Delft University of Technology

Delft Center for Systems and Control

Technical report 11-030

Model-based traffic and emission control using PWA models – A mixed-logical dynamic approach*

N. Groot, B. De Schutter, S.K. Zegeye, and H. Hellendoorn

If you want to cite this report, please use the following reference instead:
N. Groot, B. De Schutter, S.K. Zegeye, and H. Hellendoorn, "Model-based traffic and emission control using PWA models – A mixed-logical dynamic approach," *Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems (ITSC 2011)*, Washington, DC, pp. 2142–2147, Oct. 2011.

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

Model-Based Traffic and Emission Control Using PWA Models – A Mixed-Logical Dynamic Approach

Noortje Groot, Bart De Schutter, Solomon Kidane Zegeye, and Hans Hellendoorn

***Abstract*— For the purpose of traffic control a piecewise-affine (PWA) approximation of the METANET model is proposed and tested in a model-based predictive control (MPC) framework. This approximation is provided as an alternative to the rather intensive computations when using the original nonlinear nonconvex METANET traffic flow model extended with a model for vehicular emissions and fuel consumption in an MPC context. As a direct PWA-MPC computation turned out to be intractable for on-line applications due to the size of the final, full PWA model that consists of a large number of PWA regions, the PWA model equations were additionally converted into a mixed-logical dynamic (MLD) model. The resulting MLD-MPC problem – written as a mixed-integer linear program (MILP) – can be solved much more efficiently as it does not explicitly state all model equations for each particular region. In a simple case study on a traffic network including a variable speed limit and an un-metered on-ramp while optimizing the total time spent (TTS), we compared the performance of the approximate MLD-MPC approach to that of model predictive traffic control when using the original nonlinear formulation of the METANET model.**

I. INTRODUCTION

In the control of large-scale traffic networks it is important to adopt a modeling framework that is both accurate and that yields a fast solution in order to be able to apply on-line traffic control. When using the METANET traffic flow model [1], [2] complemented by the VT-macro emission and fuel consumption model [3] in combination with model predictive control (MPC) [4], [5] in order to minimize the total time spent (TTS) by traffic in the network as well as to reduce the vehicular emissions, a nonlinear nonconvex optimization problem results. Such a problem can be solved with global or multi-start local optimization (see, e.g., [6]–[8]). However, this approach is subject to computational issues and a global optimum cannot be guaranteed. Hence, the need arises to search for alternative solution approaches like the piecewise-affine (PWA)-MPC method proposed in this paper.

In previous work [9] the METANET model was approximated in a PWA manner, where the focus was on investigating the accuracy of the approximations while using different methods. In the current paper, the focus is on the actual use of the PWA-approximated model in an MPC framework. As it turns out, due to the large number of regions of the full PWA model for the entire traffic network, a direct implementation of PWA-MPC is computationally intractable. Therefore, we propose to convert the individual PWA model equations directly into a mixed-logical dynamic (MLD) formulation of the system, after which MPC can be applied. Consequently, the trade-off between time-efficiency and accuracy of the MLD-MPC approach as compared to nonlinear MPC using the original model can be analyzed.

The remainder of this paper is built up as follows. In Section II the original nonlinear METANET traffic flow model is briefly presented, together with the VT-macro equations needed to incorporate emissions and fuel consumption into the traffic control framework. In Section III the nonlinear model equations are approximated in a PWA manner and subsequently the reformulation as an MLD model and into an MILP program is described in Section IV. Section V includes a description of the MPC approach we propose for traffic control, which is applied in the case study of Section VI. The paper is concluded in Section VII.

N. Groot, B. De Schutter, S.K. Zegeye, and H. Hellendoorn are with the Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands. E-mail: n.b.groot@tudelft.nl

II. THE METANET MODEL

The original METANET model developed by Papageorgiou and Messmer [1], [2] is as follows. Let the traffic network be described by a graph with links representing homogeneous parts of a freeway, separated by nodes representing changes like on-ramps and the merging of lanes. A link is further divided into segments of equal distance. The evolution of traffic flow $q_{m,i}$ in veh/h, density $\rho_{m,i}$ (veh/km/lane) and space-mean speed $v_{m,i}$ (km/h) for segment i of link m for time step k is described by:

$$q_{m,i}(k) = \lambda_m \rho_{m,i}(k) v_{m,i}(k) \tag{1}$$

$$\rho_{m,i}(k+1) = \rho_{m,i}(k) + \frac{T_\mathrm{s}}{L_m \lambda_m}\left[q_{m,i-1}(k) - q_{m,i}(k)\right] \tag{2}$$

$$\begin{aligned} v_{m,i}(k+1) = v_{m,i}(k) &+ \frac{T_\mathrm{s}}{\tau}\left[V[\rho_{m,i}(k)] - v_{m,i}(k)\right] \\ &+ \frac{T_\mathrm{s} v_{m,i}(k)\left[v_{m,i-1}(k) - v_{m,i}(k)\right]}{L_m} \\ &- \frac{T_\mathrm{s}\eta\left[\rho_{m,i+1}(k) - \rho_{m,i}(k)\right]}{\tau L_m\left(\rho_{m,i}(k) + \kappa\right)}, \end{aligned} \tag{3}$$

with λ_m the number of lanes in link m, T_s the simulation time step (in s), L_m the length of the segments of link m (in m) and η (km^2/h), κ (veh/km/lane), and τ (h) model parameters.

The desired speed is represented by

$$\begin{aligned} V[\rho_{m,i}(k)] = \min\Bigg[&v_{\mathrm{free},m} \exp\left[-\frac{1}{a_m}\left(\frac{\rho_{m,i}(k)}{\rho_{\mathrm{cr},m}}\right)^{a_m}\right], \\ &(1+\alpha)v_{\mathrm{control},m,i}(k)\Bigg], \end{aligned} \tag{4}$$

where the second term applies only in case of variable-speed control on segment i of link m, where the speed limit is denoted by the speed control variable $v_{\text{control},m,i}(k)$ (km/h) [10]. Further, $v_{\text{free},m}$ (km/h) denotes the free-flow speed and α is a non-compliance factor. Further, a_m is a model parameter and $\rho_{\text{cr},m}$ (veh/km/lane) denotes the critical density of a link m connected to the given origin.

Mainstream origins and on-ramps are modeled as a queue where w_o (veh) represents the queue length at origin o:

$$w_o(k+1) = w_o(k) + T_\text{s}(d_o(k) - q_o(k)). \tag{5}$$

Here, d_o (veh/h) denotes the traffic demand and q_o (veh/h) the outflow of origin o:

$$q_o(k) = \min\Bigg[d_o(k) + \frac{w_o(k)}{T_\text{s}},\ r_o(k)C_o,\\ C_o\left(\frac{\rho_{\text{jam},m} - \rho_{m,1}(k)}{\rho_{\text{jam},m} - \rho_{\text{cr},m}}\right)\Bigg], \tag{6}$$

for a metered on-ramp with ramp-metering rate $r_o(k) \in [0,1]$. For non-metered on-ramps or mainstream origins the decision variable $r_o(k)$ is set to one. Further, C_o (veh/h) represents the capacity of origin o and $\rho_{\text{jam},m}$ (veh/km/lane) is the maximum density of a link m connected to the given origin.

For the first segment of an outgoing link of each on-ramp, the following speed-drop factor is added to speed equation (3) with δ a model parameter:

$$-\frac{\delta T_\text{s} q_o(k) v_{m,1}(k)}{L_m \lambda_m (\rho_{m,1}(k) + \kappa)}. \tag{7}$$

The METANET model can be further complemented to take into account e.g., merges and drops of lanes and the resulting speed drops, main-stream metering, or it can be adapted to different models for dynamic speed limits (see e.g., [2], [8], [10]).

Additionally, in order to take into account emissions and fuel consumption of the vehicles, we add to the METANET model the equations of the VT-macro model. For more detailed information on this model, the reader is referred to [3]. The VT-macro model estimates traffic emissions and fuel consumption using either the temporal or spatio-temporal accelerations of vehicles. For instance, the spatio-temporal acceleration and number of vehicles subject to it while moving from one segment to the next segment of a link m are given by

$$a_{m,i,i+1}(k) = \frac{v_{m,i+1}(k) - v_{m,i}(k-1)}{T_\text{s}} \tag{8}$$

$$n_{m,i,i+1}(k) = T_\text{s} q_{m,i}(k-1). \tag{9}$$

Similar expressions apply to e.g., on-ramps, off-ramps, junctions, etc. Slightly different, temporal accelerations and velocities refer to the values of those variables *within* the same segment.

The vehicular emissions and fuel consumption become apparent in the cost function when minimizing the following expression of total emissions or fuel consumption (TEFC) at time step k:

$$J_{\gamma,\text{TEFC}}(k) = \sum_{\ell \in L_\text{all}} n_\ell(k) \exp\left(\breve{v}_\ell^\top(k) P_\gamma \breve{a}_\ell(k)\right), \tag{10}$$

with the speed and acceleration vectors $\breve{v}^\top(k) = [1 \quad v(k) \quad v^2(k) \quad v^3(k)]^\text{T}$ and $\breve{a}^\top(k) = [1 \quad a(k) \quad a^2(k) \quad a^3(k)]^\text{T}$, and with L_all the set of indices of all triples (a_ℓ, n_ℓ, v_ℓ) of spatio-temporal or temporal accelerations and the corresponding numbers of vehicles and speeds. Moreover, P_γ denotes the model parameter for $\gamma \in \Gamma = \{$CO emission, HC emission, NO_x emission, fuel consumption$\}$. The values of the parameter matrices P_γ can be found in [11].

III. PWA Approximation

A PWA function consists of multiple affine functions defined on convex polyhedra and can be written as follows (e.g., [12]):

$$f(x) = a_i^T x + b_i \quad \text{if } x \in \Omega_i, \tag{11}$$

where $x \in \mathbb{R}^n$ is the independent variable, $a_i \in \mathbb{R}^n$ and $b_i \in \mathbb{R}$ are the (constant) parameters for each of the N convex polyhedra Ω_i in the x-space, such that $\cup_i \Omega_i = \mathbb{R}^n$ and $\text{int}(\Omega_i) \cap \text{int}(\Omega_j) = \emptyset$ for all i, j with $i \neq j$.

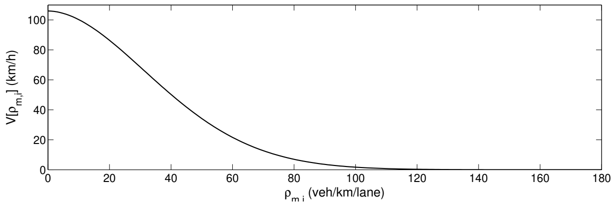
In the remainder of this section the nonlinear METANET equations will be adapted to this PWA form conform one of the few possible approximation methods shortly discussed. Here it should be noted that there is a large difference in complexity between the approximation of single and bivariate functions or two versus multidimensional cases, both of which occur in the METANET setting we consider in this paper. For more details on the approximation methods used and the quality of the approximations made, please refer to [9].

It should also be noted that a reformulation of the nonlinear functions in the METANET model can only be done as an approximation, meaning that the optimal solution yielded by solving the MPC problem will inevitably differ from the original optimum. However, in return we yield a PWA model description that is faster to solve and which can be used as a good initial starting point for control when using the nonlinear model. Here it is important to keep the number of regions Ω_i small as each region will result in additional binary variables that again increase the (time) complexity of the problem.

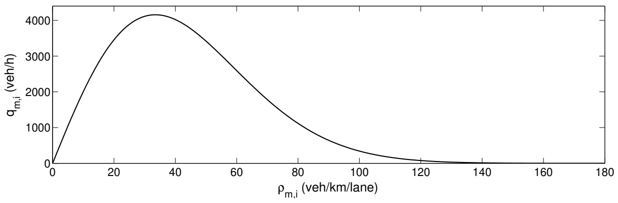
A. Flow equation (1)

Starting with the bivariate flow equation (1), an approximation approach is by hybrid or PWA identification:

> **Piecewise-affine identification**. For the purpose of our paper it suffices to apply the piecewise or hybrid identification approach. We adopt the most accurate method for bivariate identification, i.e., Multicategory Robust Linear Programming (MRLP) [13]. This method will first create local data sets after which a clustering algorithm creates local affine models by classifying



(a) Speed as a function of density



(b) Flow as a function of density

Fig. 1. Fundamental diagram of traffic flow

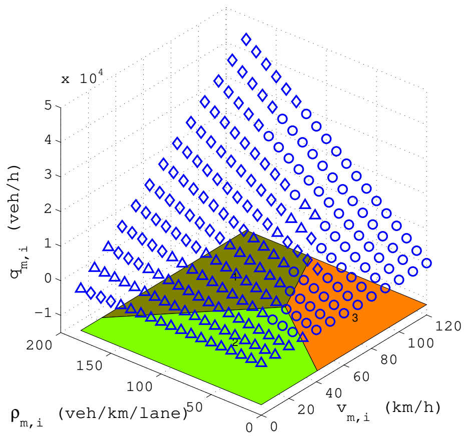


Fig. 2. Approximation of flow equation (1) by hybrid identification using the Hybrid Identification Toolbox [15]

the data points. Similar models are again grouped into clusters, depending on the number of regions required [14].

In the approximation of (1) it is further important to take into account the shape of the fundamental diagram shown in Fig. 1(a) and (b). To be more precise, in order to increase the accuracy of the approximation while keeping the set of auxiliary variables small one can put additional weight on data points where a small error is important. Looking at the shape of the fundamental diagram, it can be inferred that a situation of close-to maximum density and speed simultaneously is not likely to occur in real-life. m Therefore, the focus should be on a good match in the area around the functions determined in the fundamental diagram. The final PWA approximation can be seen in Fig. 2.

B. Speed equations (3)–(4)

Within the speed equations several issues should be dealt with, i.e.:

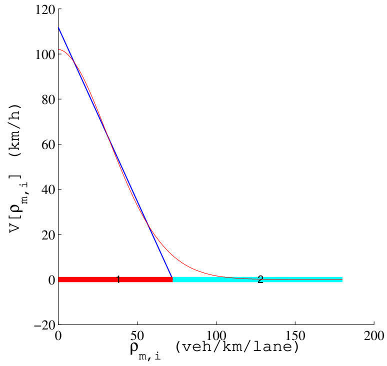


Fig. 3. Fundamental diagram – PWA in 2 equations

- the density variable arising in an exponential factor in the first term of (4),
- multiplication of speed variables,
- division of density variables by another density,
- subtraction of the term written separately in (7).

1) Density arising in exponential equation (4): Taking the parameter settings $a_m = 1.867, v_{\text{free}} = 102\,\text{km/h}$, and $\rho_{\text{crit}} = 33.5\,\text{veh/km/lane}$ [2] the curve of Fig. 1(a) is obtained and subsequently approximated as shown in Fig. 3. Approximation of this single-variate function is easier than the previous case; instead of using the PWA identification algorithms provided by the HIT-toolbox as explained in the previous section, a more accurate least-squares optimization can be run:

Least-squares optimization. For single-variate non-linear functions, one could first select the number of regions (intervals in this case) of the PWA function and next optimize the interval lengths and the parameters of the affine functions, minimizing the squared difference between the (weighted) original function and the approximation.

E.g., the following PWA problem may be solved in a least-squares manner – here given for a approximation of a function f defined on an interval $[x_{\min}, x_{\max}]$ by a continuous PWA function $f_{\text{PWA}}(x)$ with 3 intervals:

$$\min_{\alpha,\beta,\gamma,\delta,\epsilon,\zeta} \int_{x_{\min}}^{x_{\max}} \left(f_{\text{PWA}}(x) - f(x)\right)^2 dx$$

s.t.

$$f_{\text{PWA}}(x) = \begin{cases} \gamma + \dfrac{x - x_{\min}}{\alpha - x_{\min}}(\delta - \gamma) & \text{for } x_{\min} \le x < \alpha \\ \delta + \dfrac{x-\alpha}{\beta-\alpha}(\epsilon - \delta) & \text{for } \alpha \le x < \beta \\ \epsilon + \frac{x-\beta}{x_{\max}-\beta}(\zeta - \epsilon) & \text{for } \beta \le x \le x_{\max}. \end{cases}$$

Least-squares optimization can be solved using e.g., a Gauss-Newton or Levenberg-Marquardt approach [16].

2) Multiplication of speed variables – $v_{m,i}(k)[v_{m,i-1}(k) - v_{m,i}(k)]$: Rather than choosing for a PWA approximation, here we simply keep the first velocity variable $v_{m,i-1}(k)$ either constant at a value determined by historical data (in general) or equal to the currently measured value for predictions. Alternatively, a more exact approximation could also be obtained using a similar method to that of bivariate equation (1).

3) Division by density – $\frac{\rho_{m,i+1}(k)-\rho_{m,i}(k)}{\rho_{m,i}(k)+\kappa}$: As in the previous step, the density term in the denominator is kept constant at a historically-based value or taken according the last measurements.

4) Subtraction of the term written separately in (7): Final adaptations are made to this speed-drop term by adopting the same constant approach of substituting the density in the denominator, combined with the substitution of $q_0 \cdot v_{m,1}$ as in the PWA approximation of the flow equation (1).

Finally, note that the on-ramp flow equation (6) is already PWA, which means that together with the originally linear equations (2), (5), (8), and (9) we now have a system of only linear and PWA model equations. However, since the term $J_{\gamma,\mathrm{TEFC}}(k)$ from (10) causes the optimization objective to become a nonlinear nonconvex function, another PWA approximation should be made. First, in order to reduce the number of additional regions and therefore the computational complexity, $n_\ell(k)$ can be taken constant as explained in [17]. Next, the exponential function term in (10) can be approximation by a PWA function, where we refer to the approximation methods for the bivariate flow equation (1).

Now, in order to obtain a directly implementable optimization problem, some further adaptations using auxiliary binary variables are needed as will be explained next.

IV. From PWA to MLD

A. Using a full PWA model

In order to be able to apply MPC to the PWA model, a logical next step would be to combine the individual model equations approximated in the previous section and to rewrite them into one coherent PWA description of the entire traffic network, i.e., in the following structure for the PWA system in state space notation:

$$\begin{aligned} x(k+1) &= A_i x(k) + B_i u(k) + f_i \\ y(k) &= C_i x(k) + D_i u(k) + g_i \\ &\left[x^{\mathrm{T}}(k)\ u^{\mathrm{T}}(k)\right]^{\mathrm{T}} \in \Omega_i, \end{aligned}$$

where $x(k) \in \mathbb{R}^{\mathrm{n}_x}, u(k) \in \mathbb{R}^{\mathrm{n}_u}$, and $y \in \mathbb{R}^{\mathrm{n}_y}$ denote respectively the state, input and output vector, an where Ω_i is a convex polyhedron. For each region Ω_i, A_i, B_i, C_i, D_i, and f_i, g_i represent constant system matrices and vectors of appropriate sizes.

To obtain the above PWA system description, the individual PWA model equations should be combined for each link and segment of the given traffic network, yielding a cross-product of the PWA regions and therefore an exponential growth of the model. Due to the large total number of regions Ω_i this results in, the composition of the full PWA traffic model is already inefficient. Moreover, when using MPC as explained in the next section, this PWA model has to be evaluated over several future time steps, which causes this PWA-MPC approach for the METANET model to be computationally intractable already for a small network of only a few segments.

B. A tractable approach using an MLD model

In order to do be able to efficiently solve the MPC problem based on a PWA system description with a large number of regions, we do not compose the fully integrated PWA model, yet we propose to make a conversion of the individual model equations to the following MLD description (see e.g., [18]):

$$\begin{aligned} x(k+1) &= Ax(k) + B_1 u(k) + B_2 \delta(k) + B_3 z(k) + f \\ y(k) &= Cx(k) + D_1 u(k) + D_2 \delta(k) + D_3 z(k) + g \\ & E_1 x(k) + E_2 u(k) + E_3 \delta(k) + E_4 z(k) \leq h, \end{aligned}$$

where $\delta(k) \in \{0,1\}^{\mathrm{n}_b}$ denotes the vector of binary variables and $z(k) \in \mathbb{R}^{\mathrm{n}_z}$ represents the auxiliary variables resulting from the procedure discussed next. Similarly, the constraints defined through system matrices E and constant vector h arise along with the composition of the MLD model.

In the MLD representation one model applies in which the binary and auxiliary variables that are needed to define the regions are directly included in the model through additional constraints. As compared to the full PWA system description, in this MLD representation one large but tractable model applies, composed simply by stacking the individual linear and PWA model equations plus the auxiliary equations that define the PWA regions for the individual equations, resulting in a model that grows linearly.

C. Transformation into an MLD model

In order to formulate the MLD model as incorporated with the MPC method as a directly solvable optimization problem, the model can be written as an MILP where some of the decision variables are of an integer and some of a real domain. The following statements summarize the conversion (adapted from [18], [19]). Here, binary dummy variables (denoted by $\delta \in \{0,1\}$) are introduced that track whether a certain region applies, which is associated with one of the affine pieces of the PWA function. The constants m, M denote the lower respectively the upper bound of a function $f(x)$ over a bounded set of input variables x. Finally, c denotes an arbitrary constant and the small constant ϵ denotes the machine precision (used to turn a strict inequality into a non-strict inequality that fits the MILP framework):

- $f(x) \leq c \Leftrightarrow \delta = 1$ is equivalent to:

$$\begin{cases} f(x) \leq c + (M-c)(1-\delta) \\ f(x) \geq c(1-\delta) + \epsilon + (m-\epsilon)\delta. \end{cases} \tag{12}$$

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

$$\begin{cases} -\delta_1 + \delta \leq 0 \\ -\delta_2 + \delta \leq 0 \\ \delta_1 + \delta_2 - \delta \leq 1. \end{cases} \tag{13}$$

- $z = \delta f(x)$ is equivalent to:

$$\begin{cases} z \le M\delta \\ z \ge m\delta \\ z \le f(x) - m(1-\delta) \\ z \ge f(x) - M(1-\delta). \end{cases} \tag{14}$$

To briefly illustrate the transformation of a PWA model equation into an MLD model equation using the above statements, we take an expression of the form (4) or (6), i.e.: $f = \min(f_1, f_2)$ that can be replaced by $f = f_1\delta + f_2(1-\delta)$ where $\delta = 1$ iff $f_1 \le f_2$ and $\delta = 0$ otherwise, according the constraints (12) that show how to link a binary variable to a given region. The latter expression of f should again be written $f = z_1 + f_2 - z_2$ with auxiliary variables $z_i = f_i\delta$, $i = 1, 2$, according the constraints (14). Finally, the second system of inequalities (13) is needed when a function (corresponding to binary variable δ) applies only when multiple conditions hold, that again correspond to other binary variables δ_1, δ_2.

V. MPC FOR TRAFFIC CONTROL

Using MPC (see, e.g., [4], [5]), based on measurements of the current state variables at the control step k_c, future states are predicted for a prediction horizon of N_p steps. By optimization of the objective function over this horizon, the sequence of optimal decision variables is determined. Implementing only the first input, the procedure is repeated in a moving horizon fashion.

Amongst the possible optimization goals for traffic models are the maximization of traffic flow, spreading traffic density, and minimizing the variation in control variables (see, e.g., [10]). We chose as our objective function the following linear combination of terms:

$$J(k_\mathrm{c}) = c_1 \frac{J^{\mathrm{MPC}}_{\mathrm{TTS}}(k_\mathrm{c})}{\mathrm{TTS}_\mathrm{norm}} + \sum_{\gamma\in\Gamma} c_{2,\gamma} \frac{J^{\mathrm{MPC}}_{\gamma,\mathrm{TEFC}}(k_\mathrm{c})}{\mathrm{TEFC}_{\gamma,\mathrm{norm}}} + c_3 \frac{J^{\mathrm{MPC}}_{\mathrm{pen}}(k_\mathrm{c})}{\mathrm{pen}_\mathrm{norm}},$$

namely the minimization of the total time vehicles spend in the system (TTS), i.e., the time vehicles wait at an on-ramp or mainstream origin before joining the freeway plus the time spent on the freeway itself, the traffic emissions and fuel consumption, and a penalty term on the variations of the decision variables, respectively, weighted by nonnegative constants $c_1, c_{2,\gamma}$, and c_3. The three respective terms are normalized with the nominal values $\mathrm{TTS}_\mathrm{norm}$, $\mathrm{TEFC}_{\gamma,\mathrm{norm}}$, and $\mathrm{pen}_\mathrm{norm}$.

To elaborate, the first objective term of the MPC controller is to reduce the TTS over the prediction horizon N_p, i.e.,

$$J^{\mathrm{MPC}}_{\mathrm{TTS}}(k_\mathrm{c}) = T \sum_{k\in\mathcal{K}(k_\mathrm{c},k_\mathrm{c}+N_\mathrm{p})} \left(\sum_{(m,i)\in I_\mathrm{all}} L_m \lambda_m \rho_{m,i}(k) + \sum_{o\in O_\mathrm{all}} w_o(k) \right).$$

Here, I_all denotes the set of index pairs (m, i) of all links and segments in the network, and O_all denotes the set of indices of all origins. Further, $\mathcal{K}(k_\mathrm{c}, k_\mathrm{c} + N_\mathrm{p}) = \{Mk_\mathrm{c}, Mk_\mathrm{c} + 1, \ldots, M(k_\mathrm{c} + N_\mathrm{p}) - 1\}$ where M is such that $T_\mathrm{c} = MT$ for simulation time step T and control time step T_c. Note that

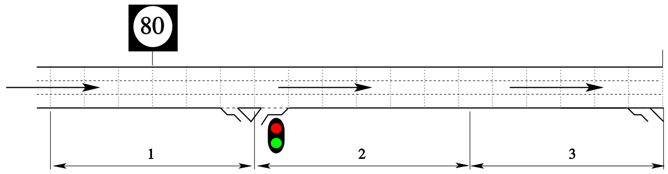


Fig. 4. Set-up of the case study

the TTS cost function term is linear in the state variables $\rho_{m,i}(k)$ and $w_o(k)$.

Further, the total vehicular emissions and fuel consumption (TEFC) introduced in (10) are captured in the linear expression

$$J^{\mathrm{MPC}}_{\gamma,\mathrm{TEFC}}(k_\mathrm{c}) = \sum_{k\in\mathcal{K}(k_\mathrm{c},k_\mathrm{c}+N_\mathrm{p}-1)} J_{\gamma,\mathrm{TEFC}}(k).$$

To reduce the number of decision variables often a control horizon $N_\mathrm{c} < N_\mathrm{p}$ is introduced and from step $k + N_\mathrm{c} - 1$ on the control signals are taken constant. In practice, one also often adds a penalty term on deviations of the decision variables:

$$J^{\mathrm{MPC}}_{\mathrm{pen}}(k_\mathrm{c}) = \sum_{j=1}^{k_\mathrm{c}+N_\mathrm{c}-1} \Bigg\{ \sum_{o\in O_\mathrm{all}} |r_o(k_\mathrm{c}+j) - r_o(k_\mathrm{c}+j-1)| + a_\mathrm{speed} \sum_{(m,i)\in C_\mathrm{all}} |v_{\mathrm{ctrl},m,i}(k_\mathrm{c}+j) - v_{\mathrm{ctrl},m,i}(k_\mathrm{c}+j-1)| \Bigg\},$$

where a_speed is a weighting coefficient and where C_all is the set of all pairs of indices (m, i) of links and segments in which a variable speed limit is active. This penalty term can also be transformed into a linear objective function by introducing additional real-valued auxiliary variables as linear constraints.

All in all, we now end up with an MILP formulation, for which efficient solvers are available (see e.g., [20], [21]).

VI. CASE STUDY, MLD-MPC

Having piecewise-approximated the METANET model and having written it in the MLD model structure, we apply MPC – minimizing the TTS (veh·h), thus for $c_{2,\gamma} = 0, c_3 = 0$ – to a traffic system consisting of a three-segment long freeway with two lanes, an on-ramp placed in between the first and second segment, and dynamic speed control on the first segment. Fig. 4 depicts this set-up. We take the standard parameter settings for the METANET model used in [2] and [10] and simulate over a time horizon of 2.5 hours.

A selection of the results is shown in Table 1, where next to the TTS and the relative difference between the TTS value for the approximated and the original nonlinear model, also the computation time for one run of the optimization step is supplied, averaged over the number of simulation steps. As can be seen in the table, the MLD-MPC approach returns values that are close to the original TTS, yet while needing a much shorter computation time.[1] Here it should be noted that

[1]These times were obtained running Matlab 7.9.0 (R2009b) on a Linux PC with a 2 GHz Intel Celeron processor and 2Gb RAM.

TABLE I

COMPARISON OF TTS (VEH·H) AND CPU TIME (S) AND THEIR RELATIVE DIFFERENCES FOR SEVERAL SCENARIO'S

Scenario	TTS nonlinear MPC	TTS MLD-MPC (% diff.)	CPU time nonlinear MPC	CPU time MLD-MPC (% diff.)
$N_\mathrm{p} = 5, N_\mathrm{c} = 3\,\mathrm{min}$	1318.7 veh·h	1323.6 veh·h (0.37%)	0.907 s	0.173 s (-80.9%)
$N_\mathrm{p} = 7, N_\mathrm{c} = 5\,\mathrm{min}$	1318.7 veh·h	1323.6 veh·h (0.37%)	3.143 s	0.299 s (-90.5%)
$N_\mathrm{p} = 10, N_\mathrm{c} = 9\,\mathrm{min}$	1318.7 veh·h	1333.5 veh·h (1.12%)	10.88 s	0.638 s (-94.14%)

for reliable nonlinear optimization results, each (nonlinear) optimization step should be run several times in order to prevent reaching a local optimum only, meaning that the CPU times for nonlinear MPC (here run 10 times) can in fact be seen as lower bounds on the actual computation times.

In case of an extension to a four-segment network with $N_\mathrm{p} = 7\,\mathrm{min}$ and $N_\mathrm{c} = 5\,\mathrm{min}$, the mean computation time of one optimization step over the simulation horizon for the MLD-MPC approach increases only a little to $0.296\,\mathrm{s}$ as compared to an increase to $3.89\,\mathrm{s}$ for nonlinear MPC. The TTS values are again comparable (1583.9 vs. 1579.9 veh·h). Therefore, and also from the increasing gains in computation time for larger prediction and control horizons that can be seen in Table 1, the MLD-MPC approach is expected to even perform better w.r.t. computational requirements and as compared to the original nonlinear MPC method, when it is applied to larger traffic networks.

VII. CONCLUSIONS

In the current paper, a piecewise-affine (PWA) formulation of the traffic model METANET has been made in order to ease the computational complexity of the original nonlinear nonconvex traffic control problem when adopting model-based predictive control (MPC). Since a direct PWA-MPC implementation turned out to be computationally intractable due to the large number of regions when taking all individual approximations together into the PWA model of the traffic network, we have proposed to use a mixed-logical dynamic (MLD) representation of the approximated model equations, after which MPC could be applied at a significantly improved efficiency and without much deterioration of the objective function value (TTS). Moreover, based on the simulations run, it is expected that the larger the traffic network of application, the more clearly the benefits of the MLD-MPC approach will show as compared to the original nonlinear formulation.

VIII. ACKNOWLEDGMENTS

Research supported by the BSIK project "Next Generation Infrastructures (NGI)", the Shell/TU Delft Sustainable Mobility program, the European 7th Framework Network of Excellence "Highly-complex and networked control systems (HYCON2)", the European COST Action TU0702, and the Transport Research Centre Delft.

REFERENCES

[1] A. Messmer and M. Papageorgiou, "METANET: A macroscopic simulation program for motorway networks," *Traffic Engineering and Control*, vol. 31, no. 9, pp. 466–470, 1990.

[2] A. Kotsialos, M. Papageorgiou, and A. Messmer, "Optimal coordinated and integrated motorway network traffic control," in *Proceedings of the 14th International Symposium of Transportation and Traffic Theory (ISTTT)*, Jerusalem, Israel, Jul. 1999, pp. 621–644.

[3] S. Zegeye, B. De Schutter, H. Hellendoorn, and E. Breunesse, "Model-based traffic control for balanced reduction of fuel consumption, emissions, and travel time," in *Proceedings of the 12th IFAC Symposium on Transportation Systems*, Redondo Beach, CA, U.S.A., Sep. 2009, pp. 149–154.

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

[5] J. Rawlings and D. Mayne, *Model Predictive Control: Theory and Design*. Madison, WI,U.S.A.: Nob Hill Publishing, 2009.

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

[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, Jul. 2006.

[8] A. Hegyi, B. De Schutter, and J. Hellendoorn, "Optimal coordination of variable speed limits to suppress shock waves," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 1, pp. 102–112, Mar. 2005.

[9] N. Groot, S. Zegeye, B. De Schutter, and H. Hellendoorn, "Model-based predictive traffic control: A piecewise-affine approach based on METANET," 2011, accepted for the *8th IFAC World Congress*, Milan, Italy, Aug.–Sept. 2011.

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

[11] K. Ahn, A. Trani, H. Rakha, and M. Van Aerde, "Microscopic fuel consumption and energt emission models," in *Proceedings of the 78th Annual Meeting of the Transportation Research Board*, Washington DC, USA, January 1999.

[12] E. Sontag, "Nonlinear regulation: The piecewise linear approach," *IEEE Transactions on Automatic Control*, vol. 26, no. 2, pp. 346–357, 1981.

[13] E. Bredensteiner and K. Bennett, "Multicategory classification by support vector machines," *Computational Optimizations and Applications*, vol. 12, no. 1–3, pp. 53–79, 1999.

[14] G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari, "A clustering technique for the identification of piecewise affine systems," *Automatica*, vol. 39, no. 2, pp. 205–217, 2003.

[15] G. Ferrari-Trecate, "Hybrid identification toolbox (HIT)," http://www-rocq.inria.fr/who/Giancarlo.Ferrari-Trecate/HIT_toolbox.html, 2005.

[16] R. Fletcher, *Practical Methods of Optimization, Volume 1: Unconstrained Optimization*. Chichester, UK: John Wiley & Sons, 1980.

[17] S. Zegeye, B. De Schutter, and J. Hellendoorn, "Model predictive traffic control to reduce vehicular emissions – An LPV-based approach," in *Proceedings of the 2010 American Control Conference*, Baltimore, MD, U.S.A., Jun.–Jul. 2010, pp. 2284–2289.

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

[19] H. Williams, *Model Building in Mathematical Programming*, 3rd ed. New York, NY, U.S.A.: Wiley, 1993.

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

[21] J. Linderoth and T. Ralphs, "Noncommercial software for mixed-integer linear programming," in *Integer Programming: Theory and Practice*, J. Karlof, Ed. CRC Press Operations Research Series, 2005, pp. 253–303.