Delft University of Technology

Delft Center for Systems and Control

Technical report 08-029

Day-to-Day Route Choice Control in Traffic Networks with Time-Varying Demand Profiles*

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

To cite this work, please refer to the published version:

M. van den Berg, B. De Schutter, A. Hegyi, and H. Hellendoorn, "Day-to-day route choice control in traffic networks with time-varying demand profiles," *Proceedings of the European Control Conference 2009*, Budapest, Hungary, pp. 1776–1781, Aug. 2009.

Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 CD Delft
The Netherlands
phone: +31-15-278.24.73 (secretary)
URL: https://www.dcsc.tudelft.nl

* This report can also be downloaded via https://dspub.eu/08-029

Day-to-Day Route Choice Control in Traffic Networks with Time-Varying Demand Profiles

Monique van den Berg, Bart De Schutter, Andreas Hegyi, and Hans Hellendoorn

Abstract— **We develop a day-to-day route choice control method that is based on model predictive control (MPC). To influence the route choice of drivers we propose to use traffic control measures like variable speed limits or outflow control. In previous papers we have developed MPC for route choice control in the case of a constant demand. In this paper we consider the case of a time-varying demand. The resulting MPC optimization problem is in general nonlinear and nonconvex. However, in the case of outflow control and for a linear or a piecewise affine cost function it is possible to approximate the problem and to recast it as a mixed integer linear programming (MILP) problem, for which efficient branch-and-bound solvers are available. The solution of the MILP problem can then be used as a good initial starting point for a nonlinear optimization method for the original MPC optimization problem. We also illustrate the proposed approach for a simple simulation example involving outflow control.**

I. Introduction

When an origin and destination are connected via multiple routes, drivers have to make a route choice. Drivers make their route choice based on, e.g., the road type, the surroundings, and most importantly the expected travel times. The route choice of all drivers together leads to a "traffic assignment", which describes how vehicles are divided over the various routes in the traffic network. The traffic assignment that is obtained in this way is the most desired assignment for the drivers. It can however result in large flows in residential areas, or near nature reserves, which is undesired from a social or environmental point of view. Therefore, road administrators want to influence the traffic assignment. In this paper we use traffic control measures to indirectly influence the day-to-day route choice by changing the travel time, which is argued for in [1], [2], and where we assume that the experienced travel time on one day influences the route choice of the next day, as proposed in [3].

As control method we use model predictive control (MPC) [4], [5]. This is a model-based control approach that uses a prediction model in combination with an optimization algorithm to determine optimal settings for the traffic control measures. The optimal settings are then applied using a rolling horizon framework. Earlier work that uses MPC for traffic management on freeways via ramp metering and dynamic speed limits is described in [6], [7], [8], which focused on within-day traffic control. We consider a different time-scale and focus on *day-to-day* route choice control, i.e., the control measures that are applied on one day will influence the route choice of the next day. In our framework two control measures can be selected to influence the route choice of the drivers: outflow control and speed limit control. Outflow control can be done via, e.g., traffic signals or ramp metering installations. In this way the traffic flow that can leave a road or a freeway link can be limited and controlled. Speed limit control uses dynamic speed limits that are displayed using variable message signs. As a result the free-flow travel time can be influenced and controlled.

We first formulate the route choice MPC problem as a nonlinear, nonconvex optimization problem. Such problems may have multiple local minima and in general they are NP-complete and hard to solve. Therefore, we also introduce an approximation that results in a mixed integer linear programming (MILP) problem, for which fast solvers are available [9]. The solution of this MILP problem can then used as a good initial starting point for a nonlinear optimization algorithm for the original MPC problem.

This research is a continuation of the work described in [10], [11] where we have considered a constant demand. In this paper, we extend these results to time-varying demand profiles. Such time-varying demand profiles can be determined using historical data since often the same demand patterns occur every day with some variations depending on the type of day (week day or weekend), the weather, and the season. Since we allow a different demand profile for each day, such variations can be taken into account in our approach. In particular, we consider piecewise constant demand profiles, which allows for a good representation/approximation of reality, while preserving the properties required to approximate the problem by an MILP problem.

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 approximate the problem and recast it as an MILP problem. Next, the proposed control approach is applied to a simulation example in Section IV. Section V concludes the paper.

II. Route choice control

In this section we extend the day-to-day route choice model of [10], [11] to the case of time-varying demand profiles, and we present the corresponding MPC-based route choice control approach.

M. van den Berg, B. De Schutter, and J. Hellendoorn are with the Delft Center for Systems and Control of Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands, {monique.vandenberg, j.hellendoorn}@tudelft.nl, b@deschutter.info

B. De Schutter is also with the Marine & Transport Technology department of Delft University of Technology

A. Hegyi is with the Department of Transport & Planning of Delft University of Technology, P.O. Box 5048, 2600 GA Delft, The Netherlands, a.hegyi@tudelft.nl

A. Route choice model

To illustrate our approach we use the simple two-route network given in Figure 1. This network consists of one origin and one destination that are connected via two routes.

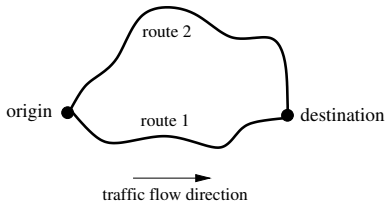

Fig. 1. Network with two routes

Network model: Each route $r \in \{1,2\}$ in the network 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). The speed limit $v_r(d)$ (km/h) gives the maximum speed that is allowed on route r at day d. This speed limit 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. This outflow limit is also bounded between a minimum value $Q_r^{\min}$ (veh/h) and a maximum value $Q_r^{\max}$ (veh/h).

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 $D(d,\cdot)$ on day d is a piecewise constant function during $[0,T]$. More specifically, we have

$$D(d,t) = D_i(d) \quad \text{for } t \in [t_i, t_{i+1})$$

for $i = 0,\ldots,n-1$ where $t_0 = 0$, $t_n = T$, and $t_i < t_{i+1}$ for $i = 0,\ldots,n-1$ (see Figure 2).

The demand $D(d,\cdot)$ is distributed over the two routes according to the turning rate $\beta(d)$, which gives the percentage of the vehicles that select route 1.

Travel time model: The "free-flow" travel time describes the time a vehicle needs to travel a route when there is no delay due to congestion. More specifically, the free-flow travel time at day d along route r is given by: $\tau_r^{\text{free}}(d) = \frac{l_r}{v_r(d)}$.

The model of [10] for the average 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}:

$$\tau_r(d) = \tau_r^{\text{queue}}(d) + \tau_r^{\text{free}}(d) \ . \tag{1}$$

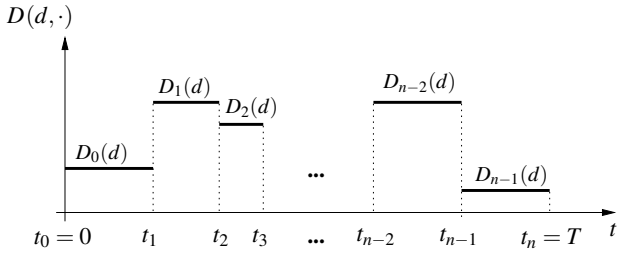

Fig. 2. Time-varying demand profile $D(d,\cdot)$ on day d.

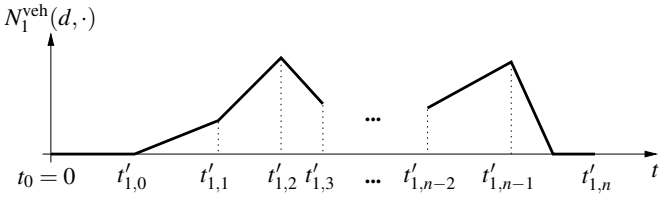

Fig. 3. Evolution of the queue length $N_1^{\text{veh}}(d,\cdot)$ on route 1 and day d during the period $[0, T + \tau_1^{\text{free}}(d)]$. For the sake of simplicity of notation we have not indicated the argument d for the time instants $t'_{1,i}(d)$ in the figure.

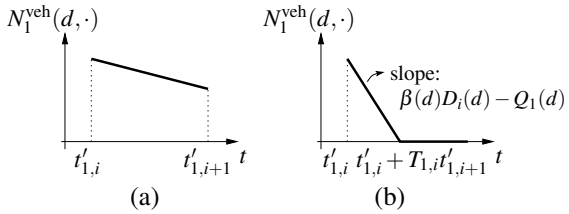

Fig. 4. Two possible cases for the evolution of the queue length $N_1^{\text{veh}}(d,\cdot)$ on route 1 and day d during the interval $[t'_{1,i}(d), t'_{1,i+1}(d)]$

The time in the queue τ_r^{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 $[0, T + \tau_r^{\text{free}}(d)]$ the queue on route r grows as shown in Figure 3, where $t'_{r,i}(d) = t_i + \tau_r^{\text{free}}(d)$ for $i = 0,\ldots,n$. Here, the term $\tau_r^{\text{free}}(d)$ is due to the fact that vehicles entering the network at time t will reach the queue at time $t + \tau_r^{\text{free}}(d)$. Next, the queue will grow or shrink depending on the net growth of the queue, which for route 1 is given by $\beta(d)D_i(d) - Q_1(d)$ for $t \in [t'_{1,i}(d), t'_{1,i+1}(d))$ and $i = 0,\ldots,n-1$. For route 2 the net growth is equal to $(1-\beta(d))D_i(d) - Q_2(d)$ for $t \in [t'_{2,i}(d), t'_{2,i+1}(d)]$ and $i = 0,\ldots,n-1$.

Let us first consider route 1. If we denote the number of vehicles in the queue on route 1 on day d and at time t by $N_1^{\text{veh}}(d,t)$, we have

$$N_1^{\text{veh}}(d, t'_{1,0}(d)) = 0 \tag{2}$$

$$\begin{aligned} N_1^{\text{veh}}(d, t'_{1,i+1}(d)) &= \max\big(0, N_1^{\text{veh}}(d, t'_{1,i}(d)) + \\ &\qquad (\beta(d)D_i(d) - Q_1(d))(t'_{1,i+1}(d) - t'_{1,i}(d))\big) \\ &= \max\big(0, N_1^{\text{veh}}(d, t'_{1,i}(d)) + \\ &\qquad (\beta(d)D_i(d) - Q_1(d))(t_{i+1} - t_i)\big) \ . \end{aligned} \tag{3}$$

In order to compute the average time the vehicles spend in the queue on route 1, we first compute the total area under the N_1^{veh} curve of Figure 3. If we denote the area under the N_1^{veh} curve between $t'_{1,i}(d)$ and $t'_{1,i+1}(d)$ by $A_{1,i}(d)$, there are two possible cases (see Figure 4):

- If $N_1^{\text{veh}}(d, t'_{1,i+1}(d)) > 0$ then we have

$$A_{1,i}(d) = \frac{1}{2}\Big(N_1^{\text{veh}}(d, t'_{1,i}(d)) + N_1^{\text{veh}}(d, t'_{1,i+1}(d))\Big)(t_{i+1} - t_i) \ . \tag{4}$$

- If $N_1^{\text{veh}}(d, t'_{1,i+1}(d)) = 0$ then the queue length already becomes 0 at some time $t'_{1,i}(d) + T_{1,i}(d)$ with $T_{1,i}(d) = N_1^{\text{veh}}(d, t'_{1,i}(d))/(Q_1(d) - \beta(d)D_i(d))$ if

$Q_1(d) \neq \beta(d)D_i(d)$, and $T_{1,i}(d) = 0$ if $Q_1(d) = \beta(d)D_i(d)$, since in the latter case the N_1^{veh} curve is horizontal and $N_1^{\text{veh}}(d, t'_{1,i+1}(d)) = N_1^{\text{veh}}(d, t'_{1,i}(d)) = 0$. Hence,

$$A_{1,i}(d) = \frac{1}{2} N_1^{\text{veh}}(d, t'_{1,i}(d)) T_{1,i}(d) = \frac{(N_1^{\text{veh}}(d, t'_{1,i}(d)))^2}{Q_1(d) - \beta(d)D_i(d)}$$

if $Q_1(d) \neq \beta(d)D_i(d)$, and $A_{1,i}(d) = 0$ if $Q_1(d) = \beta(d)D_i(d)$.

The total area under the N_1^{veh} curve is then equal to $A_1(d) = \sum_{i=0}^{n-1} A_{1,i}(d)$. So the average time that the vehicles spend in the queue at the end of route 1 is given by:

$$\tau_1^{\text{queue}}(d) = \frac{A_1(d)}{T} \quad .$$

Similar formulas (with $1 - \beta(d)$ instead of $\beta(d)$) can be written down for $\tau_2^{\text{queue}}(d)$.

Route choice model: Route choice models describe the route choice of drivers at locations where a route must be selected. The model of [10] updates the turning rates for the next day $d+1$ based on the difference in travel times on the current day d between the two routes, while also taking into account that the turning rates are bounded between 0 and 1:

$$\beta(d+1) = \min\Big(1, \max\big(0, \beta(d) + \kappa(\tau_2(d) - \tau_1(d))\big)\Big) . \quad (5)$$

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

In MPC for route choice control the goal is to determine the control inputs c that optimize some 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 $d+N_\text{p}$. Of the resulting control signal again only the first sample is applied, etc.

The above MPC problem results in nonlinear nonconvex real-valued optimization problems that can be solved using multi-start local search methods (like SQP, pattern search, etc. [12]) or (semi-)global optimization methods (like genetic algorithms or simulated annealing [12]). Note that these approaches in principle only yield a suboptimal solution since — in particular for larger networks or longer control horizons — it is in practice often not tractable to find the global optimum of the optimization problems that arise in MPC for route choice control.

In [11] we have addressed this problem by considering a limited set of discrete values for the control inputs and by considering linear cost functions, which allowed us to transform the MPC optimization problem into a mixed integer linear programming (MILP) problem, for which efficient solvers exist that guarantee to find the global optimum [9]. Unfortunately, this approach cannot be applied directly any longer in the case of time-varying demands. However, by making an approximation in the way the average time in the queue is computed, we can still transform the problem into an MILP problem, as will be explained in the next section. The solution of this MILP problem can then be used as a good initial starting point for a nonlinear optimization algorithm applied to the original MPC optimization problem.

III. Approximate solution using mixed integer linear programming

In this section we show that for linear or piecewise affine cost functions the MPC route choice optimization problem can be approximated by an MILP problem. In particular, we will consider the case of outflow control only (so there is no speed control).

A. Rules for creating mixed integer linear inequalities

To approximate and to formulate the route choice control problem described above as an MILP problem, we will have to remove the nonlinearities from the model. This is done by introducing an approximation in the way the area is computed in the case of Figure 4(b) and by recasting the nonlinear equations into linear ones. To perform the latter transformations we use the following equivalences [13], where δ, δ_1, and δ_2 represent binary-valued scalar variables, y a real-valued scalar variable, and f a real-valued scalar function defined on a bounded set X with upper and lower bounds U_f and L_f for the function values:

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

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

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

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

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

P3: $\delta = \delta_1 \delta_2$ is equivalent to

$$\begin{cases} -\delta_1 + \delta \leqslant 0 \\ -\delta_2 + \delta \leqslant 0 \\ \delta_1 + \delta_2 - \delta \leqslant 1 \end{cases} .$$

B. Transformation of the model equations

For the sake of simplicity we assume that the outflow limits can only have two non-zero values $Q_{r,a}$ and $Q_{r,b}$ (note however that an extension to more than two values is straightforward) and we consider control for route 1 only.

Later on we will see that in the model equations the factor $\frac{1}{Q_r(d)}$ will appear. This factor can be represented by introducing binary variable as follows. If we define

$$\Delta_r = \frac{1}{Q_{r,b}} - \frac{1}{Q_{r,a}} \ ,$$

then we can select $Q_{r,a}$ or $Q_{r,b}$ on route r for day d by introducing a binary variable $\delta_r(d)$ and setting

$$\frac{1}{Q_r(d)} = \frac{1}{Q_{r,a}} + \Delta_r \delta_r(d) \ . \tag{6}$$

Let us now first rewrite the equations for the evolution of N_1^{veh} (for N_2^{veh} a similar reasoning holds). If we define

$$m_1(d,i+1) = \frac{N_1^{\text{veh}}(d,t'_{1,i+1}(d))}{Q_1(d)} \ ,$$

then it follows from (3) that

$$\begin{aligned} m_1(d,i+1) &= \max\left(0, \frac{N_1^{\text{veh}}(d,t'_{1,i}(d))}{Q_1(d)} + \right. \\ &\qquad \left. \left(\frac{\beta(d)D_i(d)}{Q_1(d)} - 1\right)(t_{i+1}-t_i)\right) \\ &= \max\left(0, m_1(d,i) + \left(\frac{\beta(d)D_i(d)}{Q_1(d)} - 1\right)(t_{i+1}-t_i)\right) \end{aligned} \tag{7}$$

with $m_1(d,0) = 0$ (cf. (2)).

We will now transform (7) into mixed-integer linear equations. If we substitute (6) into (7) we get an expression of the form[1]

$$\begin{aligned} m_1(d,i+1) = \max\big(0, m_1(d,i) + a_{1,i}\beta(d) + \\ a_{2,i}\delta_1(d)\beta(d) + a_{3,i}\big) \end{aligned} \tag{8}$$

with $a_{1,i} = \frac{D_i(d)}{Q_{1,a}}(t_{i+1}-t_i)$, $a_{2,i} = D_i(d)\Delta_1(t_{i+1}-t_i)$, and $a_{3,i} = -t_{i+1}+t_i$. By introducing an extra variable $y_1(d) = \delta_1(d)\beta(d)$ and using Property **P2** with $f(x) = \beta(d)$, $L_f = 0$, and $U_f = 1$, (8) can be transformed into a system of linear inequalities together with the nonlinear equation

$$m_1(d,i+1) = \max\big(0, m_1(d,i) + a_{1,i}\beta(d) + a_{2,i}y_1(d) + a_{3,i}\big) .$$

Now we define binary variables $\delta_{3,i}(d)$ such that $\delta_{3,i}(d) = 1$ if and only if $m_1(d,i) + a_{1,i}\beta(d) + a_{2,i}y_1(d) + a_{3,i} \geqslant 0$. Using Property **P1** this equivalence can be recast as a system of linear inequalities. Then we get

$$m_1(d,i+1) = \delta_{3,i}(d)(m_1(d,i) + a_{1,i}\beta(d) + a_{2,i}y_1(d) + a_{3,i}) .$$

By introducing extra variables $y_{2,i}(d) = \delta_{3,i}(d)m_1(d,i)$, $y_{3,i}(d) = \delta_{3,i}(d)\beta(d)$, and $y_{4,i}(d) = \delta_{3,i}(d)y_1(d)$, and using Property **P2** we obtain again a system of linear inequalities together with the equation

$$m_1(d,i+1) = y_{2,i}(d) + a_{1,i}y_{3,i}(d) + a_{2,i}y_{4,i}(d) + a_{3,i}\delta_{3,i}(d) \ ,$$

which is a linear equation.

[1]For the sake of simplicity of notation we will not explicitly include the argument d in the parameters $a_{1,i}$, $a_{2,i}$, etc.

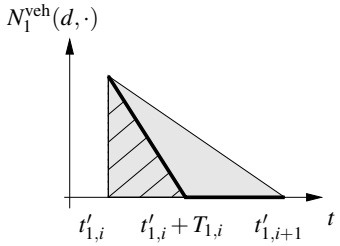

Fig. 5. The area of the hashed triangle will be approximated by the area of the shaded triangle.

Now we make the following approximation (see Figure 5): We always take expression (4) for $A_{1,i}(d)$, resulting in

$$\begin{aligned} &\tau_1^{\text{queue}}(d) \\ &= \frac{1}{2}\frac{\sum_{i=0}^{n-1}\big(N_1^{\text{veh}}(d,t'_{1,i}(d)) + N_1^{\text{veh}}(d,t'_{1,i+1}(d))\big)(t_{i+1}-t_i)}{Q_1(d)T} \\ &= \frac{1}{2T}\sum_{i=0}^{n-1}\big(m_1(d,i) + m_1(d,i+1)\big)(t_{i+1}-t_i) \ . \end{aligned}$$

Since this expression is linear in m_1, it follows from (1) that $\tau_1(d)$ is also linear in m_1. Similarly, $\tau_2(d)$ can be written as a linear expression in m_2 by introducing the additional real-valued auxiliary variables $y_5(d)$, $y_{6,i}(d)$, $y_{7,i}(d)$, $y_{8,i}(d)$, and binary auxiliary variables $\delta_{4,i}(d)$ and $\delta_{5,i}(d) = \delta_{4,i}(d)\delta_2(d)$ (cf. Property **P3**) for $i = 0,\ldots,n-1$).

Finally, we consider (5). If we define

$$\gamma(d) = \beta(d) + \kappa(\tau_2(d) - \tau_1(d)) \tag{9}$$

$$\eta(d) = \max(0, \gamma(d)) \ , \tag{10}$$

then we have

$$\beta(d+1) = \min(\eta(d), 1) \ . \tag{11}$$

Note that (9) is already linear in β, τ_1, and τ_2. Now consider (10). If we define the binary variable $\delta_6(d)$ such that $\delta_6(d) = 1$ 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 $\eta(d) = \delta_6(d)\gamma(d)$, which can in its turn also be expressed as a system of linear inequalities using Property **P2**.

Consider (11) and define the binary variable $\delta_7(d)$ such that $\delta_7(d) = 1$ if and only if $\eta(d) \leq 1$. This equivalence can be recast as a system of linear inequalities via Property **P2**. It is easy to verify that now we have

$$\beta(d+1) = \min(\eta(d), 1) = \delta_7(d)\eta(d) + 1 - \delta_7(d) \ ,$$

which after introducing the auxiliary variable $z(d) = \delta_7(d)\eta(d)$ (this equivalence can also be recast as a system of linear inequalities via Property **P1**), results in the linear equation

$$\beta(d+1) = z(d) + 1 - \delta_7(d) \ .$$

If we now collect all variables for day d in one vector $w(d)$, 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

In order to be able to transform the approximate 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 desired travel times on each of the routes, 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).

Define $D_{\text{tot}}(d+j) = \sum_{i=0}^{n-1} \frac{t_{i+1}-t_i}{T} D_i(d+j)$. The MPC cost function for a minimum flow on route 2 is given by:

$$J(d) = \sum_{j=1}^{N_\text{p}} (1-\beta(d+j)) D_{\text{tot}}(d+j) \ ,$$

which is indeed linear.

Let $\tau_1^{\text{desired}}(d+j)$ and $\tau_2^{\text{desired}}(d+j)$ denote the desired flow on respectively route 1 and route 2 at day $d+j$. The problem of reaching desired travel times on each of the routes is then given by

$$\min \sum_{j=1}^{N_\text{p}} w_1 |\tau_1(d+j) - \tau_1^{\text{desired}}(d+j)| + w_2 |\tau_2(d+j) - \tau_2^{\text{desired}}(d+j)|$$

with $w_1, w_2 > 0$. This cost function is piecewise affine, but it can be rewritten as

$$\begin{aligned} \min \sum_{j=1}^{N_\text{p}} & w_1 \phi_1(d+j) + w_2 \phi_2(d+j) \\ \text{s.t.} \quad & \\ & \phi_r(d+j) \geqslant \tau_r(d+j) - \tau_r^{\text{desired}}(d+j) \\ & \phi_r(d+j) \geqslant -\tau_r(d+j) + \tau_r^{\text{desired}}(d+j) \\ & \text{for } j = 1, \ldots, N_\text{p} \text{ and for } r = 1,2 . \end{aligned}$$

which is a linear programming problem. It is easy to verify that for the optimal solution of this problem, we have $\phi_r^*(d+j) = \max\left(\tau_r^*(d+j) - \tau_r^{\text{desired}}(d+j), -\tau_r^*(d+j) + \tau_r^{\text{desired}}(d+j)\right) = |\tau_r^*(d+j) - \tau_r^{\text{desired}}(d+j)|$ for all j and for $r = 1,2$.

Using a reasoning similar to the one used above it is easy to show that the MPC cost function corresponding to reaching a piecewise constant desired flow profile on route 2 (with the same time intervals $[t_i, t_{i+1})$ as those of the demand profile D) using the 1-norm or the ∞-norm also results in a linear programming problem (see also [11]). Using Properties **P1** and **P2** one can also recast the outflow control problem with a (general) piecewise affine cost function as an MILP problem.

D. Constraints

Because minimizing, e.g., the flow on route 2 will result in a higher flow and thus a longer travel time on route 1, it might be useful to add a constraint on the travel time on route 1:

$$\tau_1(d+j) \leqslant \tau_1^{\max}(d+j) \quad \text{for } j = 0, \ldots, N_\text{p} - 1 \ , \qquad (12)$$

where $\tau_1^{\max}(d+j)$ denotes the maximal travel time on route 1 on day $d+j$. Note that (12) is a linear constraint.

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

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

for $i = 0, \ldots, n-1$ and for $j = 1, \ldots, N_\text{p}$, 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. Overall mixed integer linear programming problem

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 $w(d), w(d+1), \ldots, w(d+N_\text{p}-1)$, $\beta(d+N_\text{p})$, $\phi_1(d+1), \ldots, \phi_1(d+N_\text{p})$, and $\phi_2(d+1), \ldots, \phi_2(d+N_\text{p})$.

Recently several efficient branch-and-bound MILP methods [9] 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 [14], [15] 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 of the MILP problem. This optimum could then be used as initial starting point for the original route choice MPC optimization problem.

IV. Simulation example

We now illustrate the possibilities of the MPC approach for day-to-day route choice by an example. Consider the simple network with two routes given in Figure 1 and with the following parameters: κ = 0.5, l_1 = 3 km, l_2 = 6 km, $v_1(d) = v_2(d) = 100$ km/h. This means that in the absence of control route 1 is preferred by the drivers, because it has the lowest free-flow travel time. Outflow control is applied on both routes, with the maximum allowed flow limit equal to 5000 veh/h, and a minimum outflow limit of 1000 veh/h. The prediction horizon is 5 days, and we simulate a period of 40 days. On each day 100 minutes are simulated. The demand pattern is equal for all days, and given in Table I. The initial turning rate $\beta(0)$ is 0.6.

TABLE I

Piecewise constant demand pattern used in the example

time interval (min)	0–20	20–40	40–60	60–80	80–100
demand (veh/h)	2000	6000	8000	5000	0

The goal of the controller is to reach a desired flow of 2500 veh/h on route 1. For this particular scenario, the optimal turning rate in steady state can be computed analytically. Indeed, minimizing $\sum_{i=0}^{4} \frac{t_{i+1}-t_i}{T} |\beta D_i - 2500|$ yields an optimal steady-state turning rate $\beta^* = 0.4167$.

We have simulated the traffic in the network with different optimization strategies for the controller. Table II gives an overview of the results. The first simulation is done without control, i.e., when the outflow is not limited and equal to its

maximal value of 5000 veh/h. Next, we have compared three approaches which all use SQP (implemented via the function fmincon of the Matlab Optimization Toolbox), but with a different number of random initial points. When the number of randomly selected initial points increases, the performance increases, but the computation time also becomes larger. The computation time can be reduced by computing an initial point with MILP, which can then be used as initial point for one run of the SQP algorithm. This reduces the computation time, and even improves the performance. To show that the SQP run is really necessary, we also have performed a simulation with MILP only. This simulation runs very fast, but the improvement of the performance is very low. The average turning rates obtained with the different controllers are given in the last column of Table II. When MILP optimization is used as initial point, the value of the turning rate returned by SQP corresponds to the analytically computed optimal value.

V. Conclusions

We have considered a method based on model predictive control (MPC) to steer day-to-day route choice in traffic networks towards an optimal situation using existing traffic control measures like outflow control and variable speed limits. In general, this results in nonlinear nonconvex optimization problems. However, for a linear or a piecewise affine cost function the MPC optimization problem can be approximated and recast as a mixed integer linear programming (MILP) problem, for which efficient solvers exist. The optimal solution of the MILP problem can then be used as initial starting point for the original route choice MPC optimization problem. The proposed approach has been illustrated by means of a simulation example with outflow limit control where the goal of the controller was to obtain a desired flow on one of the routes.

Some topics for future are: extending the proposed approach to more complex networks with multiple origins, destinations, route choice locations, and routes, as well as including additional traffic control measures and assessing the scalability of the proposed approach.

Acknowledgments

Research supported by the NWO-CONNEKT project "Advanced Multi-Agent Control and Information for Integrated Multi-Class Traffic Networks (AMICI)", the BSIK projects "Transition to Sustainable Mobility (TRANSUMO)" and "Next Generation Infrastructures (NGI)", the European STREP project "Hierarchical and distributed model predictive control (HD-MPC)", the European COST Action TU0702, 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] 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.

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

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

[6] A. Hegyi, B. De Schutter, and H. Hellendoorn, "Model predictive control for optimal coordination of ramp metering and variable speed limits," *Transportation Research C*, vol. 13, no. 3, pp. 185–209, June 2005.

[7] T. Bellemans, B. De Schutter, and B. De Moor, "Model predictive control for ramp metering of motorway traffic: A case study," *Control Engineering Practice*, vol. 14, no. 7, pp. 757–767, July 2006.

[8] 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.

[9] 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.

[10] 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, July 2008, pp. 13 052–13 057.

[11] 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," in *Proceedings of the 17th IEEE International Conference on Control Applications*, San Antonio, Texas, Sept. 2008, pp. 299–304.

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

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

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

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

[2] On a 1 GHz AMD Athlon 64x2 Dual Core processor.

TABLE II

COMPUTATION TIME, COST, RELATIVE IMPROVEMENT (W.R.T. THE UNCONTROLLED CASE), AND AVERAGE TURNING RATES FOR DIFFERENT OPTIMIZATION SCENARIOS

control method	comp. time[2](s)	cost (veh/h)	% improvement	av. turning rate
no control	0	383	0	0.728
SQP, 1 init. point	18	223	41.7%	0.407
SQP, 10 init. points	167	223	41.8%	0.412
SQP, 20 init. points	364	223	41.8%	0.412
SQP, MILP init. point	15	221	42.1%	0.417
MILP	6	293	23.6%	0.578