URBAN NETWORK THROUGHPUT OPTIMIZATION VIA MODEL PREDICTIVE CONTROL USING THE LINK TRANSMISSION MODEL

G.S. van de Weg*
Department of Transport & Planning, Delft University of Technology
P.O. Box 5048, 2600 GA Delft, the Netherlands
E-mail: g.s.vandeweg@tudelft.nl

M. Keyvan-Ekbatani
Department of Transport & Planning, Delft University of Technology
P.O. Box 5048, 2600 GA Delft, the Netherlands
E-mail: m.ekbatani@tudelft.nl

A. Hegyi
Department of Transport & Planning, Delft University of Technology
P.O. Box 5048, 2600 GA Delft, the Netherlands
E-mail: a.hegyi@tudelft.nl

S.P. Hoogendoorn
Department of Transport & Planning, Delft University of Technology
P.O. Box 5048, 2600 GA Delft, the Netherlands
E-mail: s.p.hoogendoorn@tudelft.nl

* Corresponding author

Word count: 6060 words + (5 figures)*250 words = 7310 words

Submitted only for Presentation at the 2016 Transportation Research Board 95th Annual Meeting

Submission date: November 13, 2015
ABSTRACT

Developing practicable and efficient real-time signal control strategies for urban road networks is a major challenge with significant scientific and practical relevance. For the sake of practical feasibility, many urban traffic control strategies are based on simplified models. In this way, a trade-off can be made between controller accuracy, and (computational) complexity of the controller. However, designing a controller which operates efficiently with low computational time under different traffic conditions (i.e. under-saturated, saturated and over-saturated) remains a challenge. Several linear and quadratic model predictive control approaches are described in the literature to tackle this problem without considering the shock-wave dynamics of spill-back under over-saturated conditions. The principal contribution of this paper is the formulation of a linear model predictive controller that takes the shock-wave dynamics into account. This is realized by modeling the traffic dynamics in a link using the link transmission model. The performance of the proposed controller is compared with two other existing strategies. The total time spent by all the vehicles in the network and the computation time have been applied as performance indexes for the appraisal of the control strategies. Simulation results show that the control strategy proposed in this paper achieves better throughput under over-saturated conditions within comparable, low computation time.
1 INTRODUCTION

Traffic congestion is a growing problem in modern society. In recent years, the problem has often been addressed by the management of existing capacity. This requires efficient traffic management tools and has led to the implementation of advanced traffic control systems. The field of urban traffic control (UTC) has been studied in a variety of ways during the past decades and there is still space for new developments. This paper focuses on the problem of designing a traffic controller for the coordination of intersections that improves the throughput of an urban road network under all traffic conditions with the following features: 1) it optimizes the throughput in all traffic conditions, 2) the computation time of the controller is lower than the real-time sampling time and salable, and 3) the controller takes the future effects of the control action into account.

One of the complicating factors of UTC is that intersections are coupled differently in varying traffic conditions. In this paper we distinguish three traffic conditions. The first is the under-saturated condition which we define as the situation in which all queues can be emptied during a green-time implying that a coupling from upstream to downstream intersections exists. This is exploited by strategies that create green-waves, such as, MAXBAND (1). Other widely used strategies that are mainly effective in under-saturated conditions and applicable to large-scale networks are SCOOT (2) and SCATS (3). According to Papageorgiou et al. (4) the performance of SCOOT deteriorates in saturated traffic conditions. The saturated condition is the second condition and we define it as the situation in which queues cannot be emptied during a green-time implying that no direct coupling between intersections exist. The recently proposed max-pressure (or back-pressure) algorithms (5, 6) use this mechanism to distribute queues in an urban network using only data gathered in the vicinity of the intersection. Gregoire et al. (7) extended the max-pressure algorithm to deal with over-saturated conditions as well. The TUC strategy (8, 9) is a noteworthy example of a practice implemented control strategy designed for saturated conditions that is also capable of creating green-waves. Later, Aboudolas et al. (10) formulated the problem of network-wide signal control as a quadratic programming problem that aims at minimizing and balancing the link queues so as to minimize the risk of queue spill-back. The over-saturated condition is the third condition considered in this paper. It is characterized by queues that spill-back to upstream intersections causing a coupling from downstream intersections to upstream intersections. This can result in congestion propagation through the network or even gridlock (11). Gayah and Gao (12) showed that in an extremely congested network, adaptive traffic signals might have little to no effect on the network due to downstream congestion and queue spill-back. Hence, other strategies such as gating or perimeter control might be beneficial to alleviate instability under over-saturated traffic conditions. Recently, the notion of the network fundamental diagram (NFD) (13) has been exploited as a basis for the derivation of urban signal control approaches for over-saturated conditions. The combination of the NFD concept with gating or perimeter control of traffic flow lead to control strategies that deal with over-saturated conditions (see Keyvan-Ekbatani et al. (14) for single region gating; Geroliminis et al. (15) and Hajiahmadi et al. (16) for multi-region and Keyvan-Ekbatani et al. (17) for multiple concentric regions). Thus, many different control strategies exist which are mainly effective in only one or two of the traffic conditions.

When developing an UTC strategy for intersection coordination in all traffic conditions, it is required to consider the network-wide impact of intersection control which requires anticipation. Model predictive control (MPC) strategies are especially suited to optimize the performance of a network of intersections over a longer time-horizon. In the literature, various MPC strategies for the coordination of intersections can be found. Lo (18) formulate a mixed-integer linear programming (MILP) problem based on the Cell Transmission Model to optimize the signal timings. Van den Berg et al. (19) propose a non-linear MPC based on a detailed traffic flow model – which is an extension of the model of Kashani et al. (20) – in order to optimize the network throughput in all conditions. Lin et al. (21) propose a non-linear MPC based on a simplified version of that model, called the S-Model. This non-linear MPC approach is recast into a MILP
problem in Lin et al. (22) which can be more efficiently solved. Le et al. (23) propose a linear-quadratic MPC strategy for the optimization of both signal settings and turn fractions in all traffic conditions. Aboudolas et al. (10) also propose a linear-quadratic MPC strategy based on the store-and-forward model for traffic flow optimization in saturated conditions which can be efficiently solved.

The aforementioned approaches employ different traffic models with different properties resulting in different trade-offs in controller performance and computation time. All the approaches are capable of dealing with saturated and over-saturated conditions. The approach of Aboudolas et al. (10) has been specifically designed for (over-)saturated conditions and is not capable of considering free-flow conditions. The approach of Van den Berg et al. (19) uses a sampling time of 1 second so that signal programs can be taken into account which comes at the expense of considerably increased computation time. Lo (18) also explicitly considers signal timings but the authors mention that the computation time remains a challenge. In the other approaches, the traffic flow dynamics and signal programs are aggregated to periods of several (tens of) seconds, and it is assumed that the turn-fractions are known. In this way, the controller accuracy is reduced while drastically improving computation time, since, a linear or quadratic optimization problem has to be solved.

In all the forenamed models, except for the model used by Lo (18), the way in which spill-back is modeled neglects the shock-wave speed induced by spill-back. As a result, the capacity of a link is overestimated causing a waste of green-time in (over-)saturated conditions. This may affect the performance and efficiency of the optimization-based controllers. In this paper, it is shown that taking the shock-wave speed, and the free-flow travel time into account leads to a linear optimization problem with linear inequality constraints. This is realized by including the link transmission model (LTM) of Yperman (24) into the MPC framework. Hajiahmadi et al. (25) showed that an MPC strategy based on the LTM for freeway networks can be solved using a mixed-integer linear programming problem. The contribution of this paper is the reformulation of the LTM into a linear optimization problem for the control of link outflows in an urban road traffic network and showing that it results in an efficient controller.

2 TRAFFIC FLOW MODELING

The traffic model used in this paper is based on the LTM of (24) and consists of a link model for updating the link dynamics, and a node model to connect the links. The link dynamics are described by cumulative curves of the link inflow and outflow. Using these curves it is possible to model the free-flow travel-time which is present in under-saturated conditions, and the shock-wave dynamics which are present in over-saturated conditions. In situations where spill-back occurs, the node model distributes the spill-back over the upstream links, as detailed in Section 2.2.

2.1 Link dynamics

The traffic flow model is based on cumulative curves. Hence, the dynamics of traffic in a link $i$ ($\cdots$) are described by the cumulative inflow $N_{i\text{in}}(k)$ (veh) and cumulative outflow $N_{i\text{out}}(k)$ (veh) of the link. Note that the time-step $k$ ($\cdots$) refers to the period $t \in [T(k), T(k+1))$ (h) where the time $T$ (h) is the model sampling time. The LTM is capable of modeling under-saturated, saturated, and over-saturated conditions.

To model these conditions, several assumptions are made. First of all, it is assumed that traffic enters and exits a link under first-in-first-out (FIFO) conditions, and that traffic inside a link is not affected by disturbances. In order to model under-saturated conditions, it is assumed that the average free-flow travel time $t_{i\text{free}}^\text{free}$ (h) is known. Also, it is assumed that the saturation rates $q_{i\text{sat}}^\text{sat}$ (veh/h) are known in order to model saturated conditions and that the shock-wave travel time $t_{i\text{shock}}^\text{shock}$ (h) and the maximum link storage capacity $N_{i\text{max}}^\text{max}$ (veh) are known in order to model over-saturated conditions. As in related work, e.g. (10, 21), it is
where the set \( I \) is (-) contains the indexes of links feeding link \( i \), and the fraction \( \eta_{j,i}(k) \) is the fraction of traffic that wants to go from link \( j \) towards link \( i \) during time-step \( k \).

In over-saturated conditions, the maximum flow that fits in the link is determined by the link outflow. When there is no spill-back effects the cumulative in-flow is equal to the maximum cumulative flow \( N^\text{in,\,max}(k+1) \) at time-step \( k+1 \) and the fraction \( \eta_{j,i}(k) \) is the fraction of traffic that wants to go from link \( j \) towards link \( i \) during time-step \( k \).
In the case that the desired cumulative in-flow $N_{i}^{\text{in,des}}(k + 1)$ is larger than the maximum allowed cumulative in-flow $N_{i}^{\text{in,max}}(k + 1)$, the outflows out of the upstream links need to be reduced with a factor $r_{i}^{\text{in}}(k)$ (-) given by:

$$r_{i}^{\text{in}}(k) = \min\left[\frac{N_{i}^{\text{in,max}}(k + 1) - N_{i}^{\text{in}}(k)}{N_{i}^{\text{in,des}}(k + 1) - N_{i}^{\text{in}}(k)}, 1\right]. \tag{6}$$

In order to realize these reductions, a node model is required which will be detailed in the next section. This leads to the following reduced, realized cumulative outflows $N_{j}^{\text{out}}(k + 1)$:

$$N_{j}^{\text{out}}(k + 1) = N_{j}^{\text{out}}(k) + r_{i}^{\text{in}}(k)\left(N_{j}^{\text{out,max}}(k + 1) - N_{j}^{\text{out}}(k)\right). \tag{7}$$

When links are located at the entrances or exits of the network, some small adaptations to the model have to be implemented. First of all, when a link is an exit, the maximum cumulative outflow in (2) is determined by the maximum allowed outflow at the network exit links $q_{\text{out,max}}^{j}(k)$ (veh/h) instead of the saturation rate $q_{i}^{\text{sat}}$. This maximum outflow can be imposed as a disturbance to the model.

Secondly, when a link is an entrance, a vertical queue $j$ is placed at the entrance of the link. The cumulative inflow $N_{j}^{\text{O,in}}(k)$ (veh) to this queue is given by:

$$N_{j}^{\text{O,in}}(k + 1) = N_{j}^{\text{O,in}}(k) + q_{j}^{\text{in}}(k)T, \tag{8}$$

with $q_{j}^{\text{in}}(k)$ (veh/h) the demand at this entrance. The maximum cumulative outflow $N_{j}^{\text{O,out,max}}(k)$ (veh) is given by the minimum of the in-flow and the queue outflow capacity $q_{j}^{\text{cap}}$ (veh/h):

$$N_{j}^{\text{O,out,max}}(k + 1) = \min\left[N_{j}^{\text{O,out}} + q_{j}^{\text{cap}}(k)T, N_{j}^{\text{O,in}}(k + 1)\right], \tag{9}$$
where the cumulative number $N_j^O,\text{out} \ (\text{veh})$ is the realized cumulative outflow at the current time-step. Also, when the downstream link is full, the realized cumulative outflow $N_j^O,\text{out}$ will be reduced by a reduction factor $r_j^i(k)$:

$$N_j^O,\text{out}(k+1) = N_j^O,\text{out}(k) + r_j^i(k) \left( N_j^O,\text{out,max}(k+1) - N_j^O,\text{out}(k) \right). \quad (10)$$

The state $x_l^i(k)$ of link $i$ consists of the historical information of the cumulative outflow curves of the link up to the shock-wave time and the cumulative inflow curves up to the free-flow time:

$$x_i^L(k) = \begin{bmatrix} N_i^\text{out}(k) \\ \vdots \\ N_i^\text{out}(k-k_i^{\text{shock}}) \\ N_i^\text{in}(k) \\ \vdots \\ N_i^\text{in}(k-k_i^{\text{free}}) \end{bmatrix}. \quad (11)$$

Similarly, the state $x_j^O(k)$ of an origin is given by:

$$x_j^O(k) = \begin{bmatrix} N_j^O,\text{out}(k) \\ N_j^O,\text{in}(k) \end{bmatrix}. \quad (12)$$

### 2.2 Node model

In order to match the outflows with the inflows a node model is required. The node model that is used in this paper is similar to (26). An important difference is that in (26) the outflows are distributed according to the respective outflow capacities of the links while this is not considered here. This effect could be included in future work. The task of the node model is to distribute the reduction factors $r_j^i(k)$ of the links $I_i^{n,\text{out}}$ (-) going out of the node $l$ over the links $I_i^{n,\text{in}}$ (-) feeding the node $l$. In order to realize this, an iterative procedure has to be applied. The general idea is to find the outgoing link of node $l$ with the lowest reduction factor and apply this reduction factor to the links feeding this link. This reduces the maximum outflows out of these links, implying that capacity becomes available in other outgoing links. Thus, the reduction factors $r_j^i(k)$ of the outgoing links $I_i^{n,\text{out}}$ have to be updated accordingly and will increase or stay the same. This procedure is then repeated until all the reduction factors are equal to 1.

The procedure is as follows:

1. Initialize the node model. To this end, set $N_i^{\text{in, max}}(k+1) = N_i^{\text{in, max}}(k+1)$, set $N_i^{\text{in, des}}(k+1) = N_i^{\text{in, des}}(k+1)$, and set $\eta_{j,i}(k) = \eta_{j,i}(k)$;

2. Given $N_i^{\text{in, max}}(k+1)$, $N_i^{\text{in, des}}(k+1)$, and $N_i^{\text{in}}(k)$ compute the reduction factors $r_j^i(k)$ using (6) of all the links $I_i^{n,\text{out}}$ going out of node $l$;

3. If there exist any $r_j^i(k) < 1$, find the index $i_{r,\text{min}}$ (-) of the link with the smallest reduction factor:

$$i_{r,\text{min}} = \arg\min_{i \in I_i^{n,\text{out}}} r_j^i(k). \quad (13)$$

If all the factors are equal to 1, continue to step 10;

4. Find the set $I_i^{i_{r,\text{min}}}$ of links which have a turn-fraction $\tilde{\eta}_{j,i_{r,\text{min}}}(k) > 0$ towards link $i_{r,\text{min}}$,

7
5. Reduce the maximum outflow $\tilde{N}_{\text{out}}^{\text{max}}(k+1)$ of the links in $I_{\text{tr},\text{min}}^{0}$ with the factor $r_{\text{tr},\text{min}}^{\text{in}}(k)$:

$$
\tilde{N}_{\text{out}}^{\text{max}}(k+1) = N_{\text{out}}^{\text{max}}(k) + r_{\text{tr},\text{min}}^{\text{in}}(k)(\tilde{N}_{\text{out}}^{\text{max}}(k+1) - N_{\text{out}}^{\text{max}}(k)), \forall j \in I_{\text{tr},\text{min}}^{0}.
$$

6. Now the flows out of the links $I_{\text{tr},\text{min}}^{0}$ are known. Note that they will not be affected by any reduction factor anymore, so the maximum allowed inflows $\tilde{N}_{i}^{\text{in},\text{max}}(k+1)$ can be reduced with these values:

$$
\tilde{N}_{i}^{\text{in},\text{max}}(k+1) = \tilde{N}_{i}^{\text{in},\text{max}}(k+1) - \sum_{j \in I_{\text{tr},\text{min}}^{0}} \tilde{n}_{j,i}(k)(\tilde{N}_{\text{out}}^{\text{max}}(k+1) - N_{\text{out}}^{\text{max}}(k));
$$

7. Also, the desired cumulative inflows $\tilde{N}_{i}^{\text{in},\text{des}}(k+1)$ can be reduced with the reduced maximum outflows:

$$
\tilde{N}_{i}^{\text{in},\text{des}}(k+1) = \tilde{N}_{i}^{\text{in},\text{des}}(k+1) - \sum_{j \in I_{\text{tr},\text{min}}^{0}} \tilde{n}_{j,i}(k)(\tilde{N}_{\text{out}}^{\text{max}}(k+1) - N_{\text{out}}^{\text{max}}(k));
$$

8. Next, set all the turn-fractions $\tilde{n}_{j,i}(k)$ of the links $I_{\text{tr},\text{min}}^{0}$ to zero, since, their effect is already present in the cumulative curves $\tilde{N}_{i}^{\text{in},\text{max}}(k+1)$ and $\tilde{N}_{i}^{\text{in},\text{des}}(k+1)$:

$$
\tilde{n}_{j,i}(k) = 0, \forall j \in I_{\text{tr},\text{min}}^{0} \& \forall i \in I_{i}^{\text{in}};
$$

9. Return to step 2

10. When it holds that all $r_{\text{tr},\text{min}}^{\text{in}}(k) = 1$ the procedure can be stopped. Now, the cumulative outflows out of every incoming link $j$ of node $l$ have been determined:

$$
N_{\text{out}}^{\text{max}}(k+1) = \tilde{N}_{\text{out}}^{\text{max}}(k+1), \forall I_{i}^{\text{in}},
$$

and based on that, the cumulative inflows into the links $j$ exiting node $l$ can be computed:

$$
\tilde{N}_{i}^{\text{in}}(k+1) = N_{\text{in}}^{\text{in}}(k) + \sum_{j \in I_{i}^{\text{in}}} n_{j,i}(k)(N_{\text{out}}^{\text{max}}(k+1) - N_{\text{out}}^{\text{max}}(k)), \forall I_{i}^{\text{out}}.
$$

3 CONTROLLER DESIGN

The objective of the controller that is proposed in this paper is to optimize the throughput. This section shows that, even though the simulation model presented in Section 2 is non-linear, the resulting optimization problem is linear with linear inequality constraints. To this end, first the objective function that should be optimized will be presented. After that, the linear equations and inequality constraints for updating the link dynamics are presented in Section 3.1. Section 3.2 further specifies the linear optimization problem.

The objective of the controller is to optimize the throughput. This is equivalent to minimizing the difference between the cumulative inflow and outflow $N_{i}^{\text{in}}(k^{c}) - N_{i}^{\text{out}}(k^{c})$ of every link and the difference between the cumulative inflow and outflow $N_{j}^{\text{O,in}}(k^{c}) - N_{j}^{\text{O,out}}(k^{c})$ of every origin in the network:

$$
J(k^{c}, x) = \sum_{k^{c} = k_{0}}^{k_{0}+N^{p}} T^{c} \left\{ \sum_{i \in I^{L}} \left( N_{i}^{\text{in}}(k^{c}) - N_{i}^{\text{out}}(k^{c}) \right) + \sum_{j \in I^{O}} \left( N_{j}^{\text{O,in}}(k^{c}) - N_{j}^{\text{O,out}}(k^{c}) \right) \right\},
$$

with $I^{L}$ the set of all links, $I^{O}$ the set of all origins in the network, and $N^{p} (-)$ the prediction horizon. The time-step $k^{c} (-)$ indicates the time-steps of the controller and refers to the period $t \in [T^{c} k^{c}, T^{c} (k^{c} + 1)]$ (h) where the time $T^{c}$ (h) is the controller sampling time. The controller time-steps are related to the model time steps via the ratio $C \in \mathbb{N}^{+} (-)$ as follows $k^{c} = \left\lfloor k/C \right\rfloor$, where the mathematical operator $\lfloor \cdot \rfloor$ means rounding to the nearest integer that is equal to or smaller than the argument of the function.
3.1 The linear equations

The first step in the controller design is to replace the control signals $b_i(k)$ with the effective fractions $b_i^{L,\text{eff}}(k^c)$ of green-time used by the links, and the effective fractions $b_j^{O,\text{eff}}(k^c)$ of green-time used by the origin queues. Then, the node model can be replaced by linear inequality constraints by optimizing the signals $b_i^{L,\text{eff}}(k^c)$ and $b_j^{O,\text{eff}}(k^c)$.

In the linear model, the cumulative outflow $N_i^{\text{out}}(k^c)$ of link $I$ is updated using the following equation:

$$N_i^{\text{out}}(k^c + 1) = N_i^{\text{out}}(k^c) + b_i^{L,\text{eff}}(k^c)q_i^{\text{sat}}T^c ,$$

which should satisfy:

$$N_i^{\text{out}}(k^c + 1) \leq N_i^{\text{out}}(k^c) + b_i^{L,\text{eff}}(k^c)q_i^{\text{sat}}T^c + N_i^{\text{in}}(k^c) - k_i^{\text{free}} + 2 ,$$

with the number of time-steps $k_i^{\text{free}} = \lceil t_i^{\text{free}}/T^c \rceil (-)$, and the fraction $k_i^{\text{free}} = k_i^{\text{free}} - t_i^{\text{free}}/T^c (-)$. If the link is an exit, then the following constraint is added:

$$N_i^{\text{out}}(k^c + 1) \leq N_i^{\text{out}}(k^c) + b_i^{L,\text{eff}}(k^c)q_i^{\text{out,max}}(k^c)T^c .$$

The cumulative inflow $N_i^{\text{in}}(k^c)$ to link $i$ is updated using the following equation:

$$N_i^{\text{in}}(k^c + 1) = N_i^{\text{in}}(k^c) + \sum_{j \in I_i^{\text{in}}} \left( \eta_{j,i}(k^c)b_i^{L,\text{eff}}(k^c)q_i^{\text{sat}}T^c \right) ,$$

which should satisfy:

$$N_i^{\text{in}}(k^c + 1) \leq N_i^{\text{in}}(k^c) + \sum_{j \in I_i^{\text{in}}} \left( \eta_{j,i}(k^c)b_i^{L,\text{eff}}(k^c)q_i^{\text{sat}}T^c \right) + N_i^{\text{max}} ,$$

with the number of time-steps $k_i^{\text{shock}} = \lceil t_i^{\text{shock}}/T^c \rceil (-)$, and the fraction $k_i^{\text{shock}} = k_i^{\text{shock}} - t_i^{\text{shock}}/T^c (-)$. The cumulative inflow $N_j^{O,\text{in}}(k^c)$ to origin $j$ is updated as follows:

$$N_j^{O,\text{in}}(k^c + 1) = N_j^{O,\text{in}}(k^c) + q_j^{\text{in}}(k^c)T^c .$$

The cumulative outflow $N_j^{O,\text{out}}(k^c + 1)$ out of origin $j$ is updated using:

$$N_j^{O,\text{out}}(k^c + 1) = N_j^{O,\text{out}}(k^c) + b_j^{O,\text{eff}}(k^c)q_j^{\text{cap}}T^c ,$$

which should satisfy:

$$N_j^{O,\text{out}}(k^c + 1) \leq N_j^{O,\text{out}}(k^c) + q_j^{\text{in}}(k^c)T^c .$$

Apart from these dynamical update equations and constraints, constraints on the control signals should be added:

$$0 \leq b_i^{L,\text{eff}}(k^c) \leq 1 ,$$

$$0 \leq b_j^{O,\text{eff}}(k^c) \leq 1 ,$$

$$\sum_{i \in I_j^{\text{conflict}}} b_i^{L,\text{eff}}(k^c) \leq 1 ,$$
where the set \( I_j^{conflict} \) is the set \( j \) of signals which are in conflict with each other.

The state \( x_{i,c}^{c,L}(k_c) \) of link \( i \) has the following structure:

\[
x_{i,c}^{c,L}(k_c) = \begin{bmatrix}
N_i^{out}(k_c) \\
\vdots \\
N_i^{out}(k_c - k_i^{c, shock}) \\
N_i^{in}(k_c) \\
\vdots \\
N_i^{in}(k_c - k_i^{c, free})
\end{bmatrix}.
\]  

Similarly, the state \( x_{j,c}^{c,O}(k_c) \) of an origin has the following structure:

\[
x_{j,c}^{c,O}(k_c) = \begin{bmatrix}
N_j^{O, out}(k_c) \\
N_j^{O, in}(k_c)
\end{bmatrix}.
\]

### 3.2 The linear optimization problem

Based on the equations presented in the previous section and the objective function (34) the following linear optimization problem can be formulated:

\[
\min_{\bar{u}(k_c)} Z \bar{A} \bar{u}(k_c),
\]  

Subject to

\[
M^{ineq} \bar{u}(k_c) \leq V^{ineq},
\]

\[
M^{eq} \bar{u}(k_c) = V^{eq},
\]

where the vector \( \bar{u}(k_c) \) contains all the inputs that should be optimized. The vector \( Z \) adds all the differences between cumulative inflows and cumulative outflows the matrix \( \bar{A} \) is used to compute the prediction of the states \( \bar{x}(k_c) \), the matrix \( M^{ineq} \) and the vector \( V^{ineq} \) contain the inequality constraints, and the matrix \( M^{eq} \) and the vector \( V^{eq} \) contain the equality constraints. In the remainder of this section, the matrix \( \bar{A} \) will be specified.

In order to specify the matrix \( \bar{A} \), it is required to write the model described in Section 3.1 in the standard linear form:

\[
x(k_c + 1) = Ax(k_c) + B(k_c)u(k_c) + Cd(k_c),
\]  

with the state \( x(k_c) \) given by:

\[
x(k_c) = \begin{bmatrix} x_1^{L}(k_c) & \cdots & x_{n^L}(k_c) & x_1^{O}(k_c) & \cdots & x_{n^O}(k_c) \end{bmatrix}^T,
\]  

where the number \( n^L \) (-) is the number of links, and the number \( n^O \) (-) is the number of origins. The input vector \( u(k_c) \) given by:

\[
u(k_c) = \begin{bmatrix} b_1^{L, eff}(k_c) & \cdots & b_{n^L}^{L, eff}(k_c) & b_1^{O, eff}(k_c) & \cdots & b_{n^O}^{O, eff} \end{bmatrix}^T,
\]  

and the disturbance vector \( d(k_c) \) given by:

\[
d(k_c) = \begin{bmatrix} q_1^{in}(k_c) & \cdots & q_{n^O}^{in}(k_c) & q_1^{out, max}(k_c) & \cdots & q_{n^E}^{out, max}(k_c) \end{bmatrix}^T,
\]
where the number \( n^E (\cdot) \) is the number of exits.

The matrix \( A \) is a matrix with the matrices \( A^L_i \) and \( A^O_j \) of the links and origins respectively on its diagonal:

\[
A^L = \begin{bmatrix}
A^L_1 \\
& \ddots \\
& & A^L_n
\end{bmatrix},
\]

(42)

with the matrix \( A^L_i \) of a link given by:

\[
A^L_i = \begin{bmatrix}
1 \\
& \ddots \\
& & 1 \\
& & & 1 0
\end{bmatrix},
\]

(43)

and the matrix \( A^O_j \) of origin \( j \) is given by:

\[
A^O_j = \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix},
\]

(44)

The model in (38) can be simplified to:

\[
x(k^c + 1) = Ax(k^c) + \tilde{B}(k^c)\tilde{u}(k^c),
\]

(45)

by stacking stack \( u(k^c) \) and \( d(k^c) \) to give \( \tilde{u}(k^c) \):

\[
\tilde{u}(k^c) = [u(k^c) \quad d(k^c)]^T,
\]

(46)

where \( B(k^c), C, \) and \( d(k^c) \) can be combined to give \( \tilde{B}(k^c) \).

The matrix \( \tilde{B}(k^c) \) is defined as follows:

\[
\tilde{B}(k^c) = [\tilde{B}^L_1(k^c) \ \ldots \ \tilde{B}^L_{n_i}(k^c) \ \tilde{B}^O_1(k^c) \ \ldots \ \tilde{B}^O_{n_o}(k^c)]^T.
\]

(47)

where the matrix \( \tilde{B}^L_i(k^c) \) of link \( i \) is given as:

\[
\tilde{B}^L_i(k^c) = \begin{bmatrix}
\tilde{B}^L_{i,1}(k^c) & \tilde{B}^L_{i,2}(k^c) & \tilde{B}^L_{i,3}(k^c) & \tilde{B}^L_{i,4}(k^c)
\end{bmatrix},
\]

(48)

with

\[
\tilde{B}^L_{i,1}(k^c) =
\]

(49)
Now by creating a vector $\bar{u}(k^c)$ defined as:

$$\bar{u}(k^c) = [x(k^c_0) \; \bar{u}(k^c_0) \; \ldots \; \bar{u}(k^c + Np - 1)]^\top,$$

with the vector $\bar{u}(k^c)$ containing the traffic states at every time-step $x(k^c + n)$ from time-step $k^c_0$ to $k^c + Np$ defined as:

$$\bar{x}(k^c) = [x_{k_0+1} \; \ldots \; x_{k_0+Np}]^\top,$$
and the matrix $\tilde{A}$ defined as:

$$
\tilde{A} = \begin{bmatrix}
A & B(k^c) & 0 & \ldots & 0 \\
A^2 & AB(k^c) & B(k^c+1) & \ldots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
A^{N^p} & A^{N^p-1}B(k^c) & A^{N^p-2}B(k^c+1) & \ldots & B(k^c+N^p-1)
\end{bmatrix},
$$

(61)

a prediction of the evolution of the states is given by the following linear equation:

$$
\bar{x}(k^c) = \tilde{A}\bar{u}(k^c).
$$

(62)

By applying the receding horizon principle to this optimization problem an MPC strategy is obtained. The main concept of MPC is to find the optimal control signals $b_i^{L,\text{eff}}(k^c)$ by minimizing (34) over the prediction-horizon from $k_0^c, \ldots , k^c 0 + N^p$ given the traffic state at time $k_0^c T$ and a prediction of the future disturbances $q^{\text{in}}(k^c)$ and $q^{\text{out, max}}(k^c)$. The first sample of the optimal control signal is applied to the simulation model at time-step $k_0 T$ by setting:

$$
b_i(k) = b_i^{L,\text{eff}}(k_0^c) \forall k \in [k_0 T, \ldots (k_0 + 1)T) \& i \in I^L.
$$

(63)

Then, the procedure is repeated at the next time-step $(k_0 + 1)T$ when new measurements become available.

### 4 EVALUATION SET-UP

The controller is evaluated in order to assess the behavior and performance of the controller. The indicators that are used to assess the performance are the total time spent (TTS) by all the vehicles in the network and the computation time used by the controller. The performance of the controller is compared to the controllers of Aboudolas et al. (10) and Le et al. (23). These approaches are chosen, since, both can be formulated as linear optimization problems. However, it is expected that in over-saturated conditions the approach proposed in this paper will realize higher throughput improvements, since, it contains a more detailed description of shock-wave dynamics. In under-saturated conditions it is expected that the approach of Le et al. (23) will realize similar performance, while the approach of Aboudolas et al. (10) will realize worse performance because it does not consider free-flow travel times.

The network used for the evaluation is illustrated in Figure 2. Every link has a capacity of 2000 veh/h except for link 3 which contains a bottleneck that limits the out-flow of link 3 to 600 veh/h. This spill-back can affect several upstream links depending on the traffic demand pattern. The free-flow travel time in every link is set to 20 seconds, the shock-wave travel time to 40 seconds, and the maximum number of vehicles that fit in every link is set to 80 vehicles. An exception is link 6 which has a free-flow travel time of 40 seconds, a shock-wave travel time of 80 seconds, and a maximum storage capacity of 160 vehicles.

The turn-fractions are assumed to be constant and are given in Table 1. This table also shows the traffic demand patterns. The first pattern represents both under-saturated and saturated conditions. The second pattern represents under-saturated conditions only. It must be noted that in this situation the bottleneck is removed. The third pattern represents (over-)saturated conditions. In order to obtain a fair comparison, the control approach proposed in this paper is applied to this pattern during the first 100 seconds so that under-saturated conditions are not present. Also, the first 100 seconds are excluded from the TTS computation.

A scenario of 3600 seconds is evaluated and the simulation model – i.e. the model to which the control strategy is applied – sampling time is set to 1 second. The optimization model is sampled with 10 seconds and the prediction horizon is set to 600 seconds which was found by tuning. In general, the prediction horizon should be long enough so that the benefits of the control action are represented in the...
objective function. Every 60 seconds the MPC computes the optimal control signal for the next 600 seconds, and the optimal signal for the next 60 seconds – i.e. 6 controller sampling-time steps – is implemented to the simulation model. This value showed good performance while limiting the number of MPC computations.

The cell-transmission model of Daganzo (27) is chosen as the simulation model in combination with the node model as presented in Section 2.2. The reason why this simulation model is chosen is that (1) it can reproduce both free-flow travel times and spill-back, and (2) it enables a fair comparison, since, it is different from all of the optimization models that will be evaluated in this paper. In the simulation model, every link consists of 20 segments, except for link 6 which consists of 40 segments. The free-flow speed is set to 10 m/s, the congestion wave speed is set to 5 m/s, and the maximum number of vehicles allowed in every segment is set to 4. In this way, the maximum number of vehicles that fits in every link – except for link 6 – is 80, the free-flow travel-time is 20 seconds, and the shock-wave travel time is 40 seconds. For link 6 these values are 160 vehicles, 40 seconds, and 80 seconds respectively.

The evaluations are carried out using Matlab R2015a on a computer with a 3.6 GHz processor and 16 Gb RAM. The linear optimization is carried out using the ‘dual-simplex’ algorithm implemented in the standard linear optimization function ‘linprog’ of Matlab.

5 EVALUATION RESULTS

The quantitative results of the evaluation are presented in Table 2. It can be observed that in under-saturated conditions – i.e. demand pattern 2 – the method proposed in this paper realizes the same TTS as the approach of Le et al. (23). In that situation the approach of Aboudolas et al. (10) has a worse performance, since, it does not consider free-flow travel times. It can also be observed that in saturated conditions – i.e. demand pattern 3 –, the approach proposed in this paper has improved performance. The reason for this is that the controller can make more effective use of the available storage space in the links. The method proposed in this paper can realize a lower TTS for the first demand pattern as well.

From Table 2 it can also be observed that the maximum computation times used by the approach proposed in this paper are below 0.8 seconds. The approach of Aboudolas et al. (10) has the lowest computation time. The reason for this is that the dimension of the optimization problem – i.e. 1155 variables – is smaller compared to the dimension of the optimization problem – i.e. 1224 variables – proposed in this paper. The maximum computation time of the approach proposed by Le et al. (23) is the largest. The reason
TABLE 1 The turn-fractions and the demand patterns. Note that the demand patterns 2 and 3 consist of patterns of 2 minutes which are repeated over the entire simulation horizon of 3600 seconds.

<table>
<thead>
<tr>
<th>Turn fractions</th>
<th>Demand pattern 1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Interval (s)</td>
</tr>
<tr>
<td>$\eta_{1,2} = 0.78$</td>
<td>0-450</td>
</tr>
<tr>
<td>$\eta_{4,1} = 0.73$</td>
<td>450-1800</td>
</tr>
<tr>
<td>$\eta_{6,3} = 0.6$</td>
<td>1800-3600</td>
</tr>
<tr>
<td>$\eta_{8,6} = 0.56$</td>
<td>0-30</td>
</tr>
<tr>
<td>$\eta_{8,9} = 0.44$</td>
<td>30-50</td>
</tr>
<tr>
<td></td>
<td>50-90</td>
</tr>
<tr>
<td></td>
<td>90-120</td>
</tr>
<tr>
<td></td>
<td>0-50</td>
</tr>
<tr>
<td></td>
<td>30-50</td>
</tr>
<tr>
<td></td>
<td>50-90</td>
</tr>
<tr>
<td></td>
<td>90-120</td>
</tr>
</tbody>
</table>

for this is that a link is divided into segments – or classes – and for every class a dummy variable is added which has to be optimized resulting in 1826 optimization variables.

Figure 3 A shows the outflows of links 2, 3, 6, and 9, and Figure 3 B shows the number of vehicles in links 2, 3, and 6 for demand pattern 1. From these figures the qualitative behavior of the controller can be studied. Note that the demand pattern has been chosen in such a way that it represents all traffic conditions.

- During the first 450 seconds the demand is under-saturated. From Figure 3 A it can be observed that it takes some time before the flow reaches the links.

- After time 450 s the demand increases and the capacity of the bottleneck at link 3 is exceeded. This causes the number of vehicles in link 3 to increase. The number of vehicles in link 6 also starts to increase, since, the combined demand of link 2 and link 6 is approximately 2500 veh/h and the turn-fraction from 2 to 7 is larger compared to the turn-fraction from link 6 to 7, so it is beneficial to give priority to link 2.

- Around time 1350 s link 3 is full and the controller reduces the outflow of link 6 to 0 veh/h. The flow from link 2 to link 3 is then exactly 600 veh/h so that the inflow to link 3 is equal to its outflow. The outflow of link 7 is then 900 veh/h.

- At time 1390 s link 6 is full as well and now the outflow out of link 6 is increased to 1000 veh/h so that the queue does not spill-back to link 8 and blocking of link 9 is prevented. Simultaneously, the outflow from link 2 is reduced to 0 veh/h so that the number of vehicles in this queue starts to increase. This causes the outflow of link 9 to be preserved at 800 veh/h while the flow out of link 7 is reduced to 400 veh/h.

- Link 2 is full around time 1520 seconds. At that time, the controller reduces the outflow from link 6 to 0 veh/h which causes spill-back to link 8 but prevents spill-back to links 1 and 5. This spill-back reduces the outflow from link 9 from 800 veh/h to 0 veh/h and increases the outflow of link 7 to 600 veh/h and preserves the outflow of link 5 at 500 veh/h.
TABLE 2 Overview of the results comparing the maximum CPU time (MCPU) in seconds used by the optimization and the total time spent (TTS) in veh·h by all the vehicles in the network for the different demand patterns.

<table>
<thead>
<tr>
<th>Method</th>
<th>Demand pattern 1</th>
<th>Demand pattern 2</th>
<th>Demand pattern 3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MCPU</td>
<td>TTS</td>
<td>MCPU</td>
</tr>
<tr>
<td>This paper</td>
<td>0.80</td>
<td>184.59</td>
<td>0.58</td>
</tr>
<tr>
<td>Aboudolas et al. (10)</td>
<td>0.27</td>
<td>186.87(+1.2%)</td>
<td>0.30</td>
</tr>
<tr>
<td>Le et al. (23)</td>
<td>1.00</td>
<td>186.48(+1.0%)</td>
<td>1.05</td>
</tr>
</tbody>
</table>

- At time 1880 seconds the flow out of link 9 increases again, since then the demand has decreased again and the flow out of link 6 is increased as well.

Another observation that can be made from Figure 3 B is that the maximum number of vehicles that fits in a link changes over time. For instance, the maximum number of vehicles in link 2 at time 1200 s is smaller than the maximum number of vehicles in link 2 at time 1530 s. The reason for this is that the smaller the link outflow, the more vehicles can be stored in the link because less voids between vehicles have to propagate through the link. This behavior is not included in most MPC approaches.

![Figure 3](image.png)

**FIGURE 3** (A) The outflows out of links 2, 3, 6, and 9 over time. (B) The number of vehicles in links 2, 3, and 6 over time.
6 CONCLUSION AND RECOMMENDATIONS

This paper presented a linear model predictive control strategy for the optimization of the throughput of an urban road traffic network. The main contribution of this paper is that the linear MPC strategy considers the impact of the shock-wave speed caused by spill-back on the link capacity. This is realized by including the LTM model in a linear MPC framework. Evaluations showed that the approach can realize higher throughput compared to other approaches within a comparable amount of CPU time. The following assumptions have been made in this study which should be relaxed in future research: a) availability of a prediction of the turn-fractions, b) aggregated signal timings, c) no disturbances inside a link, and d) no noise or uncertainties.

ACKNOWLEDGEMENT

This work is part of the research programme ‘The Application of Operations Research in Urban Transport’, which is (partly) financed by the Netherlands Organisation for Scientific Research (NWO).

REFERENCES


