very efficiently as has been shown in Section 3.2 — whereas omitting the maximum operator initially leads to simpler models but finally results in control schemes that for non-saturated intersections have an inferior performance.

Example B.1 Consider the intersection of Figure 1 with the same data as in Example 5.1: $\bar{\lambda}_1 = 0.25$, $\bar{\lambda}_2 = 0.12$, $\bar{\lambda}_3 = 0.20$, $\bar{\lambda}_4 = 0.1$, $\bar{\mu}_1 = \bar{\mu}_3 = 0.5$, $\bar{\mu}_2 = \bar{\mu}_4 = 0.4$, $\bar{\kappa}_1 = \bar{\kappa}_3 = 0.05$, $\bar{\kappa}_2 = \bar{\kappa}_4 = 0.03$, $\boldsymbol{x}_0 = [\,20 \;\; 19 \;\; 14 \;\; 12\,]^T$, $\delta_{\mathrm{amb}} = 3$, $\delta_{\mathrm{min,green,1}} = \delta_{\mathrm{min,green,2}} = 6$, $\delta_{\mathrm{max,green,1}} = \delta_{\mathrm{max,green,2}} = 60$, $\boldsymbol{x}_{\mathrm{max}} = [\,25 \;\; 20 \;\; 25 \;\; 20\,]^T$ and $\boldsymbol{w} = [\,2 \;\; 1 \;\; 2 \;\; 1\,]^T$. Suppose that we want to compute a traffic light switching sequence $t_0, t_1, \ldots, t_7$ that minimizes J_1.
In Example 5.1 we have computed several suboptimal switching interval vectors based on the model (6) – (7). Let us now compute a minimum $\boldsymbol{\delta}^*_{\mathrm{os}}$ of the objective function J_1 based on the model (43) – (44) for oversaturated intersections using the `e04ucf` routine of the NAG library. This results in[18]

$$\boldsymbol{\delta}^*_{\mathrm{os}} = [\,20.000 \quad 45.750 \quad 18.600 \quad 34.150 \quad 38.433 \quad 30.122 \quad 13.741\,]^T \ .$$

In Table 3 we have listed the values of the objective functions for the switching interval vectors $\boldsymbol{\delta}^*_{\mathrm{ELCP}}$ (the optimal solution of the original problem using the ELCP approach), $\tilde{\boldsymbol{\delta}}^*$ (the optimal solution of the relaxed problem), $\boldsymbol{\delta}^*_{\mathrm{lin}}$ (the optimal solution of the linear programming problem) and $\boldsymbol{\delta}^*_{\mathrm{os}}$. The evolution of the queue lengths for the various control strategies is represented in Figures 6 and 8.
Clearly, the suboptimal solutions based on the model (6) – (7) correspond to much lower values of the objective function J_1 than the optimal solution based on the model (43) – (44) for oversaturated intersections. □

[18] We have listed the best solution over 20 runs with random initial points. The mean of the objective values of the local minima returned by the minimization routine was 73.717 with a standard deviation of 1.336.

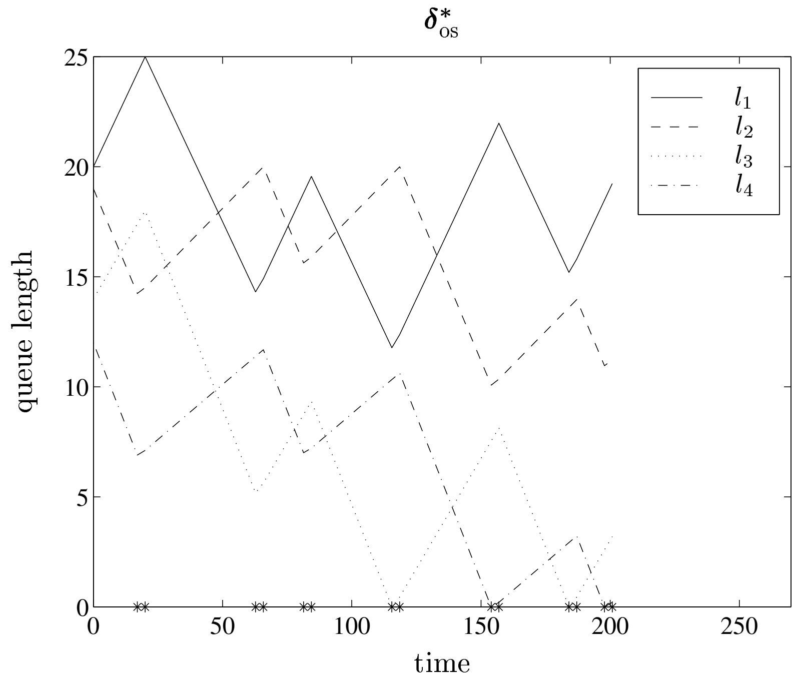

Figure 8: The queue lengths in the various lanes as a function of time for the traffic light switching sequence that corresponds to the switching interval vector $\boldsymbol{\delta}^*_{\text{os}}$ of Example B.1. The $*$ signs on the time axis correspond to the switching time instants.

C Evaluation of the objective functions

Let $\boldsymbol{x}_0 \in (\mathbb{R}^+)^4$, $\boldsymbol{\delta}^* \in (\mathbb{R}_0^+)^{4N}$ and let $\{\delta_k\}_{k=0}^{N-1}$ be the sequence of switching time intervals that corresponds to $\boldsymbol{\delta}^*$. First we derive a formula that expresses $J_3(\boldsymbol{\delta}^*)$ as a function of the sequence of queue length vectors $\{\boldsymbol{x}_k\}_{k=0}^{N}$ that is compatible with $\{\delta_k\}_{k=0}^{N-1}$ for a given $\boldsymbol{x}_0$.
We have assumed that $\bar{\kappa}_i \leqslant \bar{\mu}_i$ for all i. Recall that this implies that a situation such as in the left plot of Figure 3 where $\bar{\lambda}_i - \bar{\mu}_i > 0$ and $\bar{\lambda} - i - \bar{\kappa}_i < 0$ is not possible. So if $k \in \mathcal{G}_i$, then the maximum value of l_i in the interval (t_k, t_{k+1}) where T_i is first green and then amber will be reached in t_k or in t_{k+1}. Furthermore, in an interval (t_{k+1}, t_{k+2}) where T_i is red the maximum value of l_i will be reached in t_{k+1} or in t_{k+2}. Since $l_i(\cdot, \boldsymbol{\delta}^*)$ is a piecewise-linear function, this implies that l_i reaches its maximum over the interval $[t_0, t_N]$ in one of the switching time instants t_k. As a consequence, we have

$$J_3(\boldsymbol{\delta}^*) \;=\; \max_{i,\,k} \big(w_i(\boldsymbol{x}_k)_i\big) \enspace .$$

Now we derive a formula for the evaluation of

$$\int_{t_0}^{t_N} l_1(t)\,dt = \sum_{k=0}^{N-1} \int_{t_k}^{t_{k+1}} l_1(t)\,dt \enspace .$$

Define $y_k = (\boldsymbol{x}_k)_1 = l_1(t_k)$ for $k = 0, 1, \ldots, N$ and $\tilde{y}_{2k+2} = l_1(t_{2k+2} - \delta_{\text{amb}})$ for $k = 0, 1, \ldots, \left\lfloor \dfrac{N}{2} \right\rfloor - 1$. Note that

$$\tilde{y}_{2k+2} = \max(y_{2k+1} + (\bar{\lambda}_1 - \bar{\mu}_1)(\delta_{2k+1} - \delta_{\text{amb}}), 0) \enspace .$$

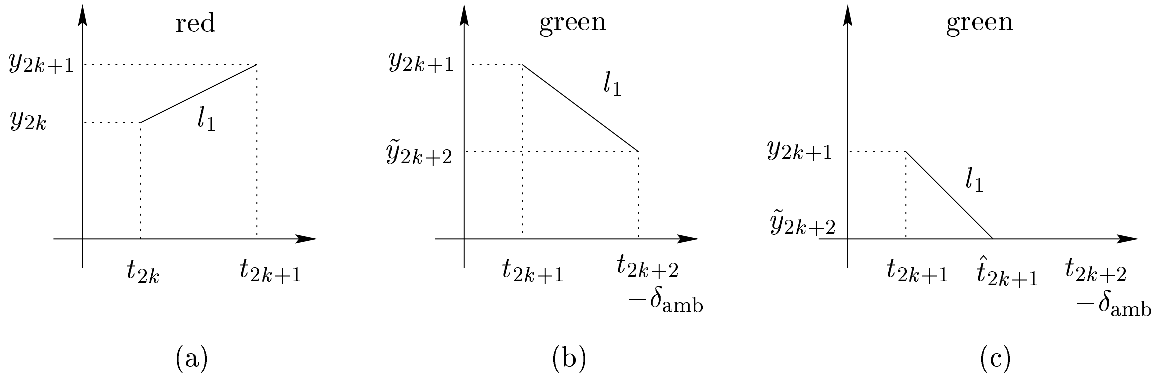

Figure 9: Three possible basic cases for the evolution of the queue length l_1 as a function of time t in an interval (t_{2k}, t_{2k+1}) where the traffic light is red, or in an interval $(t_{2k+1}, t_{2k+2} - \delta_{\text{amb}})$ where the traffic light is green.

If we want to evaluate $\int_{t_\text{b}}^{t_\text{e}} l_1(t)\,dt$ where t_b and t_e are respectively the beginning and the end of a red or a green phase, there are three possible basic cases (see Figure 9).

- In an interval of the form (t_{2k}, t_{2k+1}) the light is red, which means that l_1 is a nondecreasing function. In this case $\int_{t_{2k}}^{t_{2k+1}} l_1(t)\,dt$ is equal to the surface of the trapezium defined by the points $(t_{2k}, 0)$, (t_{2k}, y_{2k}), (t_{2k+1}, y_{2k+1}) and $(t_{2k+1}, 0)$. Hence,

$$\int_{t_{2k}}^{t_{2k+1}} l_1(t)\,dt \;=\; \frac{y_{2k} + y_{2k+1}}{2}\,\delta_{2k} \quad . \tag{45}$$

- In an interval of the form $(t_{2k+1}, t_{2k+2} - \delta_{\text{amb}})$ the light is green. If the queue length l_1 is identically 0 in $(t_{2k+1}, t_{2k+2} - \delta_{\text{amb}})$ or if l_1 never becomes 0 in $(t_{2k+1}, t_{2k+2} - \delta_{\text{amb}})$, $\int_{t_{2k+1}}^{t_{2k+2}-\delta_{\text{amb}}} l_1(t)\,dt$ is equal the surface of the trapezium defined by the points $(t_{2k+1}, 0)$, (t_{2k+1}, y_{2k+1}), $(t_{2k+2} - \delta_{\text{amb}}, \tilde{y}_{2k+2})$ and $(t_{2k+2} - \delta_{\text{amb}}, 0)$. So,

$$\int_{t_{2k+1}}^{t_{2k+2}-\delta_{\text{amb}}} l_1(t)\,dt \;=\; \frac{y_{2k+1} + \tilde{y}_{2k+2}}{2}\,(\delta_{2k+1} - \delta_{\text{amb}}) \qquad \text{if } \tilde{y}_{2k+2} \neq 0\,. \tag{46}$$

- If the queue length l_1 becomes 0 in the interval $(t_{2k+1}, t_{2k+2} - \delta_{\text{amb}})$, $\int_{t_{2k+1}}^{t_{2k+2}-\delta_{\text{amb}}} l_1(t)\,dt$ is equal to the surface of the triangle defined by the points $(t_{2k+1}, 0)$, (t_{2k+1}, y_{2k+1}), $(\hat{t}_{2k+1}, 0)$ where $\hat{t}_{2k+1}$ is the smallest value of $t \in (t_{2k+1}, t_{2k+2} - \delta_{\text{amb}})$ for which $l_1(t) = 0$ (see Figure 9(c)). Since in this case the absolute value of the slope of $l_1(t)$ is equal to $\bar{\mu}_1 - \bar{\lambda}_1$ in $(t_{2k+1}, t_{2k+2} - \delta_{\text{amb}})$, we have $y_{2k+1} = (\bar{\mu}_1 - \bar{\lambda}_1)(\hat{t}_{2k+1} - t_{2k+1})$ and thus $\hat{t}_{2k+1} - t_{2k+1} = \dfrac{y_{2k+1}}{\bar{\mu}_1 - \bar{\lambda}_1}$. As consequence, we have

$$\int_{t_{2k+1}}^{t_{2k+2}-\delta_{\text{amb}}} l_1(t)\,dt \;=\; \frac{y_{2k+1}^2}{2(\bar{\mu}_1 - \bar{\lambda}_1)} \qquad \text{if } \tilde{y}_{2k+2} = 0\,. \tag{47}$$

Before we consider the amber phase, we shall show that the expressions (46) and (47) for the green phase also cover expression (45) for the red phase if the appropriate changes of variables are made, i.e., we shall prove that

$$\int_{t_{2k}}^{t_{2k+1}} l_1(t)\,dt \;=\; \begin{cases} \dfrac{y_{2k}+y_{2k+1}}{2}\,\delta_{2k} & \text{if } y_{2k+1}\neq 0\,, \\ \dfrac{y_{2k+1}^2}{2\bar{\lambda}_1} & \text{if } y_{2k+1}=0\,. \end{cases} \tag{48}$$

If the queue length y_{2k+1} at the end of the red phase is different from 0, then (45) corresponds to the first case of (48). On the other hand, if y_{2k+1} is equal to 0, then y_{2k} will also be zero since l_1 is nondecreasing in (t_{2k}, t_{2k+1}). So then both (45) and the second case of (48) yield 0.

When the traffic light is amber (i.e., in an interval of the form $(t_{2k+1}-\delta_{\text{amb}}, t_{2k+2})$), we either have the situation of case (a) if the queue length is nondecreasing in the given interval (i.e., if $\bar{\lambda}_i - \bar{\kappa}_i \geqslant 0$), or case (b) or (c) if the queue length is decreasing in the given interval (i.e., if $\bar{\lambda}_i - \bar{\kappa}_i < 0$). Since expressions (46) and (47) also cover expression (45) if the appropriate changes of variables are made, this implies that

$$\int_{t_{2k+2}-\delta_{\text{amb}}}^{t_{2k+2}} l_1(t)\,dt \;=\; \begin{cases} \dfrac{\tilde{y}_{2k+2}+y_{2k+2}}{2}\,\delta_{\text{amb}} & \text{if } y_{2k+2}\neq 0\,, \\ \dfrac{\tilde{y}_{2k+2}^2}{2(\bar{\kappa}_1-\bar{\lambda}_1)} & \text{if } y_{2k+2}=0\,. \end{cases}$$

So

$$\begin{aligned} \int_{t_0}^{t_N} l_1(t)\,dt \;=\; & \sum_{k=0}^{\lfloor \frac{N-1}{2}\rfloor} \frac{y_{2k}+y_{2k+1}}{2}\,\delta_{2k} \;+\; \sum_{\substack{k=0\\ \tilde{y}_{2k+2}\neq 0}}^{\lfloor \frac{N}{2}\rfloor -1} \frac{y_{2k+1}+\tilde{y}_{2k+2}}{2}\,(\delta_{2k+1}-\delta_{\text{amb}}) \;+ \\ & \sum_{\substack{k=0\\ \tilde{y}_{2k+2}= 0}}^{\lfloor \frac{N}{2}\rfloor -1} \frac{y_{2k+1}^2}{2(\bar{\mu}_1-\bar{\lambda}_1)} \;+\; \sum_{\substack{k=0\\ y_{2k+2}\neq 0}}^{\lfloor \frac{N}{2}\rfloor -1} \frac{\tilde{y}_{2k+2}+y_{2k+2}}{2}\,\delta_{\text{amb}} \;+ \\ & \sum_{\substack{k=0\\ y_{2k+2}= 0}}^{\lfloor \frac{N}{2}\rfloor -1} \frac{\tilde{y}_{2k+2}^2}{2(\bar{\kappa}_1-\bar{\lambda}_1)}\;. \end{aligned}$$

We can write down similar expressions for $\displaystyle\int_{t_0}^{t_N} l_i(t)\,dt$ for $i=2,3,4$.

D Convexity or concavity of the objective functions

First we show that J_3 is convex as a function of $\boldsymbol{\delta}^*$, which implies that problem (13) – (18) with $J=J_3$ can be solved efficiently (if there is no upper bound on the queue lengths or if we introduce a convex penalty term if one or more components of $\boldsymbol{x}_{\max}$ are finite).

Proposition D.1 *For given $\boldsymbol{x}_0$, δ_{amb}, $\bar{\lambda}_i$'s, $\bar{\mu}_i$'s and $\bar{\kappa}_i$'s the function J_3 is convex as a function of $\boldsymbol{\delta}^*$.*

Proof: In Section C we have already shown that

$$J_3(\boldsymbol{\delta}^*) \;=\; \max_{i,\,k} \big(w_i (\boldsymbol{x}_k)_i \big) \tag{49}$$

if $\boldsymbol{x}^*$ and $\boldsymbol{\delta}^*$ are compatible for a given $\boldsymbol{x}_0$.
Let $\boldsymbol{x}_0 \in (\mathbb{R}^+)^4$ and consider $\boldsymbol{\delta}^*, \boldsymbol{\eta}^* \in (\mathbb{R}_0^+)^N$. Let $\{\delta_k\}_{k=0}^{N-1}$ and $\{\eta_k\}_{k=0}^{N-1}$ be the sequences of switching time intervals that correspond to $\boldsymbol{\delta}^*$ and $\boldsymbol{\eta}^*$ respectively. Define $\boldsymbol{y}_0 = \boldsymbol{z}_0 = \boldsymbol{x}_0$. Let the sequences $\{\boldsymbol{x}_k\}_{k=0}^{N}$ and $\{\boldsymbol{y}_k\}_{k=0}^{N}$ be compatible with $\{\delta_k\}_{k=0}^{N-1}$ and $\{\eta_k\}_{k=0}^{N-1}$ respectively. Consider an arbitrary number $u \in [0,1]$ and let the sequence $\{\boldsymbol{z}_k\}_{k=0}^{N}$ be compatible with the sequence $\{u\delta_k + (1-u)\eta_k\}_{k=0}^{N-1}$.
Note that the sequences $\{\boldsymbol{x}_k\}_{k=0}^{N}$, $\{\boldsymbol{y}_k\}_{k=0}^{N}$ and $\{\boldsymbol{z}_k\}_{k=0}^{N}$ all satisfy recurrence equations of the form (6) – (7).
Let us now show by induction that

$$\boldsymbol{z}_k \leqslant u\boldsymbol{x}_k + (1-u)\boldsymbol{y}_k \qquad \text{for } k = 0, 1, \ldots, N \ . \tag{50}$$

We have $\boldsymbol{z}_0 = u\boldsymbol{x}_0 + (1-u)\boldsymbol{y}_0 = u\boldsymbol{x}_0 + (1-u)\boldsymbol{x}_0 = \boldsymbol{x}_0$.
Now we assume that $\boldsymbol{z}_k \leqslant u\boldsymbol{x}_k + (1-u)\boldsymbol{y}_k$ for $k = 0, 1, \ldots, K$ with $K < N$ and we show that $\boldsymbol{z}_{K+1} \leqslant u\boldsymbol{x}_{K+1} + (1-u)\boldsymbol{y}_{K+1}$.
Suppose that K is even. Hence, $K = 2l$ for some integer l. Now we have

$$\begin{aligned}
& u\boldsymbol{x}_{2l+1} + (1-u)\boldsymbol{y}_{2l+1} \\
&\quad = u \max\left(\boldsymbol{x}_{2l} + \boldsymbol{b}_1\delta_{2l} + \boldsymbol{b}_3,\, \boldsymbol{b}_5\right) + \\
&\qquad\quad (1-u) \max\left(\boldsymbol{y}_{2l} + \boldsymbol{b}_1\eta_{2l} + \boldsymbol{b}_3,\, \boldsymbol{b}_5\right) && \text{(by (6))} \\
&\quad = \max\big(u(\boldsymbol{x}_{2l} + \boldsymbol{b}_1\delta_{2l} + \boldsymbol{b}_3) + (1-u)(\boldsymbol{y}_{2l} + \boldsymbol{b}_1\eta_{2l} + \boldsymbol{b}_3), \\
&\qquad\quad u(\boldsymbol{x}_{2l} + \boldsymbol{b}_1\delta_{2l} + \boldsymbol{b}_3) + (1-u)\boldsymbol{b}_5, \\
&\qquad\quad u\boldsymbol{b}_5 + (1-u)(\boldsymbol{y}_{2l} + \boldsymbol{b}_1\eta_{2l} + \boldsymbol{b}_3), u\boldsymbol{b}_5 + (1-u)\boldsymbol{b}_5\big) \\
&\quad \geqslant \max\big(u(\boldsymbol{x}_{2l} + \boldsymbol{b}_1\delta_{2l} + \boldsymbol{b}_3) + (1-u)(\boldsymbol{y}_{2l} + \boldsymbol{b}_1\eta_{2l} + \boldsymbol{b}_3), \boldsymbol{b}_5\big) \\
&\quad \geqslant \max\Big(u\boldsymbol{x}_{2l} + (1-u)\boldsymbol{y}_{2l} + \\
&\qquad\quad \boldsymbol{b}_1\big(u\delta_{2l} + (1-u)\eta_{2l}\big) + \boldsymbol{b}_3, \boldsymbol{b}_5\Big) \\
&\quad \geqslant \max\Big(\boldsymbol{z}_{2l} + \boldsymbol{b}_1\big(u\delta_{2l} + (1-u)\eta_{2l}\big) + \boldsymbol{b}_3, \boldsymbol{b}_5\Big) && \text{(by the induction hypothesis)} \\
&\quad \geqslant \boldsymbol{z}_{2l+1} \ .
\end{aligned}$$

If $K = 2l+1$ is odd, then we can show in a similar way that $u\boldsymbol{x}_{2l+2} + (1-u)\boldsymbol{y}_{2l+2} \geqslant \boldsymbol{z}_{2l+2}$.
As a consequence, we have

$$\begin{aligned}
J_3(u\boldsymbol{\delta}^* + (1-u)\boldsymbol{\eta}^*) &= \max_{i,\,k}\big(w_i(\boldsymbol{z}_k)_i\big) && \text{(by (49))} \\
&\leqslant \max_{i,\,k}\big(w_i(u\boldsymbol{x}_k + (1-u)\boldsymbol{y}_k)_i\big) && \text{(by (50))} \\
&\leqslant u \max_{i,\,k}\big(w_i(\boldsymbol{x}_k)_i\big) + (1-u)\max_{i,\,k}\big(w_i(\boldsymbol{y}_k)_i\big) \\
&\leqslant u J_3(\boldsymbol{\delta}^*) + (1-u) J_3(\boldsymbol{\eta}^*) && \text{(by (49))}\ ,
\end{aligned}$$

which implies that J_3 is convex as a function of $\boldsymbol{\delta}^*$. □

Proposition D.2 *Let the set of feasible solutions of the ELCP that corresponds to a given optimal traffic light switching problem be characterized by the set of vertices* $V = \left\{ \begin{bmatrix} \boldsymbol{x}_i^* \\ \boldsymbol{\delta}_i^* \end{bmatrix} \middle| \, i = 1, 2, \ldots, r \right\}$ *and the set of index sets* Λ. *Let* $\phi_j = \{i_1, i_2, \ldots, i_s\} \in \Lambda$. *Then the function* $J_3\left(\sum_{j=1}^{s} \nu_j \boldsymbol{\delta}_{i_j}^* \right)$ *with* $\nu_j \geqslant 0$ *for all* j *and* $\sum_{j=1}^{s} \nu_j = 1$ *is a convex function of the* ν_j*'s.*

Proof: Note that we may assume without loss of generality that $\phi_j = \{1, 2, \ldots, s\}$. Let $\boldsymbol{\nu} = \begin{bmatrix} \nu_1 & \nu_2 & \ldots & \nu_s \end{bmatrix}^T$ with $\nu_j \geqslant 0$ for all j and $\sum_{j=1}^{s} \nu_j = 1$. Let $\boldsymbol{\delta}^*(\boldsymbol{\nu}) = \sum_{j=1}^{s} \nu_j \boldsymbol{\delta}_j^*$ and $\boldsymbol{x}^*(\boldsymbol{\nu}) = \sum_{j=1}^{s} \nu_j \boldsymbol{x}_j^*$. Let $\{\boldsymbol{x}_{j,k}\}_{k=1}^{N}$ be the sequence of 4-component queue length vectors that corresponds to the $4N$-component queue length vector $\boldsymbol{x}_j^*$ for $j = 1, 2, \ldots, s$.

Define $I_3(\boldsymbol{\nu}) = J_3(\boldsymbol{\delta}^*(\boldsymbol{\nu})) = J_3\left(\sum_{j=1}^{s} \nu_j \boldsymbol{\delta}_j^* \right)$. Now we prove that I_3 is a convex function.

Since $\begin{bmatrix} \boldsymbol{x}^*(\boldsymbol{\nu}) \\ \boldsymbol{\delta}^*(\boldsymbol{\nu}) \end{bmatrix}$ is a convex combination of $\begin{bmatrix} \boldsymbol{x}_1^* \\ \boldsymbol{\delta}_1^* \end{bmatrix}, \begin{bmatrix} \boldsymbol{x}_2^* \\ \boldsymbol{\delta}_2^* \end{bmatrix}, \ldots, \begin{bmatrix} \boldsymbol{x}_s^* \\ \boldsymbol{\delta}_s^* \end{bmatrix}$, it is also a solution of the ELCP that corresponds to the given optimal traffic light switching problem. As a consequence, $\boldsymbol{x}^*(\boldsymbol{\nu})$ and $\boldsymbol{\delta}^*(\boldsymbol{\nu})$ are compatible for the given $\boldsymbol{x}_0$. This implies that

$$I_3(\boldsymbol{\nu}) = \max_{i,k} \left(w_i \left(\sum_{j=1}^{s} \nu_j (\boldsymbol{x}_{j,k})_i \right), \; w_i (\boldsymbol{x}_0)_i \right) \qquad \text{(by (49))}.$$

Let $\boldsymbol{\eta} = \begin{bmatrix} \eta_1 & \eta_2 & \ldots & \eta_s \end{bmatrix}^T$ with $\eta_j \geqslant 0$ for all j and $\sum_{j=1}^{s} \eta_j = 1$. Let $u \in [0, 1]$. Note that $u\nu_j + (1-u)\eta_j \geqslant 0$ for all j and that $\sum_{j=1}^{s} u\nu_j + (1-u)\eta_j = u + (1-u) = 1$. As a consequence, we have

$$\begin{aligned}
I_3(u\boldsymbol{\nu} + (1-u)\boldsymbol{\eta}) &= \max_{i,k} \left(w_i \left(\sum_{j=1}^{s} (u\nu_j + (1-u)\eta_j)(\boldsymbol{x}_{j,k})_i \right), \; w_i(\boldsymbol{x}_0)_i \right) \\
&= \max_{i,k} \Bigg(u \left(w_i \sum_{j=1}^{s} \nu_j (\boldsymbol{x}_{j,k})_i \right) + (1-u) \left(w_i \sum_{j=1}^{s} \eta_j (\boldsymbol{x}_{j,k})_i \right), \\
&\qquad\qquad u w_i (\boldsymbol{x}_0)_i + (1-u) w_i (\boldsymbol{x}_0)_i \Bigg) \\
&\leqslant u \max_{i,k} \left(w_i \sum_{j=1}^{s} \nu_j (\boldsymbol{x}_{j,k})_i, \; w_i (\boldsymbol{x}_0)_i \right) +
\end{aligned}$$

$$
(1-u)\max_{i,k}\left(w_i\sum_{j=1}^{s}\eta_j(\boldsymbol{x}_{j,k})_i,\ w_i(\boldsymbol{x}_0)_i\right)
$$

$$
\leqslant uI_3(\boldsymbol{\nu})+(1-u)I_3(\boldsymbol{\eta})\ .
$$

So I_3 is convex. Hence, J_3 is a convex function of the ν_i's. $\square$

Note that the objective functions J_1, J_2, J_4 and J_5 do not depend directly on $\boldsymbol{x}^*$ since for given $\bar{\lambda}_i$'s, $\bar{\mu}_i$'s, $\bar{\kappa}_i$'s, $\boldsymbol{x}_0$ and δ_{amb} the switching interval vector $\boldsymbol{\delta}^*$ uniquely determines $\boldsymbol{x}^*$. The following example shows that J_1, J_2, J_4 and J_5 are in general neither convex nor concave as a function of $\boldsymbol{\delta}^*$. Recall that we use the notation $l_i(\cdot,\boldsymbol{\delta}^*)$ to indicate that the queue length function $l_i(\cdot)$ corresponds to the switching interval vector $\boldsymbol{\delta}^*$.

Example D.3 Let $\delta_{\text{amb}}=3$ and $\bar{\lambda}_i=0.25$, $\bar{\mu}_i=0.5$, $\bar{\kappa}_i=0$ for $i,1,2,3,4$. Let

$$
\boldsymbol{x}_0=\begin{bmatrix}2\\0\\2\\0\end{bmatrix},\ \boldsymbol{w}=\begin{bmatrix}1\\1\\1\\1\end{bmatrix},\ \boldsymbol{\delta}_1^*=\begin{bmatrix}10\\10\end{bmatrix},\ \boldsymbol{\delta}_2^*=\begin{bmatrix}10\\30\end{bmatrix}\ \text{and}\ \boldsymbol{\delta}_3^*=\begin{bmatrix}10\\20\end{bmatrix}\ .
$$

In Figure 10 we have plotted the evolution of l_1 as a function of time for the switching sequences defined by $\boldsymbol{\delta}_1^*$, $\boldsymbol{\delta}_2^*$, $\boldsymbol{\delta}_3^*=\dfrac{\boldsymbol{\delta}_1^*+\boldsymbol{\delta}_2^*}{2}$ and $\boldsymbol{\delta}_4^*=\dfrac{\boldsymbol{\delta}_1^*+\boldsymbol{\delta}_3^*}{2}$.
Define

$$
f(\boldsymbol{\delta}^*)=\frac{\displaystyle\int_{t_0}^{t_N} l_1(t,\boldsymbol{\delta}^*)\,dt}{t_N-t_0}\ .
$$

We have $f(\boldsymbol{\delta}_1^*)\approx 3.363$, $f(\boldsymbol{\delta}_2^*)\approx 1.853$,

$$
f\Big(\frac{\boldsymbol{\delta}_1^*+\boldsymbol{\delta}_2^*}{2}\Big)\approx 2.492\quad\text{and}\quad\frac{f(\boldsymbol{\delta}_1^*)+f(\boldsymbol{\delta}_2^*)}{2}\approx 2.608\ .
$$

So

$$
f\Big(\frac{\boldsymbol{\delta}_1^*+\boldsymbol{\delta}_2^*}{2}\Big)<\frac{f(\boldsymbol{\delta}_1^*)+f(\boldsymbol{\delta}_2^*)}{2}\ ,
$$

which implies that f is not concave.
On the other hand, we have $f(\boldsymbol{\delta}_1^*)\approx 3.363$, $f(\boldsymbol{\delta}_3^*)\approx 2.492$,

$$
f\Big(\frac{\boldsymbol{\delta}_1^*+\boldsymbol{\delta}_3^*}{2}\Big)=2.965\quad\text{and}\quad\frac{f(\boldsymbol{\delta}_1^*)+f(\boldsymbol{\delta}_3^*)}{2}\approx 2.927\ .
$$

So

$$
f\Big(\frac{\boldsymbol{\delta}_1^*+\boldsymbol{\delta}_3^*}{2}\Big)>\frac{f(\boldsymbol{\delta}_1^*)+f(\boldsymbol{\delta}_3^*)}{2}\ ,
$$

which implies that f is not convex.

As a consequence, the objective functions J_1, J_2, J_4 and J_5 are in general neither convex nor concave.

Indeed, we have

$$
J_l\Big(\frac{\boldsymbol{\delta}_1^*+\boldsymbol{\delta}_2^*}{2}\Big)<\frac{J_l(\boldsymbol{\delta}_1^*)+J_l(\boldsymbol{\delta}_2^*)}{2}
$$

and

$$
J_l\Big(\frac{\boldsymbol{\delta}_1^*+\boldsymbol{\delta}_3^*}{2}\Big)>\frac{J_l(\boldsymbol{\delta}_1^*)+J_l(\boldsymbol{\delta}_3^*)}{2}
$$

for $l=1,2,4,5$ (see Table 4). $\square$

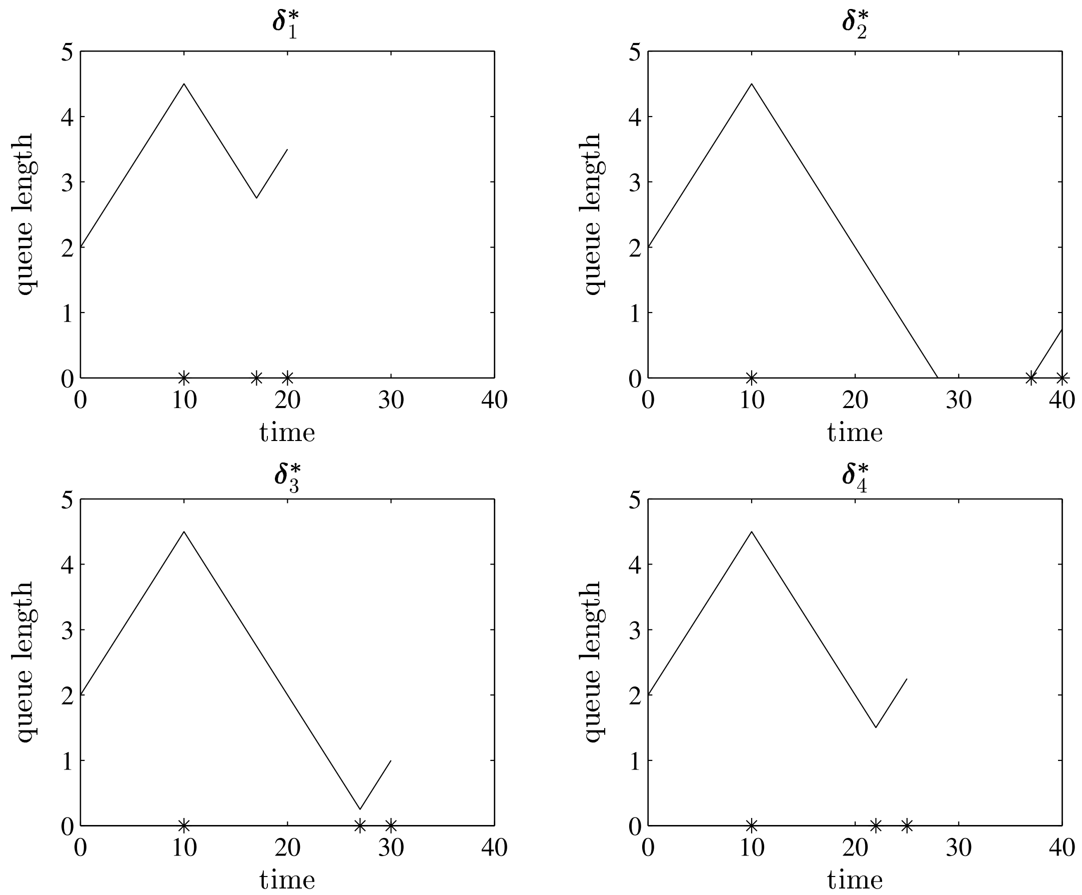

Figure 10: The evolution of the queue length in lane L_1 as a function of time for the switching interval vectors $\boldsymbol{\delta}_1^*$, $\boldsymbol{\delta}_2^*$, $\boldsymbol{\delta}_3^* = \dfrac{\boldsymbol{\delta}_1^* + \boldsymbol{\delta}_2^*}{2}$ and $\boldsymbol{\delta}_4^* = \dfrac{\boldsymbol{\delta}_1^* + \boldsymbol{\delta}_3^*}{2}$ of Example D.3. The $*$ signs on the time axis correspond to the switching time instants.

Let us now look at the convexity or concavity of the objective functions over the faces that constitute the solution set of the ELCP defined by (26) – (29).

In the next proposition we shall make the following extra assumption:

- in each lane, the average departure rate when the light is amber is less than the average arrival rate of vehicles, i.e., $\bar{\kappa}_i < \bar{\lambda}_i$.

Note that this is again a reasonable assumption if we take into account that designing optimal traffic light switching schemes is only useful if the arrival rates $\bar{\lambda}_i$ are high and that under normal circumstances $\bar{\kappa}_i$ is very small. This assumption implies that the net queue growth rate during the amber phase $\bar{\lambda}_i - \bar{\kappa}_i$ is positive. Therefore, the queue length at the end of a green phase is given by: $l_i(t_{k+1} - \delta_{\mathrm{amb}}) = l_i(t_{k+1}) - (\bar{\lambda}_i - \bar{\kappa}_i)\delta_{\mathrm{amb}}$ where $k \in \mathcal{G}_i(N)$.

Proposition D.4 *Consider an optimal traffic light switching problem with $\bar{\kappa}_i < \bar{\lambda}_i$ for all i. Let the set of feasible solutions of the ELCP that corresponds to the given problem be characterized by the set of vertices $V = \left\{ \begin{bmatrix} \boldsymbol{x}_i^* \\ \boldsymbol{\delta}_i^* \end{bmatrix} \,\middle|\, i = 1, 2, \ldots, r \right\}$ and the set of index sets Λ.*

l	$J_l(\boldsymbol{\delta}_1^*)$	$J_l(\boldsymbol{\delta}_2^*)$	$J_l\Big(\dfrac{\boldsymbol{\delta}_1^*+\boldsymbol{\delta}_2^*}{2}\Big)$	$\dfrac{J_l(\boldsymbol{\delta}_1^*)+J_l(\boldsymbol{\delta}_2^*)}{2}$
1	8.838	10.513	9.392	9.675
2	3.363	3.403	2.492	3.383
4	35.350	42.050	37.567	38.700
5	13.450	13.613	9.967	13.531

l	$J_l(\boldsymbol{\delta}_1^*)$	$J_l(\boldsymbol{\delta}_3^*)$	$J_l\Big(\dfrac{\boldsymbol{\delta}_1^*+\boldsymbol{\delta}_3^*}{2}\Big)$	$\dfrac{J_l(\boldsymbol{\delta}_1^*)+J_l(\boldsymbol{\delta}_3^*)}{2}$
1	8.838	9.392	9.170	9.115
2	3.363	2.492	2.965	2.927
4	35.350	37.567	36.680	36.458
5	13.450	9.967	11.860	11.708

Table 4: The values of the objective functions J_1, J_2, J_4 and J_5 (up to 3 decimal places) for the switching time vectors $\boldsymbol{\delta}_1^*$, $\boldsymbol{\delta}_2^*$, $\boldsymbol{\delta}_3^* = \dfrac{\boldsymbol{\delta}_1^*+\boldsymbol{\delta}_2^*}{2}$ and $\boldsymbol{\delta}_4^* = \dfrac{\boldsymbol{\delta}_1^*+\boldsymbol{\delta}_3^*}{2}$ of Example D.3.

Let $\phi_j = \{i_1, i_2, \ldots, i_s\} \in \Lambda$. Consider a vector $\boldsymbol{\nu} \in \mathbb{R}^s$ with $\nu_j \geqslant 0$ for all j and $\sum_{j=1}^{s} \nu_j = 1$. Define $\boldsymbol{\delta}^*(\boldsymbol{\nu}) = \sum_{j=1}^{s} \nu_j \boldsymbol{\delta}_{i_j}^*$. Then the function I_i defined by $I_i(\boldsymbol{\nu}) = \int_{t_0}^{t_N} l_i\big(t, \boldsymbol{\delta}^*(\boldsymbol{\nu})\big)\, dt$ is a convex function of $\boldsymbol{\nu}$.

Proof: We may assume without loss of generality that $i = 1$ and $\phi_j = \{1, 2, \ldots, s\}$. Define $\boldsymbol{x}^*(\boldsymbol{\nu}) = \sum_{j=1}^{s} \nu_j \boldsymbol{x}_j^*$. Since $\begin{bmatrix} \boldsymbol{x}^*(\boldsymbol{\nu}) \\ \boldsymbol{\delta}^*(\boldsymbol{\nu}) \end{bmatrix}$ is a convex combination of the vertices of V that are indexed by ϕ_j, it is a solution of the ELCP that corresponds to the given optimal traffic light switching problem. So $\boldsymbol{x}^*(\boldsymbol{\nu})$ and $\boldsymbol{\delta}^*(\boldsymbol{\nu})$ are compatible for the given $\boldsymbol{x}_0$. For each $\boldsymbol{x}_i^*$ with $i \in \phi_j$ we define a sequence $y_{i,0}, y_{i,1}, \ldots, y_{i,N}$ that contains the components of $\boldsymbol{x}_0$ and $\boldsymbol{x}_i^*$ that correspond to the queue length in lane L_1: $y_{i,0} = (\boldsymbol{x}_0)_1$ and $y_{i,j} = (\boldsymbol{x}_i^*)_{4(j-1)+1}$ for $j = 1, 2, \ldots, N$. Let $y_0 = (\boldsymbol{x}_0)_1$ and $y_j = (\boldsymbol{x}^*)_1$ for $j = 1, 2, \ldots, N$. Define $\delta_{i,k} = (\boldsymbol{\delta}_i^*)_{k+1}$ for $i = 1, 2, \ldots, s$ and $k = 0, 1, \ldots, N-1$. Note that $\sum_{j=1}^{s} \nu_j y_{j,0} = y_0 = (\boldsymbol{x}_0)_1$.

Since $\bar{\kappa}_1 < \bar{\lambda}_1$ we have

$$\tilde{y}_{2k+2} \stackrel{\text{def}}{=} l_1(t_{2k+2} - \delta_{\text{amb}}) = y_{2k+2} - (\bar{\lambda}_1 - \bar{\kappa}_1)\delta_{\text{amb}} \ .$$

If we define

$$\tilde{y}_{j,2k+2} = y_{j,2k+2} - (\bar{\lambda}_1 - \bar{\kappa}_1)\delta_{\text{amb}}$$

for all j, k, then we have

$$\begin{aligned}\sum_{j=1}^{s} \nu_j \tilde{y}_{j,2k+2} &= \sum_{j=1}^{s} \nu_j \left(y_{j,2k+2} - (\bar{\lambda}_1 - \bar{\kappa}_1)\delta_{\text{amb}}\right) \\ &= \sum_{j=1}^{s} \nu_j y_{j,2k+2} - \Big(\sum_{j=1}^{s} \nu_j\Big)(\bar{\lambda}_1 - \bar{\kappa}_1)\delta_{\text{amb}} \\ &= y_{2k+2} - (\bar{\lambda}_1 - \bar{\kappa}_1)\delta_{\text{amb}} \\ &= \tilde{y}_{2k+2} \quad .\end{aligned}$$

Furthermore, the assumption $\bar{\kappa}_1 < \bar{\lambda}_1$ also implies that $y_{2k+2} \neq 0$ and $y_{i,2k+2} \neq 0$ for all i, k. Recall that $\boldsymbol{x}^*(\boldsymbol{\nu})$ and $\boldsymbol{\delta}^*(\boldsymbol{\nu})$ are compatible for $\boldsymbol{x}_0$. As a consequence, we have (cf. Section C)

$$I_1(\boldsymbol{\nu}) \;=\; S_1(\boldsymbol{\nu}) + S_{2,1}(\boldsymbol{\nu}) - S_{2,2}(\boldsymbol{\nu}) + S_3(\boldsymbol{\nu}) + S_4(\boldsymbol{\nu})$$

with

$$S_1(\boldsymbol{\nu}) \;=\; \sum_{k=0}^{\lfloor \frac{N-1}{2} \rfloor} \frac{\left(\sum_{j=1}^{s} \nu_j\, y_{j,2k}\right) + \left(\sum_{j=1}^{s} \nu_j\, y_{j,2k+1}\right)}{2} \left(\sum_{j=1}^{s} \nu_j\, \delta_{j,2k}\right)$$

$$S_{2,1}(\boldsymbol{\nu}) \;=\; \sum_{\substack{k=0 \\ \sum_{j=1}^{s} \nu_j \tilde{y}_{j,2k+2} \neq 0}}^{\lfloor \frac{N}{2} \rfloor - 1} \frac{\left(\sum_{j=1}^{s} \nu_j\, y_{j,2k+1}\right) + \left(\sum_{j=1}^{s} \nu_j\, \tilde{y}_{j,2k+2}\right)}{2} \left(\sum_{j=1}^{s} \nu_j\, \delta_{j,2k+1}\right)$$

$$S_{2,2}(\boldsymbol{\nu}) \;=\; \sum_{\substack{k=0 \\ \sum_{j=1}^{s} \nu_j \tilde{y}_{j,2k+2} \neq 0}}^{\lfloor \frac{N}{2} \rfloor - 1} \frac{\left(\sum_{j=1}^{s} \nu_j\, y_{j,2k+1}\right) + \left(\sum_{j=1}^{s} \nu_j\, \tilde{y}_{j,2k+2}\right)}{2}\, \delta_{\text{amb}}$$

$$S_3(\boldsymbol{\nu}) \;=\; \sum_{\substack{k=0 \\ \sum_{j=1}^{s} \nu_j \tilde{y}_{j,2k+2} = 0}}^{\lfloor \frac{N}{2} \rfloor - 1} \frac{\left(\sum_{j=1}^{s} \nu_j\, y_{j,2k+1}\right)^2}{2\,(\bar{\mu}_1 - \bar{\lambda}_1)}$$

$$S_4(\boldsymbol{\nu}) \;=\; \sum_{k=0}^{\lfloor \frac{N}{2} \rfloor - 1} \frac{\left(\sum_{j=1}^{s} \nu_j\, \tilde{y}_{j,2k+2}\right) + \left(\sum_{j=1}^{s} \nu_j\, y_{j,2k+2}\right)}{2}\, \delta_{\text{amb}} \quad .$$

It easy to verify that S_3 is a convex, quadratic function of $\boldsymbol{\nu}$. Furthermore, $S_{2,2}$ and S_4 are linear functions of $\boldsymbol{\nu}$, which implies that they are also convex. So from now on, we only consider S_1 and $S_{2,1}$. Since S_1 and $S_{2,1}$ are continuous functions of the ν_j's, it is sufficient to prove that S_1 and $S_{2,1}$ are convex functions in the relative interior of the feasible region, i.e., for positive ν_j's. As a consequence, we have $\displaystyle\sum_{j=1}^{s} \nu_j \tilde{y}_{j,2k+2} \neq 0$ if and only if there exists an index $i \in \{1, 2, , \ldots, s\}$ such that $\tilde{y}_{i,2k+2} \neq 0$. Note that is condition is independent of the values of the ν_j's (provided that they are positive). If we define

$$
\begin{aligned}
z_{j,2k} &= y_{j,2k} + y_{j,2k+1} \\
z_{j,2k+1} &= \begin{cases} y_{j,2k+1} + \tilde{y}_{j,2k+2} & \text{if } \exists i \in \{1,2,,\ldots,s\} \text{ such that } \tilde{y}_{i,2k+2} \neq 0\,, \\ 0 & \text{otherwise}\,, \end{cases}
\end{aligned}
$$

for all j, k, then we have

$$
\begin{aligned}
S_1(\boldsymbol{\nu}) + S_{2,1}(\boldsymbol{\nu}) &= \sum_{k=0}^{N} \frac{\left(\displaystyle\sum_{j=1}^{s} \nu_j z_{j,k}\right)\left(\displaystyle\sum_{j=1}^{s} \nu_j \delta_{j,k}\right)}{2} \\
&= \sum_{k=0}^{N} \frac{1}{2}\, \boldsymbol{\nu}^T \boldsymbol{Q}_k \boldsymbol{\nu} \ .
\end{aligned}
$$

with

$$
\boldsymbol{Q}_k = \begin{bmatrix} z_{1,k} \\ z_{2,k} \\ \vdots \\ z_{s,k} \end{bmatrix} \begin{bmatrix} \delta_{1,k} & \delta_{2,k} & \ldots & \delta_{s,k} \end{bmatrix} .
$$

So $\boldsymbol{Q}_k$ is a matrix of rank 1 with nonnegative entries. This implies that $\boldsymbol{Q}_k$ is a positive semi-definite matrix and that $S_1 + S_{2,1}$ is a convex, quadratic function.
Hence, $I_1 = S_1 + S_{2,1} - S_{2,2} + S_3 + S_4$ is a convex, quadratic function of $\boldsymbol{\nu}$. □

Note that in order to obtain J_1, J_2, J_4 or J_5 we have to divide $\displaystyle\int_{t_0}^{t_N} l_1(t, \boldsymbol{\delta}^*(\boldsymbol{\nu}))\, dt$ by $t_N - t_0 = \displaystyle\sum_{k=0}^{N-1} \sum_{j=1}^{s} \nu_j \delta_{i_j,k}$ which is a linear function of $\boldsymbol{\nu}$. As a consequence, J_1, J_2, J_4 and J_5 are in general not convex functions of $\boldsymbol{\nu}$. However, computational experiments have shown that in most cases J_1, J_2, J_4 and J_5 are very smooth functions of $\boldsymbol{\nu}$ (for many faces they are even almost linear or almost convex). This means that finding the combination ν_1, ν_2, $\ldots$, ν_s for which $J_l\left(\displaystyle\sum_{j=1}^{s} \nu_j \boldsymbol{\delta}^*_{i_j}\right)$ with $l = 1, 2, 4$ or 5 reaches a global minimum is a well-behaved problem in the sense that for almost all initial starting points the same numerical solution (within a certain tolerance) will be obtained.

Since there exist very efficient algorithms to minimize convex objective functions over a convex feasible sets, we now examine whether the approximate objective functions $\tilde{J}_1$ and

$\tilde{J}_4$ are convex over the feasible set of the relaxed problem $\tilde{\mathcal{P}}$. Note that it makes only sense to determine convexity (or concavity) for objective functions that are strictly monotonous functions of $\boldsymbol{x}^*$, since only for these objective functions we can use Proposition 3.2 to minimize over the convex feasible set of the relaxed problem $\tilde{\mathcal{P}}$ instead of over the feasible set of the original problem $\mathcal{P}$.

The following example shows that the approximate objective functions $\tilde{J}_1$ and $\tilde{J}_4$ are in general neither convex nor concave as a function of $\boldsymbol{x}^*$ and $\boldsymbol{\delta}^*$.

Example D.5 Let $\delta_{\text{amb}} = 4$ and $\bar{\lambda}_i = 0.25$, $\bar{\mu}_i = 0.5$, $\bar{\kappa}_i = 0$ for $i = 1, 2, 3, 4$. Let

$$\boldsymbol{x}_0 = \begin{bmatrix} 4 \\ 0 \\ 4 \\ 0 \end{bmatrix}, \ \boldsymbol{w} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \ \boldsymbol{\delta}_1^* = \begin{bmatrix} 12 \\ 12 \end{bmatrix}, \ \boldsymbol{\delta}_2^* = \begin{bmatrix} 12 \\ 32 \end{bmatrix}, \ \boldsymbol{\delta}_3^* = \begin{bmatrix} 12 \\ 40 \end{bmatrix},$$

$$\boldsymbol{y}_1^* = \begin{bmatrix} 7 \\ 6 \end{bmatrix} \quad \text{and} \quad \boldsymbol{y}_2^* = \boldsymbol{y}_3^* = \begin{bmatrix} 7 \\ 1 \end{bmatrix} \ .$$

Note that the queue length sequence that corresponds to $\boldsymbol{y}_1^*$ is compatible with $\boldsymbol{\delta}_1^*$ for $l_1(0) = y_0 \stackrel{\text{def}}{=} (\boldsymbol{x}_0)_1$. This also holds for $\boldsymbol{y}_2^*$ and $\boldsymbol{\delta}_2^*$, and for $\boldsymbol{y}_3^*$ and $\boldsymbol{\delta}_3^*$. In Figure 11 we have plotted the evolution of l_1 as a function of time for the switching sequences defined by $\boldsymbol{\delta}_1^*$, $\boldsymbol{\delta}_2^*$, $\boldsymbol{\delta}_3^*$, $\boldsymbol{\delta}_4^* = \dfrac{\boldsymbol{\delta}_1^* + \boldsymbol{\delta}_2^*}{2}$ and $\boldsymbol{\delta}_5^* = \dfrac{\boldsymbol{\delta}_2^* + \boldsymbol{\delta}_3^*}{2}$.

If $y_0 \in \mathbb{R}^+$, $\boldsymbol{y}^* \in (\mathbb{R}^+)^N$ and $\boldsymbol{\delta}^* \in (\mathbb{R}_0^+)^N$, then $\tilde{l}_1(\cdot, \boldsymbol{y}^*, \boldsymbol{\delta}^*)$ is the piecewise-linear function that interpolates in the points (t_0, y_0), (t_1, y_1^*), ..., (t_N, y_N^*). Define

$$\tilde{f}(\boldsymbol{y}^*, \boldsymbol{\delta}^*) = \frac{\displaystyle\int_{t_0}^{t_N} \tilde{l}_1(t, \boldsymbol{y}^*, \boldsymbol{\delta}^*)}{t_N - t_0} \ .$$

We have $\tilde{f}(\boldsymbol{y}_1^*, \boldsymbol{\delta}_1^*) = 6$, $\tilde{f}(\boldsymbol{y}_2^*, \boldsymbol{\delta}_2^*) \approx 4.409$,

$$\tilde{f}\Big(\frac{\boldsymbol{y}_1^* + \boldsymbol{y}_2^*}{2}, \frac{\boldsymbol{\delta}_1^* + \boldsymbol{\delta}_2^*}{2}\Big) \approx 5.338 \quad \text{and} \quad \frac{\tilde{f}(\boldsymbol{y}_1^*, \boldsymbol{\delta}_1^*) + \tilde{f}(\boldsymbol{y}_2^*, \boldsymbol{\delta}_2^*)}{2} \approx 5.205 \ .$$

So

$$\tilde{f}\Big(\frac{\boldsymbol{y}_1^* + \boldsymbol{y}_2^*}{2}, \frac{\boldsymbol{\delta}_1^* + \boldsymbol{\delta}_2^*}{2}\Big) > \frac{\tilde{f}(\boldsymbol{y}_1^*, \boldsymbol{\delta}_1^*) + \tilde{f}(\boldsymbol{y}_2^*, \boldsymbol{\delta}_2^*)}{2} \ ,$$

which implies that $\tilde{f}$ is not convex.
On the other hand, we have $\tilde{f}(\boldsymbol{y}_2^*, \boldsymbol{\delta}_2^*) \approx 4.409$, $\tilde{f}(\boldsymbol{y}_3^*, \boldsymbol{\delta}_3^*) \approx 4.346$,

$$\tilde{f}\Big(\frac{\boldsymbol{y}_2^* + \boldsymbol{y}^*3}{2}, \frac{\boldsymbol{\delta}_2^* + \boldsymbol{\delta}_3^*}{2}\Big) = 4.375 \quad \text{and} \quad \frac{\tilde{f}(\boldsymbol{y}_2^*, \boldsymbol{\delta}_2^*) + \tilde{f}(\boldsymbol{y}_3^*, \boldsymbol{\delta}_3^*)}{2} \approx 4.378 \ .$$

So

$$\tilde{f}\Big(\frac{\boldsymbol{y}_2^* + \boldsymbol{y}_3^*}{2}, \frac{\boldsymbol{\delta}_2^* + \boldsymbol{\delta}_3^*}{2}\Big) < \frac{\tilde{f}(\boldsymbol{y}_2^*, \boldsymbol{\delta}_2^*) + \tilde{f}(\boldsymbol{y}_3^*, \boldsymbol{\delta}_3^*)}{2} \ ,$$

which implies that $\tilde{f}$ is not concave.

As a consequence, the objective functions $\tilde{J}_1$ and $\tilde{J}_4$ are in general neither convex nor concave.

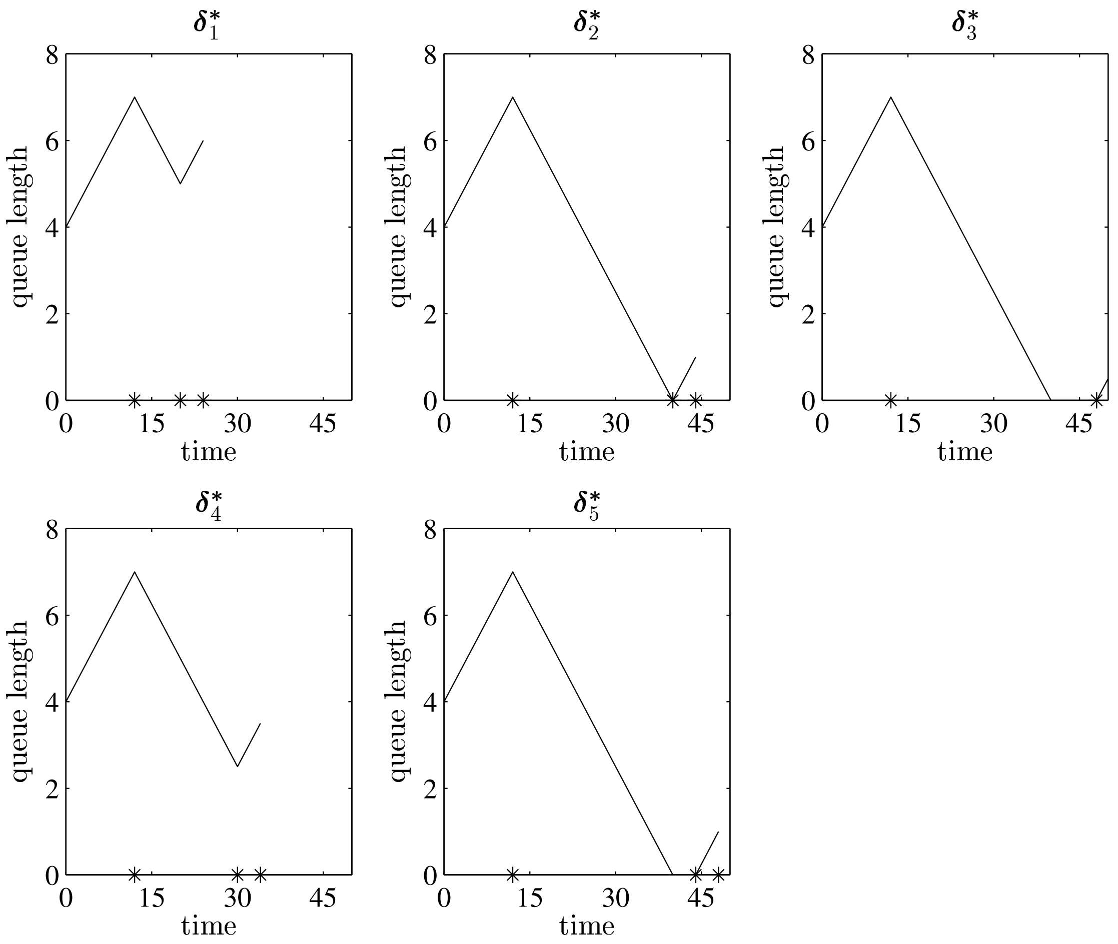

Figure 11: The evolution of the queue length in lane L_1 as a function of time t for the switching interval vectors $\boldsymbol{\delta}_1^*$, $\boldsymbol{\delta}_2^*$, $\boldsymbol{\delta}_3^*$, $\boldsymbol{\delta}_4^* = \dfrac{\boldsymbol{\delta}_1^* + \boldsymbol{\delta}_2^*}{2}$ and $\boldsymbol{\delta}_5^* = \dfrac{\boldsymbol{\delta}_2^* + \boldsymbol{\delta}_3^*}{2}$ of Example D.5. The $*$ signs on the time axis correspond to the switching time instants.

Indeed, for the queue length vectors $\boldsymbol{x}_1^*$, $\boldsymbol{x}_2^*$ and $\boldsymbol{x}_3^*$ that are compatible with respectively $\boldsymbol{\delta}_1^*$, $\boldsymbol{\delta}_2^*$ and $\boldsymbol{\delta}_3^*$ for $\boldsymbol{x}_0$, we have

$$\tilde{J}_l\Big(\frac{\boldsymbol{x}_1^* + \boldsymbol{x}_2^*}{2}, \frac{\boldsymbol{\delta}_1^* + \boldsymbol{\delta}_2^*}{2}\Big) \;>\; \frac{\tilde{J}_l(\boldsymbol{x}_1^*, \boldsymbol{\delta}_1^*) + \tilde{J}_l(\boldsymbol{x}_2^*, \boldsymbol{\delta}_2^*)}{2}$$

and

$$\tilde{J}_l\Big(\frac{\boldsymbol{x}_2^* + \boldsymbol{x}_3^*}{2}, \frac{\boldsymbol{\delta}_2^* + \boldsymbol{\delta}_3^*}{2}\Big) \;<\; \frac{\tilde{J}_l(\boldsymbol{x}_2^*, \boldsymbol{\delta}_2^*) + \tilde{J}_l(\boldsymbol{x}_3^*, \boldsymbol{\delta}_3^*)}{2}$$

for $l = 1$ and 4 (see Table 5). $\square$

E Another approximation for the objective functions J_1, J_2, J_3, J_4 and J_5

In this section we make again the following extra assumption:

l	$\tilde{J}_l(\boldsymbol{x}_1^*, \boldsymbol{\delta}_1^*)$	$\tilde{J}_l(\boldsymbol{x}_2^*, \boldsymbol{\delta}_2^*)$	$\tilde{J}_l\Big(\dfrac{\boldsymbol{x}_1^* + \boldsymbol{x}_2^*}{2}, \dfrac{\boldsymbol{\delta}_1^* + \boldsymbol{\delta}_2^*}{2}\Big)$	$\dfrac{\tilde{J}_l(\boldsymbol{x}_1^*, \boldsymbol{\delta}_1^*) + \tilde{J}_l(\boldsymbol{x}_2^*, \boldsymbol{\delta}_2^*)}{2}$
1	15.000	16.364	15.882	15.682
4	60.000	65.455	63.529	62.727

l	$\tilde{J}_l(\boldsymbol{x}_2^*, \boldsymbol{\delta}_2^*)$	$\tilde{J}_l(\boldsymbol{x}_3^*, \boldsymbol{\delta}_3^*)$	$\tilde{J}_l\Big(\dfrac{\boldsymbol{x}_2^* + \boldsymbol{x}_3^*}{2}, \dfrac{\boldsymbol{\delta}_2^* + \boldsymbol{\delta}_3^*}{2}\Big)$	$\dfrac{\tilde{J}_l(\boldsymbol{x}_2^*, \boldsymbol{\delta}_2^*) + \tilde{J}_l(\boldsymbol{x}_3^*, \boldsymbol{\delta}_3^*)}{2}$
1	16.364	18.154	17.250	17.259
4	65.455	72.615	69.000	69.035

Table 5: The values of the objective functions $\tilde{J}_1$ and $\tilde{J}_4$ (up to 3 decimal places) for the queue length vectors $\boldsymbol{x}_1^*$, $\boldsymbol{x}_2^*$ and $\boldsymbol{x}_3^*$ that are compatible with respectively the switching interval vectors $\boldsymbol{\delta}_1^*$, $\boldsymbol{\delta}_2^*$ and $\boldsymbol{\delta}_3^*$ of Example D.5.

- in each lane, the average departure rate when the light is amber is less than the average arrival rate of vehicles, i.e., $\bar{\kappa}_i < \bar{\lambda}_i$.

Recall that this assumption implies that the net queue growth rate during the amber phase $\bar{\lambda}_i - \bar{\kappa}_i$ is positive. As a consequence, the queue length at the end of the green phase $(t_k, t_{k+1} - \delta_{\text{amb}})$ with $k \in \mathcal{G}_i(N)$ is given by: $l_i(t_{k+1} - \delta_{\text{amb}}) = l_i(t_{k+1}) - (\bar{\lambda}_i - \bar{\kappa}_i)\delta_{\text{amb}}$ and the queue length at the end of the subsequent amber phase $l_i(t_{k+1})$ is positive.

For a given $\boldsymbol{x}_0$ and t_0, we define the function $\check{l}_i(\cdot, \boldsymbol{x}^*, \boldsymbol{\delta}^*)$ — or $\check{l}_i(\cdot)$ for short — as the piecewise-linear function that interpolates in the points $(t_0, l_i(t_0))$, $(t_{k+1} - \delta_{\text{amb}}, l_i(t_{k+1} - \delta_{\text{amb}}))$ for $k \in \mathcal{G}_i(N)$ — i.e., the points at the beginning and the end of the green phase for T_i — and the point $(t_N, l_i(t_N))$. The approximate objective functions $\check{J}_l$ for $l = 1, 2, 3, 4, 5$, are defined as in (8) – (12) but with l_i replaced by $\check{l}_i$.
The values of J_3 and $\check{J}_3$ always coincide. Now let $l \in \{1, 2, 4, 5\}$. Recall that the value of J_l and $\check{J}_l$ is determined by the surface under the functions l_i and $\check{l}_i$ respectively. If the queue lengths never become zero during the green phases and if no vehicles depart when the traffic light is amber (i.e., $\bar{\kappa}_i = 0$ for all i), then the functions l_i and $\check{l}_i$ and the values of J_l and $\check{J}_l$ coincide (cf. Figure 12). In practice, the departure rate during the amber phase will be small. Moreover, the length of the amber phase will also be small compared to the length of the green or the red phase. Furthermore, if we have an optimal traffic light switching scheme, then the periods during which the queue length in some lane is equal to 0 are in general short. As a consequence, for traffic light switching schemes in the neighborhood of the optimal scheme $\check{J}_l$ will be a good approximation of J_l. It is easy to verify that under normal circumstances $\check{J}_l$ will be a better approximation of J_l than $\tilde{J}_l$ (cf. Figures 2 and 12). However, in general (i.e., if we allow large values for δ_{amb}) we cannot impose a relative order on $\tilde{J}_l$ and $\check{J}_l$.

Using proofs that are similar to those of Propositions 3.1 and 3.3 it can be shown that the following two propositions hold:

Proposition E.1 *Let $\boldsymbol{x}_0 \in (\mathbb{R}^+)^4$, $\boldsymbol{x}^* \in (\mathbb{R}^+)^{4N}$ and $\boldsymbol{\delta}^* \in (\mathbb{R}_0^+)^N$. If $\boldsymbol{x}^*$ and $\boldsymbol{\delta}^*$ are compatible for $\boldsymbol{x}_0$ then we have $J_3(\boldsymbol{\delta}^*) = \check{J}_3(\boldsymbol{x}^*, \boldsymbol{\delta}^*)$, and $J_l(\boldsymbol{\delta}^*) \leqslant \check{J}_l(\boldsymbol{x}^*, \boldsymbol{\delta}^*)$ for $l = 1, 2, 4, 5$.*

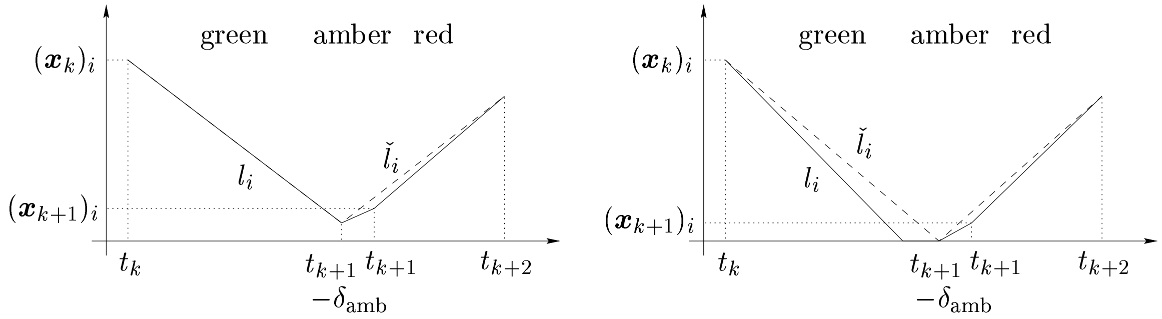

Figure 12: The functions l_i (full line) and $\check{l}_i$ (dashed line). The left plot shows a situation in which the queue length does not become 0 during the green phase and then l_i and $\check{l}_i$ coincide during the green phase. The right plot shows a situation where the queue length becomes 0 during the green phase.

Proposition E.2 *For given $\boldsymbol{x}_0$, δ_{amb}, $\bar{\lambda}_i$'s, $\bar{\mu}_i$'s, $\bar{\kappa}_i$'s and a given $\boldsymbol{\delta}^*$ the functions $\check{J}_1$ and $\check{J}_4$ are strictly monotonous functions of $\boldsymbol{x}^*$.*

So for $\check{J}_1$ and $\check{J}_4$ we can compute optimal traffic light switching schemes using the relaxed problem $\tilde{\mathcal{P}}$ instead of the original problem $\mathcal{P}$.
Let us now derive a formula for the evaluation of

$$\int_{t_0}^{t_N} \check{l}_1(t, \boldsymbol{x}^*, \boldsymbol{\delta}^*)\, dt = \sum_{k=0}^{N-1} \int_{t_k}^{t_{k+1}} \check{l}_1(t, \boldsymbol{x}^*, \boldsymbol{\delta}^*)\, dt \ .$$

Define $y_k = (\boldsymbol{x}_k)_1 = l_1(t_k)$ for $k = 0, 1, \ldots, N$ and $\tilde{y}_{2k+2} = l_1(t_{2k+2} - \delta_{\text{amb}})$ for $k = 0, 1, \ldots, \left\lfloor \dfrac{N}{2} \right\rfloor - 1$. Let the function *even* be defined by

$$even(n) = \begin{cases} 1 & \text{if } n \text{ is an even integer,} \\ 0 & \text{otherwise.} \end{cases}$$

Now it is easy to verify that

$$\begin{aligned} \int_{t_0}^{t_N} \check{l}_1(t, \boldsymbol{x}^*, \boldsymbol{\delta}^*)\, dt = \frac{y_0 + y_1}{2}\, \delta_0 \ + & \sum_{k=0}^{\lfloor \frac{N}{2} \rfloor - 1} \frac{y_{2k+1} + \tilde{y}_{2k+2}}{2}\, (\delta_{2k+1} - \delta_{\text{amb}}) \ + \\ & \sum_{k=0}^{\lfloor \frac{N-3}{2} \rfloor} \frac{\tilde{y}_{2k+2} + y_{2k+3}}{2}\, (\delta_{2k+2} + \delta_{\text{amb}}) \ + \ even(n)\, \frac{\tilde{y}_N + y_N}{2}\, \delta_{\text{amb}} \ . \end{aligned}$$

Note that in contrast to $\tilde{J}_1$ and $\tilde{J}_4$ making the assumption $\delta_k \approx \dfrac{t_N - t_0}{N}$ does not lead a linear objective function for $\check{J}_1$ and $\check{J}_4$.

Let us now compute a suboptimal traffic light switching scheme based on the objective function $\check{J}_1$ for the set-up of Example 5.1.

Example E.3 Consider the intersection of Figure 1 with the same data as in Example 5.1. Suppose that we want to compute a traffic light switching sequence t_0, t_1, …, t_7 that minimizes J_1. We use the `e04ucf` routine of the NAG library to compute a solution $\check{\boldsymbol{x}}^*$, $\check{\boldsymbol{\delta}}^*$

$\boldsymbol{\delta}^*, \boldsymbol{x}^*$	$J_1(\boldsymbol{\delta}^*)$	$\check{J}_1(\boldsymbol{x}^*, \boldsymbol{\delta}^*)$	$\tilde{J}_1(\boldsymbol{x}^*, \boldsymbol{\delta}^*)$	$\hat{J}_1(\boldsymbol{x}^*, \boldsymbol{\delta}^*)$	$J_{\text{lin}}(\boldsymbol{x}^*)$	CPU time
$\check{\boldsymbol{\delta}}^*, \check{\boldsymbol{x}}^*$	60.669	62.768	64.268	69.036	433.751	4.01
$\boldsymbol{\delta}^*_{\text{ELCP}}, \boldsymbol{x}^*_{\text{ELCP}}$	60.657	62.780	64.267	69.190	434.827	404.83
$\boldsymbol{\delta}^*_{\text{pen}}, \boldsymbol{x}^*_{\text{pen}}$	61.150	63.258	64.740	69.916	439.909	78.69
$\boldsymbol{\delta}^*_{\text{mul}}, \boldsymbol{x}^*_{\text{mul}}$	61.613	63.513	65.118	67.881	425.664	13.83
$\tilde{\boldsymbol{\delta}}^*, \tilde{\boldsymbol{x}}^*$	60.659	62.772	64.264	69.117	434.319	2.49
$\boldsymbol{\delta}^*_{\text{lin}}, \boldsymbol{x}^*_{\text{lin}}$	64.551	66.239	67.905	67.199	420.895	0.94
$\boldsymbol{\delta}^*_{\text{con}}, \boldsymbol{x}^*_{\text{con}}$	63.101	64.741	66.363	67.565	423.455	96.40

Table 6: The values of the objective functions J_1, $\check{J}_1$, $\tilde{J}_1$, $\hat{J}_1$ and J_{lin} (up to 3 decimal places) and the CPU time (up to 2 decimal places) needed to compute the suboptimal switching interval vector $\check{\boldsymbol{\delta}}^*$ of Example E.3 and the (sub)optimal switching interval vectors $\boldsymbol{\delta}^*_{\text{ELCP}}$, $\boldsymbol{\delta}^*_{\text{pen}}$, $\boldsymbol{\delta}^*_{\text{mul}}$, $\tilde{\boldsymbol{\delta}}^*$, $\boldsymbol{\delta}^*_{\text{lin}}$ and $\boldsymbol{\delta}^*_{\text{con}}$ of Example 5.1. The queue length vectors $\boldsymbol{x}^*$ are compatible with the switching interval vectors $\boldsymbol{\delta}^*$ for $\boldsymbol{x}_0$.

that minimizes the approximate objective function $\check{J}_1$ (using the relaxed problem $\tilde{\mathcal{P}}$). This yields[19]:

$$\check{\boldsymbol{\delta}}^* = [\,20.000 \quad 45.750 \quad 30.964 \quad 63.000 \quad 30.964 \quad 63.000 \quad 55.509\,]^T \ .$$

In Table 6 we have listed the values of the various objective functions for the switching interval vector $\check{\boldsymbol{\delta}}^*$ and for the switching interval vectors of Example 5.1. The evolution of the queue lengths for the traffic light control strategy that corresponds to $\check{\boldsymbol{\delta}}^*$ is represented in Figure 13. Clearly, the $\check{\boldsymbol{\delta}}^*$ solution also offers a good trade-off between optimality and efficiency. Note that for all the switching interval vectors of Table 6 the value of the objective function $\check{J}_1$ is lower than the value of the objective function $\tilde{J}_1$. □

[19] In this case using different starting points always leads to more or less the same numerical value of the optimal objective function: in an experiment with 20 random starting points the first 12 decimal places of the final objective function always had the same value. Therefore, we have only performed one run with an arbitrary random initial point here.

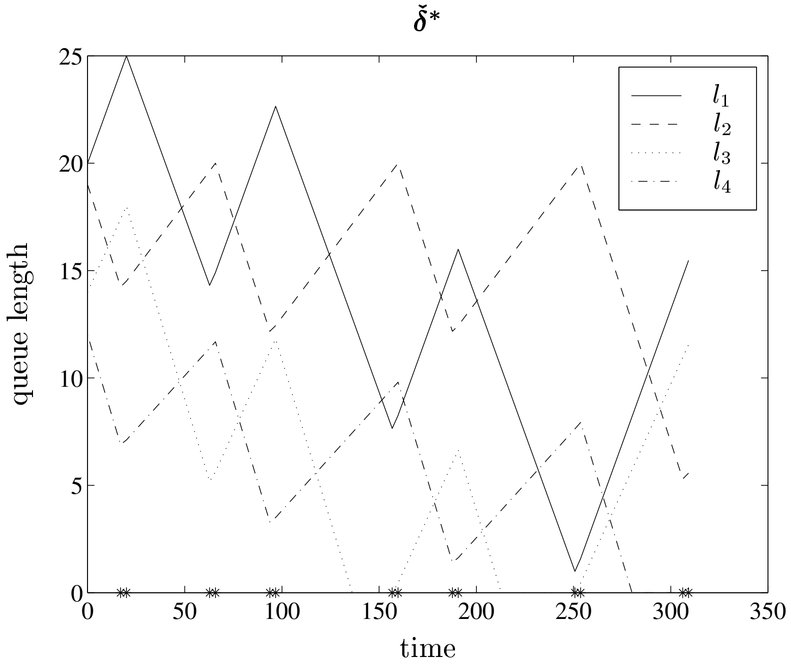

Figure 13: The queue lengths in the various lanes as a function of time for the traffic light switching sequence that corresponds to the switching interval vector $\check{\boldsymbol{\delta}}^*$ of Example E.3. The $*$ signs on the time axis correspond to the switching time instants.