Delft University of Technology

Delft Center for Systems and Control

Technical report 08-010

Influencing route choice in traffic networks: A model predictive control approach based on mixed-integer linear programming*

M. van den Berg, B. De Schutter, J. Hellendoorn, and A. Hegyi

If you want to cite this report, please use the following reference instead:
M. van den Berg, B. De Schutter, J. Hellendoorn, and A. Hegyi, "Influencing route choice in traffic networks: A model predictive control approach based on mixed-integer linear programming," *Proceedings of the 17th IEEE International Conference on Control Applications*, San Antonio, Texas, pp. 299–304, Sept. 2008.

Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 CD Delft
The Netherlands
phone: +31-15-278.51.19 (secretary)
fax: +31-15-278.66.79
URL: `http://www.dcsc.tudelft.nl`

Influencing route choice in traffic networks: A model predictive control approach based on mixed-integer linear programming

M. van den Berg, B. De Schutter, J. Hellendoorn, A. Hegyi

***Abstract*— Traffic control measures like variable speed limits or outflow control can be used to influence the route choice of drivers. In this paper we develop a day-to-day route choice control method that is based on model predictive control (MPC). A basic route choice model forms the basis for the controller. We show that for the given model and for a linear cost function it is possible to reformulate the MPC optimization problem as a mixed-integer linear programming (MILP) problem. For MILP problems efficient branch-and-bound solvers are available that guarantee to find the global optimum. We also illustrate the efficiency of the proposed approach for a simple simulation example involving speed limit control.**

I. INTRODUCTION

Route choice takes place when there exist two or more routes between an origin and a destination. In this case, drivers select a route based on their preferences. The choices of the drivers lead to a traffic assignment, which describes how the vehicles are divided over the network. When drivers select their route solely based on their own preferences, this traffic assignment may lead to large traffic flows on narrow or dangerous roads, to socially undesired situations (e.g., too many vehicles in residential areas or near primary schools), or to too large flows near urban areas or nature reserves causing pollution and noise. Road administrators can try to prevent these unwanted situations by influencing the route choice of the drivers. In [1], [2] it has been shown that traffic control measures that do not directly influence route choice but that do have an impact on the travel time (such as traffic signals, variable speed limits, and ramp metering) can be used for this purpose.

Traffic control methods that incorporate the effect of control measures on route choice are described in, e.g., [3]–[5]. In this paper we consider control methods that can be used for steering the traffic flows in a network to a desired traffic assignment. This will result in settings for the outflow capacity of the links, or for the speed limits on these links.

We use model predictive control (MPC) [6], [7] as control method. As prediction model we use the static route choice model we have presented in [8]. This model is based on the assumption that the experienced travel time is the most important factor in route choice, which is also argued for in [9]. Moreover, the model allows for an analytical description of the behavior of traffic flows in a network. MPC uses this route choice model combined with an optimization algorithm to determine the optimal settings for the traffic control measures. MPC has already been applied previously for traffic control in, e.g., [10]–[12], where it resulted in nonlinear nonconvex continuous or mixed-integer optimization problems. In our specific case the whole control problem can be formulated as a mixed-integer linear programming (MILP) problem, for which fast solvers are available, which reduces the required computation time. Furthermore, the MILP approach also results in a globally optimal solution.

This paper is organized as follows. We first describe the route choice model and the MPC-based approach for route choice control in Section II. In Section III we reformulate the problem as an MILP problem. Next, the proposed control approach is applied to a simulation example in Section IV.

II. ROUTE CHOICE CONTROL

In this section we briefly recapitulate the day-to-day route choice model and the corresponding MPC-based route choice control approach that we have developed in [8].

A. Route choice model

To illustrate our approach we will use the simple two-route network given in Figure 1 throughout the paper. This network consists of one origin and one destination that are connected via two routes. Note however that the proposed approach can also be extended to more complex networks with multiple origins, multiple destinations, multiple route choice locations, and multiple routes.

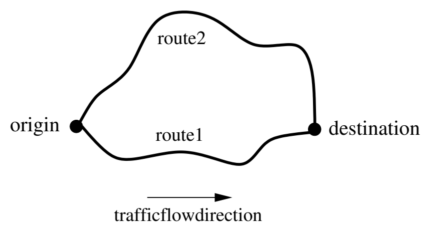

Fig. 1. Network with two routes.

1) Network variables: Consider the network of Figure 1. Each route r ($r \in \{1,2\}$) can be described by the following parameters, where d is the counter for the days. The length of route r is denoted by l_r (km), and its capacity is denoted by C_r (veh/h). The speed limit $v_r(d)$ (km/h) gives the maximum speed that is allowed on route r at day d. This speed limit

M. van den Berg, B. De Schutter, J. Hellendoorn are with the Delft Center for Systems and Control, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands, email: {monique.vandenberg,j.hellendoorn}@tudelft.nl, b@deschutter.info
B. De Schutter is also with the Marine and Transport Technology department of Delft University of Technology.
A. Hegyi is with the Department of Transport & Planning, Faculty of Civil Engineering and Geosciences, Delft University of Technology, P.O.Box 5048, 2600 GA Delft, The Netherlands, email: a.hegyi@tudelft.nl

will be bounded between a minimum speed limit $v_r^{\min}$ (km/h) and a maximum speed limit $v_r^{\max}$ (km/h). The outflow limit $Q_r(d)$ (veh/h) gives the number of vehicles per hour that are allowed to leave the route. The maximum value of the outflow limit, $Q_r^{\max}$ (veh/h), is equal to or lower than the actual capacity of the road: $Q_r^{\max} \leqslant C_r$. The minimum value $Q_r^{\min}$ (veh/h) can be selected to prevent almost total closure of the road when outflow control is applied.

We consider one part of the day, e.g., the morning peak. We denote this period by the time interval $[0,T]$ and we assume that the demand $Q_{\text{in}}(d)$ (veh/h) in the network is constant during $[0,T]$. The demand is distributed over the two routes according to the turning rate $\beta(d)$, which gives the percentage of the vehicles that select route 1.

An important characteristic of the routes is the "free-flow" travel time, which describes the time that a vehicle needs to travel a route when there is no delay due to congestion. The free-flow travel time at day d along route r is given by:

$$\tau_r^{\text{free}}(d) = \frac{l_r}{v_r(d)} \quad . \tag{1}$$

2) Travel time model: The model of [8] for the mean experienced travel time assumes that the travel time τ_r on a route has two components: the time spent in the queue τ_r^{queue} and the free-flow travel time τ_r^{free}:

$$\tau_r(d) = \tau_r^{\text{queue}}(d) + \tau_r^{\text{free}}(d) \quad .$$

The time in the queue τ_1^{queue} depends on the number of vehicles in the queue. We assume that the queues are vertical queues that build up at the end of each route. So during the period $[0,T]$ the queue grows as shown in Figure 2.

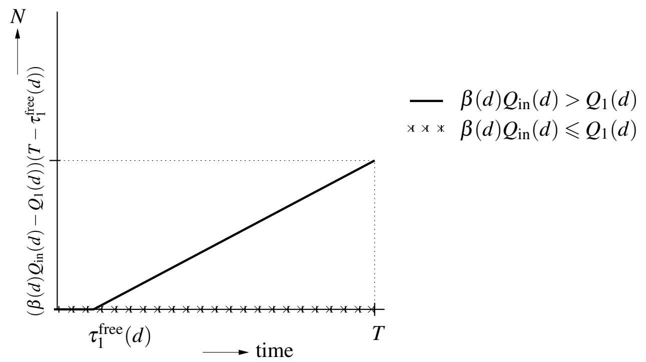

Fig. 2. Evolution of the queue length N on route 1 during the period $[0,T]$.

Since the number of vehicles leaving the queue per time unit is at most Q_1, the mean time that the vehicles spend in the queue at the end of route 1 is given by:

$$\begin{aligned}\tau_1^{\text{queue}}(d) &= \frac{N_1^{\text{mean}}(d)}{Q_1(d)} \\ &= \frac{(\beta(d)Q_{\text{in}}(d) - Q_1(d))(T - \tau_1^{\text{free}}(d))}{2Q_1(d)} \\ &\text{if } \beta(d)Q_{\text{in}}(d) > Q_1(d)\end{aligned}$$

and

$$\tau_1^{\text{queue}}(d) = 0 \qquad \text{if } \beta(d)Q_{\text{in}}(d) \leqslant Q_1(d) \ ,$$

where $N_1^{\text{mean}}(d)$ is the average number of vehicles in the queue at the end of route 1. Hence,

$$\tau_1^{\text{queue}}(d) = \max\left(0, \frac{(\beta(d)Q_{\text{in}}(d) - Q_1(d))(T - \tau_1^{\text{free}}(d))}{2Q_1(d)}\right) \ ,$$

and thus

$$\begin{aligned}\tau_1(d) = \max&\left(0, \frac{(\beta(d)Q_{\text{in}}(d) - Q_1(d))(T - \tau_1^{\text{free}}(d))}{2Q_1(d)}\right) \\ &+ \tau_1^{\text{free}}(d) \quad .\end{aligned} \tag{2}$$

A similar reasoning for route 2 results in

$$\begin{aligned}\tau_2(d) = \max&\left(0, \frac{((1-\beta(d))Q_{\text{in}}(d) - Q_2(d))(T - \tau_2^{\text{free}}(d))}{2Q_2(d)}\right) \\ &+ \tau_2^{\text{free}}(d) \quad .\end{aligned} \tag{3}$$

3) Route choice model: Route choice models describe the route choice of drivers at locations where a route must be selected. The model of [8] updates the turning rates based on the difference in travel times between the two routes while also taking into account that the turning rates are bounded between 0 and 1. This yields

$$\beta(d+1) = \min\left(1, \max(0, \beta(d) + \kappa(\tau_2(d) - \tau_1(d)))\right) \ . \tag{4}$$

Here κ includes the fraction of drivers that change their route from one day to the next based on the travel time difference.

B. Route choice control using MPC

1) Outflow control and speed limit control: Now two control inputs can be selected for influencing the route choice of the drivers with existing control methods: outflow limits and speed limits. Both inputs influence the travel time of the drivers, and thus indirectly the route choice. Outflow control limits the flow that can leave a link. The outflow can be lowered using, e.g., traffic signals and ramp metering installations. The control via the speed limits influences the free-flow travel time on the two routes. Variable message signs could be used to display the speed limits.

2) MPC for route choice control: Just as in [8] we will use Model Predictive Control (MPC) [13] to determine the optimal values for the outflow control limits and speed control limits. Below we will briefly present this method.

In MPC for route choice control the goal is to determine the control inputs c that optimize a cost function J over a given prediction period of N_{p} days ahead, given the current state of the network, the future demand, and a model of the system, and subject to operational and other constraints. This results in a sequence of optimal control inputs $c^*(d), c^*(d+1), \ldots, c^*(d+N_{\text{p}}-1)$. To reduce the computational complexity often a control horizon N_{c} ($N_{\text{c}} < N_{\text{p}}$) is introduced and the control sequence is constrained to vary only for the first N_{c} days, after which the control inputs are set to stay constant (i.e., $c(d+j) = c(d+N_{\text{c}}-1)$ for $j = N_{\text{c}}, \ldots, N_{\text{p}}-1$).

MPC uses a receding horizon approach, i.e., of the optimal control signal sequence only the first sample $c^*(d)$ is applied to the system. Next, at day $d+1$, the procedure is repeated given the new state of the system, and a new optimization is performed for days $d+1$ up to $d+N_\mathrm{p}$. Of the resulting control signal again only the first sample is applied, and so on. This is called the receding horizon approach.

In the context of route choice control typical examples of cost functions are the total time the vehicles spend in the network, the total queue length, or the norm of the difference between the realized flows and the desired flows on the routes. These cost functions serve either to handle as much traffic as possible in a short time, or to keep vehicles away from protected routes (e.g., routes through residential areas or nature reserves).

The state of the system is in our case given by the mean travel times, and the mean turning rates for the day. As (prediction) model we could use the route choice model of Section II. The control inputs are the outflow limits and/or the speed limits. Typical constraints are maximum and minimum values for these limits as well as maximal travel times or maximal waiting times in the queues.

In [8] the control signal were assumed to be real-valued. This results in continuous nonlinear nonconvex optimization problems that could be solved using multi-start local search methods (like SQP, pattern search, etc.) or (semi-)global optimization methods like genetic algorithms or simulated annealing [14]. Note that these approaches in principle only yield a suboptimal solution as in practice it is not tractable to find the global optimum of the continuous nonlinear nonconvex optimization problems that arise in MPC for route choice control.

In the remainder of this paper we will only allow discrete values for the control input (in particular for the speed limits). In the next section we will show that for linear cost functions this will then result in a mixed-integer linear programming (MILP) problem, for which efficient solvers exist that guarantee to find the global optimum.

III. MPC FOR ROUTE CHOICE CONTROL USING MIXED-INTEGER PROGRAMMING

In this section we show that for linear cost functions the MPC route choice optimization problem can be recast as an MILP problem.

A. Rules for creating mixed-integer linear inequalities

To formulate the route choice control problem described above as an MILP problem, we first have to remove the nonlinearities from the model. This is done by recasting the nonlinear equations into linear ones, and by introducing additional auxiliary variables. To perform these transformations we use the following equivalences [15], where δ represents a binary valued scalar variable, y a real valued scalar variable, and f a function defined on a bounded set X with upper and lower bounds M and m for the function values:

P1: $[f(x) \leqslant 0] \leftrightarrow [\delta = 1]$ is true if and only if

$$\begin{cases} f(x) \leqslant M(1-\delta) \\ f(x) \geqslant \varepsilon + (m-\varepsilon)\delta \end{cases} ,$$

where ε is a small positive number (typically the machine precision),

P2: $y = \delta f(x)$ is equivalent to

$$\begin{cases} y \leqslant M\delta \\ y \geqslant m\delta \\ y \leqslant f(x) - m(1-\delta) \\ y \geqslant f(x) - M(1-\delta) \end{cases} .$$

B. Reformulating the MPC route choice control problem

In this section we will consider the case of speed control with no outflow control. For outflow control without speed limit control and for combined speed and outflow control a similar reasoning in combination with Properties **P1** and **P2** will also result in an MILP problem.

1) Model equations: For simplicity we assume that the speed limits can only have two values v_a and v_b (note however that an extension to more than two values is straightforward). The free-flow travel times corresponding to these values can be represented by one binary variable δ as follows. Define (cf. (1))

$$\tau_{r,a}^{\mathrm{free}} = \frac{l_r}{v_a}, \quad \tau_{r,b}^{\mathrm{free}} = \frac{l_r}{v_b}, \text{ and } \Delta_r = \tau_{r,b}^{\mathrm{free}} - \tau_{r,a}^{\mathrm{free}} .$$

Then we can select v_a or v_b on route r for day d by introducing a binary variable $\delta_r(d)$ and setting

$$\tau_r^{\mathrm{free}}(d) = \tau_{r,a}^{\mathrm{free}} + \Delta_r \delta_r(d) .$$

Recall that we consider the case of speed control with no outflow control; so $Q_1(d) = C_1$ and $Q_2(d) = C_2$ for all d. If we substitute the above expression for $\tau_1^{\mathrm{free}}(d)$ in (2) we get

$$\tau_1(d) = \max(0, y_3(d)) + \tau_{1,a}^{\mathrm{free}} + \Delta_1 \delta_1(d) \tag{5}$$

with

$$y_3(d) = a_1\beta(d) + a_2\delta_1(d)\beta(d) + a_3\delta_1(d) + a_4 \tag{6}$$

with $a_1 = \frac{1}{2C_1}Q_{\mathrm{in}}(d)(T - \tau_{1,a}^{\mathrm{free}})$, $a_2 = -\frac{1}{2C_1}Q_{\mathrm{in}}(d)\Delta_1$, $a_3 = \frac{1}{2}\Delta_1$, and $a_4 = -\frac{1}{2}(T - \tau_{1,a}^{\mathrm{free}})$. By introducing an extra variable $y_1(d) = \delta_1(d)\beta(d)$ and using Property **P2** with $f(x) = \beta(d)$, $m = 0$, and $M = 1$, (6) can be transformed into a system of linear inequalities. In a similar way $\tau_2(d)$ can be expressed as

$$\tau_2(d) = \max(0, y_4(d)) + \tau_{2,a}^{\mathrm{free}} + \Delta_2 \delta_2(d) \tag{7}$$

with $y_4(d)$ given by

$$y_4(d) = a_5\beta(d) + a_6 y_2(d) + a_7\delta_2(d) + a_8$$

with $a_5 = -\frac{1}{2C_2}Q_{\mathrm{in}}(d)(T - \tau_{2,a}^{\mathrm{free}})$, $a_6 = \frac{1}{2C_2}Q_{\mathrm{in}}(d)\Delta_2$, $a_7 = -\frac{1}{2C_2}\Delta_2(Q_{\mathrm{in}}(d) - C_2)$, and $a_8 = \frac{1}{2C_2}(Q_{\mathrm{in}}(d) - C_2)(T - \tau_{2,a}^{\mathrm{free}})$, and with $y_2(d) = \delta_2(d)\beta(d)$. Using Property **P2** these equations can also be transformed into a system of linear inequalities.

Now define the auxiliary variables $\eta(d)$ and $\gamma(d)$ such that (cf. (4))

$$\gamma(d) = \beta(d) + \kappa(\tau_2(d) - \tau_1(d)) \tag{8}$$

$$\eta(d) = \max(0, \gamma(d)) . \tag{9}$$

Then we have

$$\beta(d+1) = \min(\eta(d), 1) . \tag{10}$$

Let us now discuss how these equations can be recast as a system of mixed-integer linear inequalities.

Combining (5), (7), and (8) we get

$$\begin{aligned}\gamma(d) = \beta(d) - \kappa \max(0, y_3(d)) + \kappa \max(0, y_4(d)) \\ - \kappa\Delta_1\delta_1(d) + \kappa\Delta_2\delta_2(d) + \kappa(\tau^{\text{free}}_{2,a} - \tau^{\text{free}}_{1,a}) \ .\end{aligned}$$

Now we define binary variables $\delta_3(d)$ and $\delta_4(d)$ such that $\delta_3(d) = 1$ if and only if $y_3(d) \geqslant 0$, and $\delta_4(d) = 1$ if and only if $y_4(d) \geqslant 0$. Note that using Property **P1** these equivalences can be recast as a system of linear inequalities. Now we have $\max(0, y_3(d)) = \delta_3(d)y_3(d)$ and $\max(0, y_4(d)) = \delta_4(d)y_4(d)$. So after introducing $y_5(d) = \delta_3(d)y_3(d)$ and $y_6(d) = \delta_4(d)y_4(d)$ and noting that both these expressions can be recast as a system of linear inequalities via Property **P2**, we get

$$\gamma(d) = \beta(d) - \kappa y_5(d) + \kappa y_6(d) + b_1\delta_1(d) + b_2\delta_2(d) + b_3 \tag{11}$$

with $b_1 = -\kappa\Delta_1$, $b_2 = \kappa\Delta_2$, and $b_3 = \kappa(\tau^{\text{free}}_{2,a} - \tau^{\text{free}}_{1,a})$. Note that equation (11) is linear.

Now consider (9). If we define the binary variable $\delta_5(d)$ such that $\delta_5(d) = 1$ if and only if $\gamma(d) \geqslant 0$ (note that this equivalence can be recast as a system of linear inequalities via Property **P2**), we get $\eta(d) = \delta_5(d)\gamma(d)$, which can in its turn also be expressed as a system of linear inequalities using Property **P2**.

Consider (10) and define the binary variable $\delta_6(d)$ such that

$$\delta_6(d) = 1 \text{ if and only if } \eta(d) \leq 1 \ .$$

Note that this equivalence can be recast as a system of linear inequalities via Property **P2**. It is easy to verify that now we have

$$\beta(d+1) = \min(\eta(d), 1) = \delta_6(d)\eta(d) + 1 - \delta_6(d) \ ,$$

which after introducing the auxiliary variable $z(d) = \delta_6(d)\eta(d)$ (this equivalence can also be recast as a system of linear inequalities via Property **P1**), results in the linear equation

$$\beta(d+1) = z(d) + 1 - \delta_6(d) \ .$$

If we now collect all variables for day d in one vector $w(d) = \big[\beta(d)\ \delta_1(d)\ \dots\ \delta_6(d)\ y_1(d)\ \dots\ y_6(d)\ \gamma(d)\ \eta(d)\ z(d)\big]^{\text{T}}$, we can express $\beta(d+1)$ as an affine function of $w(d)$: $\beta(d+1) = aw(d) + b$ for a properly defined vector a and scalar b, where $w(d)$ satisfies a system of linear equations $Cw(d) = e$, $Fw(d) \leqslant g$, which corresponds to the various linear equations and constraints introduced above.

C. Cost function

To be able to transform the route choice control problem into an MILP problem, the cost function should be linear or piecewise affine. Possible goals of the controller that allow for such cost functions are reaching a desired flow on one of the routes, or minimizing the flow on a route (with as constraint, e.g., a maximum allowed travel time on the other route — see also Section III-D). The MPC cost function for a minimum flow on route 1 is given by:

$$J(d) = \min \sum_{j=1}^{N_{\text{p}}} \beta(d+j)Q_{\text{in}}(d+j) \ .$$

Let us define

$$\tilde{F}(d) = \begin{bmatrix} \beta(d+1)Q_{\text{in}}(d+1) \\ \vdots \\ \beta(d+N_{\text{p}})Q_{\text{in}}(d+N_{\text{p}}) \end{bmatrix},$$

$$\tilde{F}^{\text{desired}}(d) = \begin{bmatrix} q_1^{\text{desired}}(d+1) \\ \vdots \\ q_1^{\text{desired}}(d+N_{\text{p}}) \end{bmatrix},$$

where $q_1^{\text{desired}}(d+j)$ denotes the desired flow on route 1 at day $d+j$. The MPC cost function corresponding to reaching a desired flow on route 1 is then given by:

$$J(d) = \min \|\tilde{F}^{\text{desired}}(d) - \tilde{F}(d)\|$$

using either the 1-norm or the ∞-norm. When a 1-norm is used, the problem can transformed into a linear one as follows:

$$\begin{aligned}&\min \|\tilde{F}^{\text{desired}}(d) - \tilde{F}(d)\|_1 \\ &= \min \sum_{j=1}^{N_{\text{p}}} |q_1^{\text{desired}}(d+j) - \beta(d+j)Q_{\text{in}}(d+j)| \\ &= \min \sum_{j=1}^{N_{\text{p}}} q(d+j) \\ &\quad \text{s.t. } q(d+j) \geqslant q_1^{\text{desired}}(d+j) - \beta(d+j)Q_{\text{in}}(d+j) \\ &\quad\quad\ \ q(d+j) \geqslant -q_1^{\text{desired}}(d+j) + \beta(d+j)Q_{\text{in}}(d+j) \\ &\quad\quad\ \ \text{for } j = 1, \dots, N_{\text{p}}.\end{aligned}$$

It is easy to verify that for the optimal solution of the latter problem we have

$$\begin{aligned}q^*(d+j) &= \max\big(q_1^{\text{desired}}(d+j) - \beta^*(d+j)Q_{\text{in}}(d+j), \\ &\qquad -q_1^{\text{desired}}(d+j) + \beta^*(d+j)Q_{\text{in}}(d+j)\big) \\ &= |q_1^{\text{desired}}(d+j) - \beta^*(d+j)Q_{\text{in}}(d+j)|\end{aligned}$$

for all j.

Similarly, for the ∞-norm we have

$$\begin{aligned}&\min \|\tilde{F}^{\text{desired}}(d) - \tilde{F}(d)\|_\infty \\ &\min \max_{j=1,\dots,N_{\text{p}}} |q_1^{\text{desired}}(d+j) - \beta(d+j)Q_{\text{in}}(d+j)| \\ &= \min q \\ &\quad \text{s.t. } q \geqslant q_1^{\text{desired}}(d+j) - \beta(d+j)Q_{\text{in}}(d+j) \\ &\quad\quad\ \ q \geqslant -q_1^{\text{desired}}(d+j) + \beta(d+j)Q_{\text{in}}(d+j) \\ &\quad\quad\ \ \text{for } j = 1, \dots, N_{\text{p}},\end{aligned}$$

which is also a linear programming problem.

D. Constraints

It might be useful to add a constraint on the travel time on the second route, because minimizing, e.g., the flow on route 1 results in a higher flow and thus a longer travel time on route 2:

$$\tau_2(d+j) \leqslant \tau_2^{\max}(d+j) \quad \text{for } j = 0,\ldots,N_\mathrm{p}-1 \ , \tag{12}$$

where $\tau_2^{\max}(d+j)$ denotes the maximal travel time on route 2 on day $d+j$. Note that $\tau_2(d+j)$ will not be a variable in the optimization problem. However, using (7) we can easily eliminate it from the constraint (12). This yields the equivalent system of constraints

$$\tau_{2,a}^{\text{free}} + \Delta_2\delta_2(d+j) \leqslant \tau_2^{\max}(d+j)$$
$$y_4(d+j) + \tau_{2,a}^{\text{free}} + \Delta_2\delta_2(d+j) \leq \tau_2^{\max}(d+j)$$

for $j = 0,\ldots,N_\mathrm{p}-1$. Note that these constraints are also linear.

An alternative constraint is to have a minimal or maximal flow on a given route. For route 2 this would result in

$$F_2^{\min}(d+j) \leqslant (1-\beta(d+j))Q_{\text{in}}(d+j) \leqslant F_2^{\max}(d+j) \ ,$$

for $j = 1,\ldots,N_\mathrm{p}$, where $F_2^{\min}(d+j)$ and $F_2^{\max}(d+j)$ denote respectively the minimal and maximal allowed flow on route 2 on day $d+j$. This constraint is also linear.

E. MILP

If we collect the linear objective function and all the linear constraints introduced above into one big problem, we get an MILP problem in the variables $w(d), w(d+1),\ldots,w(d+N_\mathrm{p}-1), \beta(d+N_\mathrm{p})$ and $q(d+1), q(d+2),\ldots,q(d+N_\mathrm{p})$ (when the 1-norm is used) or q (when the ∞-norm is used).

Although MILP problems are in general NP-hard, recently several efficient branch-and-bound MILP solvers [16] have become available. Moreover, there exist several commercial and free solvers for MILP problems such as, e.g., CPLEX, Xpress-MP, GLPK, or lp_solve (see [17], [18] for an overview). In principle, — i.e., when the algorithm is not terminated prematurely due to time or memory limitations, — these algorithms guarantee to find the global optimum. This global optimization feature is not present in most of the other mixed-integer optimization methods that are usually used for MPC (such as simulated annealing, genetic programming, tabu search, etc.).

IV. Simulation Example

In this section we illustrate the possibilities of the route choice control method with a simulation example. As network we have selected the simple network given in Figure 1 with speed limit control. The parameters are selected as follows: $\kappa = 0.25$, $C_1 = C_2 = 2000$, $l_1 = 4$, $l_2 = 6$, $v_a = 40$, $v_b = 100$, and $T = 1$. This means that both routes have the same capacity, but that route 1 has a lower free-flow travel time because it is shorter. The total demand is $Q_{\text{in}}(d) = 3000$ veh/h for all days. The initial turning rate $\beta(0)$ is 0.4.

As cost function we have taken the 1-norm, as described in Section III-C. We have simulated a period of 20 days,

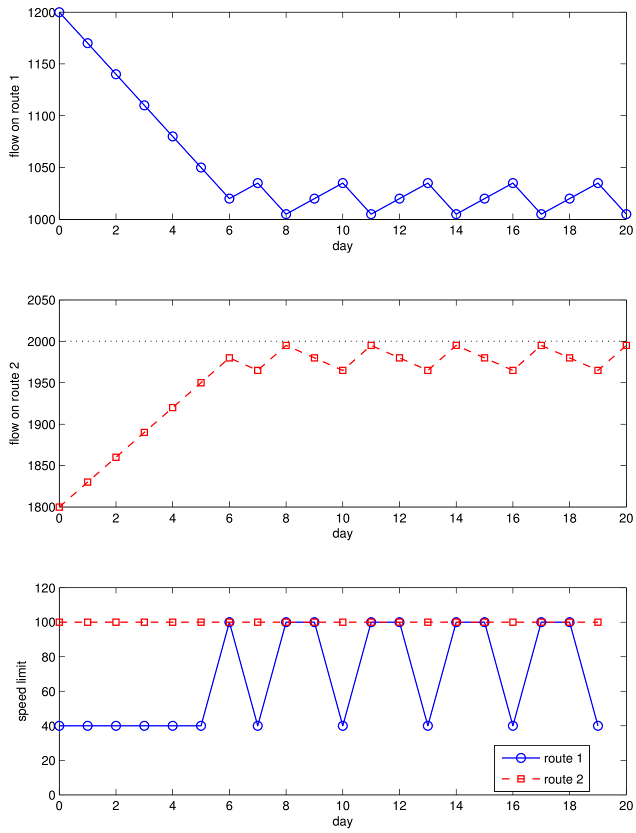

Fig. 3. Flows and speeds limits with an MILP-based MPC controller.

where the prediction and control horizons of the MPC-based controllers are set to 8 days. For the sake of simplicity and to eliminate possible influences of model mismatches, we have used the same model for the simulation and for the prediction by the MPC controller. We have set the desired flow on route 1 to 1000 veh/h, which can, e.g., be useful when the route crosses a residential area. This lower desired flow on route 1 can lead to a large flow on route 2, which will result in congestion on this route. To prevent this congestion, we have put a limit of 2000 veh/h on the flow on route 2.

First, we have used the MILP formulation within the controller. The results are shown in Figure 3. The top plot shows the flow on route 1, which starts at 1200 veh/h, and then decreases until it starts oscillating around 1020 veh/h. These oscillations are caused by the interplay between the bound of 2000 veh/h on the flow on route 2 (see Figure 3 (middle)), and the repeated switching between the maximum and minimum values of the speed limit on route 1 (see Figure 3 (bottom)).

Recall that we have introduced the MILP formulation because it allows for finding the global optimum, and because it is fast. To illustrate these properties we compare the MILP formulation with three other optimization methods: a multi-run genetic algorithm [19], multi-start simulated annealing [20], and brute-force enumeration. As MILP solver we have used CPLEX, implemented through the `cplex` interface function of the Matlab Tomlab toolbox [21]. For

method	cost (veh/h)	CPU time (s)
MILP	850.0	2.23
genetic algorithm	850.0	138.55
simulated annealing	850.0	171.43
enumeration	850.0	296.55

TABLE I

COSTS AND COMPUTATION TIMES FOR DIFFERENT OPTIMIZATION ALGORITHMS.

method	cost (veh/h)
MILP	850.0
genetic algorithm	1709.0
simulated annealing	1080.8

TABLE II

COSTS FOR A MAXIMAL COMPUTATION TIME OF 2.23 S.

the genetic algorithm and simulated annealing method we have used the `ga` and `simulannealbnd` functions of the Matlab Genetic Algorithm and Direct Search Toolbox [22] respectively. For all these optimization functions we have specified that the values of the optimization variables should be integers. For `cplex` the default settings were used; for `ga` and `simulannealbnd` we have implemented the constraint of the flow on route 2 through a penalty term and tuned the settings to get a near-optimal solution in the shortest possible time. The resulting costs J and the required computation times (on a 3 GHz Intel Pentium 4 processor) for the complete closed-loop simulation are given in Table I. Clearly, the MILP approach outperforms the other approaches.

Next, we have re-run the genetic algorithm and the simulated annealing approach giving each of them a maximal CPU time of 2.23 s (i.e., the CPU time required by the MILP algorithm). The results of this experiment are given in Table II. These results once again show that the MILP approach outperforms the other approaches.

V. CONCLUSIONS

We have considered a method based on model predictive control (MPC) to influence day-to-day route choice in traffic networks using existing traffic control measures like outflow control and variable speed limits. In general, this results in nonlinear nonconvex optimization problems. However, we have shown that for a linear cost function the MPC optimization problem can be recast as a mixed-integer linear programming problem for which efficient solvers exist that guaranteedly converge to a global optimum.

The proposed approach has been illustrated by a simulation example with speed limit control where the goal of the controller was to obtain a desired flow on one of the routes. The developed MILP algorithm has been compared with some other available optimization approaches. For the case study the MILP algorithm has shown to be the fastest, and in addition it always returns the globally optimal solution.

ACKNOWLEDGMENTS

Research supported by the NWO-CONNEKT project "Advanced Multi-Agent Control and Information for Integrated Multi-Class Traffic Networks (AMICI)" (014-34-523), the BSIK projects "Transition to Sustainable Mobility (TRANSUMO)" and "Next Generation Infrastructures (NGI)", the Transport Research Centre Delft, and the Delft Research Center Next Generation Infrastructures.

REFERENCES

[1] H. Haj-Salem and M. Papageorgiou, "Ramp metering impact on urban corridor traffic: Field results," *Transportation Research A*, vol. 29, no. 4, pp. 303–319, 1995.

[2] H. Taale and H. van Zuylen, "Traffic control and route choice: Occurrence of instabilities," in *TRAIL 5th Annual Congress 1999: Five Years Crossroads of Theory and Practice*, vol. 2. TRAIL, 1999, pp. 1–19.

[3] T. Bellemans, B. De Schutter, and B. De Moor, "Anticipative model predictive control for ramp metering in freeway networks," in *Proceedings of the 2003 American Control Conference*, Denver, Colorado, Jun. 2003, pp. 4077–4082.

[4] A. Karimi, A. Hegyi, B. De Schutter, J. Hellendoorn, and F. Middelham, "Integration of dynamic route guidance and freeway ramp metering using model predictive control," in *Proceedings of the 2004 American Control Conference (ACC 2004)*, Boston, MA, USA, 2004, pp. 5533–5538.

[5] Y. Wang and M. Papageorgiou, "A predictive feedback routing control strategy for freeway network traffic," in *Proceedings of the American Control Conference*, Anchorage, Alaska, USA, May 2002, pp. 3606–3611.

[6] E. Camacho and C. Bordons, *Model Predictive Control in the Process Industry*. Berlin, Germany: Springer-Verlag, 1995.

[7] J. Maciejowski, *Predictive Control with Constraints*. Harlow, England: Prentice Hall, 2002.

[8] M. van den Berg, A. Hegyi, B. De Schutter, and J. Hellendoorn, "Influencing long-term route choice by traffic control measures — A model study," in *Proceedings of the 17th IFAC World Congress*, Seoul, Korea, Jul. 2008.

[9] E. Bogers, F. Viti, and S. Hoogendoorn, "Joint modeling of ATIS, habit and learning impacts on route choice by laboratory simulator experiments," in *Proceedings of the 84th Annual Meeting of the Transportation Research Board*, Washington DC, USA, Jan. 2005.

[10] A. Hegyi, "Model predictive control for integrating traffic control measures," Ph.D. thesis, TRAIL Thesis Series T2004/2, Delft University of Technology, Delft, The Netherlands, Feb. 2004, ISBN 90-5584-053-X, http://www.hegyi.info/thesis/dep_hegyi_20040203.pdf.

[11] T. Bellemans, "Traffic control on motorways," Ph.D. dissertation, Faculty of Applied Sciences, K.U.Leuven, Leuven, Belgium, May 2003.

[12] A. Kotsialos, M. Papageorgiou, M. Mangeas, and H. Haj-Salem, "Coordinated and integrated control of motorway networks via nonlinear optimal control," *Transportation Research C*, vol. 10, no. 1, pp. 65–84, Feb. 2002.

[13] J. Maciejowski, *Predictive Control with Constraints*. Harlow, England: Prentice Hall, 2002.

[14] P. Pardalos and M. Resende, Eds., *Handbook of Applied Optimization*. Oxford, UK: Oxford University Press, 2002.

[15] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407–427, Mar. 1999.

[16] R. Fletcher and S. Leyffer, "Numerical experience with lower bounds for MIQP branch-and-bound," *SIAM Journal on Optimization*, vol. 8, no. 2, pp. 604–616, May 1998.

[17] A. Atamtürk and M. Savelsbergh, "Integer-programming software systems," *Annals of Operations Research*, vol. 140, no. 1, pp. 67–124, Nov. 2005.

[18] J. Linderoth and T. Ralphs, "Noncommercial software for mixed-integer linear programming," *Optimization Online*, Jan. 2005.

[19] L. Davis, Ed., *Handbook of Genetic Algorithms*. New York: Van Nostrand Reinhold, 1991.

[20] R. Eglese, "Simulated annealing: A tool for operations research," *European Journal of Operational Research*, vol. 46, pp. 271–281, 1990.

[21] "http://www.tomlab.biz/," last visited on January 8, 2008.

[22] The MathWorks, *Genetic Algorithm and Direct Search Toolbox 2 — User's Guide*, Natick, Massachusetts, 2007.