

Technical report 13-003

# Integrated model predictive traffic and emission control using a piecewise-affine approach\*

N. Groot, B. De Schutter, and H. Hellendoorn

*If you want to cite this report, please use the following reference instead:*

N. Groot, B. De Schutter, and H. Hellendoorn, "Integrated model predictive traffic and emission control using a piecewise-affine approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 587–598, June 2013. doi:[10.1109/TITS.2012.2227314](https://doi.org/10.1109/TITS.2012.2227314)

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://pub.bartdeschutter.org/abs/13\\_003.html](https://pub.bartdeschutter.org/abs/13_003.html)

# Integrated Model Predictive Traffic and Emission Control Using a Piecewise-Affine Approach

Noortje Groot, *Student Member, IEEE*, Bart De Schutter, *Senior Member, IEEE*, and Hans Hellendoorn

**Abstract**—This paper addresses the computational intractability of traffic control when applying the integrated METANET freeway traffic model and the VT-macro emission model in a model-based predictive control (MPC) framework. In order to facilitate real-time implementation, a piecewise-affine (PWA) approximation of the nonlinear METANET model is proposed. While a direct MPC approach based on the full PWA model is intractable for on-line applications, a conversion to a mixed-logical dynamic (MLD) model description is made instead. The resulting MLD-MPC problem, written as a mixed-integer linear program, can be solved much more efficiently as it does not explicitly state all model equations for each particular region. As a benchmark, the computational efficiency and accuracy of the MLD-MPC approach is tested on a case study including variable speed limits and a metered on-ramp while optimizing the total time spent, as well as taking into account emissions and fuel consumption of the vehicles. The performance is evaluated against the original nonlinear and nonconvex MPC problem and shows an improved computational speed at the cost of some deviation in the cost function values.

**Index Terms**—Emission control, model-based predictive control (MPC), piecewise linear approximation, ramp metering, variable speed limits

## I. INTRODUCTION

**I**N the model-based control of large-scale traffic networks it is important to adopt a modeling framework that is both accurate and that yields a fast solution when incorporated with the optimization framework, in order to be able to apply on-line traffic control. An often used model for freeways is the second-order macroscopic METANET model [1]–[3]. Such a model is commonly used as it has shown to provide good accuracy while it does not require as much computation time as microscopic traffic models that take individual vehicles into account [4]. The main variables considered in METANET are the density, flow, and average velocity of traffic. This model can be complemented by the VT-macro model for vehicular emission and fuel consumption [5].

As control framework, a well-known method is model-based predictive control (MPC) [6], [7]. In the application of MPC to traffic systems, based on measurements of the system, optimal control inputs are computed, e.g., on-ramp metering rates and variable speed limits that yield an optimal traffic throughput based on both current and predicted, future state variables. After the implementation of the first set of these control inputs the process is repeated, which is referred to as the moving horizon approach of MPC. MPC has often been adopted as

it easily incorporates various constraints and adapts well to uncertain systems and structural changes in the system due to the moving horizon strategy. Additionally, the prediction model can be adapted during the control process.

When using the METANET traffic flow model complemented by the VT-macro emission and fuel consumption model in combination with MPC in order to minimize the total time spent (TTS) by traffic in the network as well as to reduce the vehicular emissions, a nonlinear and nonconvex optimization problem results. Such a problem can be solved with global or multi-start local optimization [8]–[10]. However, this approach is subject to computational issues that prevent the real-time implementation on traffic networks, and a global optimum cannot be guaranteed. Hence, the main objective we like to address is to develop an accurate yet computationally efficient approach to applying MPC based on the integrated METANET and VT-macro models. One way to address this computational inefficiency is to approximate the underlying model [11]. Other ways are to address the optimization approach itself and reformulate, e.g., decompose or distribute the problem solving [12], [13].

In the current paper, we focus on the underlying model and propose to adopt a piecewise-affine (PWA) approximation of the nonlinear elements within METANET and VT-macro. Nonlinear functions can be approximated by PWA functions with arbitrary accuracy, partitioning the function domain in a finite number of polyhedra, each associated with an affine function. Based on the PWA model, an iteration of the MPC problem can then be more easily solved to optimality when formulated as a mixed-integer linear program (MILP). However, since the class of integer programs is proven to be NP-complete [14], attention should be paid to keeping the number of binary variables caused by the PWA model formulation small, as they increase the MILP’s complexity. At the same time, the fewer such variables are allowed in the approximation and thus the fewer the number of affine pieces, the larger the discrepancy with the original function. In other words, it is important to find a good trade-off between accuracy of the approximation and computational complexity.

In earlier work, the intractability of MPC based on a fully PWA model was pointed at, resulting from the large number of regions that need to be considered in the integrated PWA model for the entire traffic network [15]. Instead, applying MPC based on a mixed-logical dynamic (MLD) description of the PWA system showed encouraging results towards the real-time application of MPC in traffic control with increased computational gains when considering larger control and prediction horizons. Details on methods to arrive at a PWA approximation

N. Groot, B. De Schutter, 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).

of the METANET model can be found in [16], where several levels of accuracy of the individual approximations were evaluated. In the current paper, the superior performance of the PWA-MPC method as compared to the original nonlinear MPC is illustrated in the more encompassing context of integrated speed limit and ramp metering controls, optimizing also the vehicular emissions.

The remainder of this paper is organized as follows. In Section II the original METANET traffic flow model is presented together with the VT-macro model for vehicular emissions and fuel consumption. Section III includes a description of the MPC approach and the optimization objectives we propose for traffic control. In Section IV we describe how the nonlinear model equations can be approximated in a PWA manner according one of the selected methods. Subsequently, it is shown how the resulting PWA model can be recast as an MLD model in order to arrive at a feasible MILP (Section V). The proposed MLD-MPC approach is applied in the case study of Section VI, where its performance w.r.t. accuracy and computational speed is briefly analyzed. Conclusions and recommendations are presented in Section VII.

## II. THE METANET AND VT-MACRO MODELS

### A. METANET

The original METANET model for traffic flows as developed by Papageorgiou and Messmer [1], [2] is discrete in time and space. The traffic network can be described by a graph with links representing homogeneous parts of a freeway, separated by nodes representing changes like on-ramps and the increase or decrease of the number of lanes. Links are further divided into segments of equal distance. As regards the discretization in time, typically a simulation time step  $T_s$  of about 10 s is used, where  $t = kT_s$  for a time instant  $t$  and the corresponding time step counter  $k$ .

The evolution of traffic flow  $q_{m,i}$  (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:

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

$$\rho_{m,i}(k+1) = \rho_{m,i}(k) + \frac{T_s}{L_m \lambda_m} [q_{m,i-1}(k) - q_{m,i}(k)] \quad (2)$$

$$\begin{aligned} v_{m,i}(k+1) = & v_{m,i}(k) + \frac{T_s}{\tau} [\rho_{m,i}(k)] - v_{m,i}(k) \\ & + \frac{T_s v_{m,i}(k) [v_{m,i-1}(k) - v_{m,i}(k)]}{L_m} \\ & - \frac{T_s \eta [\rho_{m,i+1}(k) - \rho_{m,i}(k)]}{\tau L_m (\rho_{m,i}(k) + \kappa)}, \end{aligned} \quad (3)$$

with  $\lambda_m$  the number of lanes in link  $m$ ,  $L_m$  the length of the segments of link  $m$  (m), and  $\eta$  (km<sup>2</sup>/h),  $\kappa$  (veh/km/lane), and  $\tau$  (h) model parameters. Commonly used values of these and other parameters are provided in Table I.

The desired speed  $V[\rho_{m,i}(k)]$  (km/h) is represented by:

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

where the second term applies in case of variable-speed control on segment  $i$  of link  $m$ , where the speed limit is denoted by the speed control variable  $v_{\text{ctrl},m,i}(k)$  (km/h) [17]. Here,  $v_{\text{free},m}$  (km/h) denotes the free-flow speed and  $\alpha$  is a factor set to model the non-compliance of traffic participants to the speed limit based on whether the limit is obligatory or recommended, resulting in a lower respectively higher speed. 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  $o$ :

$$w_o(k+1) = w_o(k) + T_s(d_o(k) - q_o(k)). \quad (5)$$

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

$$\begin{aligned} q_o(k) = \min \left[ d_o(k) + \frac{w_o(k)}{T_s}, r_o(k) C_o, \right. \\ \left. C_o \left( \frac{\rho_{\text{jam},m} - \rho_{m,1}(k)}{\rho_{\text{jam},m} - \rho_{\text{cr},m}} \right) \right], \end{aligned} \quad (6)$$

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) represents the maximum density of a link  $m$  connected to the given origin.

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

$$-\frac{\delta T_s q_o(k) v_{m,1}(k)}{L_m \lambda_m (\rho_{m,1}(k) + \kappa)}. \quad (7)$$

As a final part of the model, equations should be added that take into account the existence of multiple incoming and outgoing links of the nodes and accordingly the distribution of flow and density. The following two equations represent the total flow  $Q_n(k)$  (veh/h) entering a node  $n$  and the flow  $q_{m,0}(k)$  leaving  $n$  via link  $m$ :

$$Q_n(k) = \sum_{\mu \in I_n} q_{\mu, N_\mu}(k) \quad (8)$$

$$q_{m,0}(k) = \beta_{n,m}(k) Q_n(k). \quad (9)$$

Here,  $I_n$  denotes the set of links entering  $n$ ,  $N_m$  the index of the last segment of a link  $m$  that enters node  $n$ , and  $\beta_{n,m}(k)$  the fraction of the total flow to node  $n$  leaving via link  $m$ .

For a node with multiple outgoing links, the virtual downstream density  $\rho_{m, N_m+1}(k)$  ('virtual' due to the nonexistence of a segment of index  $N_m + 1$ ) of link  $m$  entering an origin is modeled as:

$$\rho_{m, N_m+1}(k) = \frac{\sum_{\mu \in O_n} \rho_{\mu,1}^2(k)}{\sum_{\mu \in O_n} \rho_{\mu,1}(k)}, \quad (10)$$

with  $O_n$  the set of outgoing links for node  $n$ .

Similarly, for a node with multiple incoming links, the virtual upstream speed  $v_{m,0}(k)$  of outgoing link  $m$  is modeled as:

$$v_{m,0}(k) = \frac{\sum_{\mu \in I_n} v_{\mu, N_\mu}(k) q_{\mu, N_\mu}(k)}{\sum_{\mu \in I_n} q_{\mu, N_\mu}(k)}. \quad (11)$$

TABLE I  
METANET PARAMETER SETTINGS [2], [17]

$T_s = 10$ s	$\kappa = 40$ veh/km/lane	$\delta = 0.0122$	$\eta = 60$ km <sup>2</sup> /h
$\tau = 18$ s	$\rho_{\max} = 180$ veh/km/lane	$a_m = 1.867$	$v_{\text{free}} = 102$ km/h
$L_m = 1$ km	$\rho_{\text{crit}} = 33.5$ veh/km/lane	$\alpha = 0.1$	

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 [2], [10], [17], [18].

### B. VT-macro Emission Model

In order to take into account emissions and fuel consumption of the vehicles, the METANET model can be extended with the equations of the VT-macro model. For more detailed information on this model, the reader is referred to [5]. 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 segment  $i$  to the next segment  $i + 1$  of a link  $m$  are given by:

$$a_{m,i,i+1}(k) = \frac{v_{m,i+1}(k) - v_{m,i}(k-1)}{T_s} \quad (12)$$

$$n_{m,i,i+1}(k) = T_s q_{m,i}(k). \quad (13)$$

Similar expressions apply to e.g., on-ramps, off-ramps, junctions, etc. Slightly different, the temporal accelerations and number of vehicles subject to it refer to the values of those variables within the same segment  $i$  of a link  $m$ :

$$a_{m,i}(k) = \frac{v_{m,i}(k) - v_{m,i}(k-1)}{T_s} \quad (14)$$

$$n_{m,i}(k) = L_m \lambda_m \rho_{m,i}(k) - T_s q_{m,i}(k). \quad (15)$$

The vehicular emissions and fuel consumption become apparent in the cost function of the traffic control problem when minimizing the following expression for the total emissions [g] or fuel consumption [l] within the time period  $[kT_s, (k+1)T_s]$ :

$$J_{\gamma, \text{TEFC}}(k) = T_s \sum_{\ell \in L_{\text{all}}} n_{\ell}(k) \exp(\check{v}_{\ell}^{\top}(k) P_{\gamma} \check{a}_{\ell}(k)), \quad (16)$$

with the speed and acceleration vectors  $\check{v}_{\ell}^{\top}(k) = [1 \ v_{\ell}(k) \ v_{\ell}^2(k) \ v_{\ell}^3(k)]$  and  $\check{a}_{\ell}^{\top}(k) = [1 \ a_{\ell}(k) \ a_{\ell}^2(k) \ a_{\ell}^3(k)]$ , and with  $L_{\text{all}}$  the set of indices of all triples  $(a_{\ell}, n_{\ell}, v_{\ell})$  of spatio-temporal or temporal accelerations and the corresponding numbers of vehicles and speeds. Moreover,  $P_{\gamma}$  denotes the model parameter for  $\gamma \in \Gamma = \{\text{CO emission, HC emission, NO}_x \text{ emission, fuel consumption}\}$ . The values of the parameter matrices  $P_{\gamma}$  can be found in [19]. Plots of the different emission components can be found in Fig. 1.

### III. MPC FOR TRAFFIC CONTROL

Using MPC [6], [7], based on measurements of the current state variables at the control step  $k_c$ , future states are predicted for a prediction horizon of  $N_p$  control steps, using the model presented in the previous section. By optimization of the objective function over this horizon, the sequence of optimal

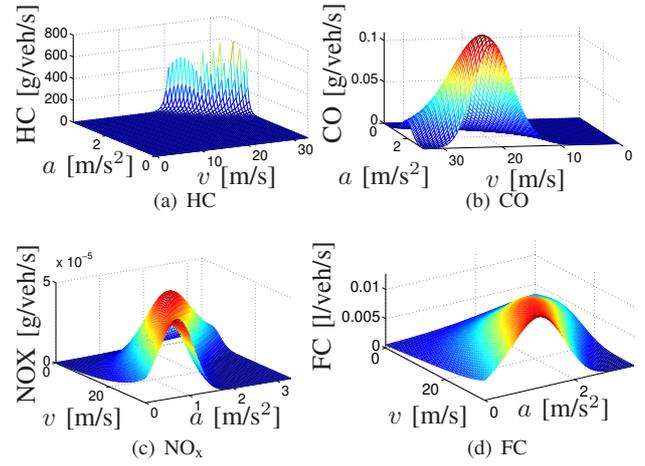


Fig. 1. Plots of the vehicular emissions and fuel consumption term  $\exp(\check{v}_{\ell}^{\top} P_{\gamma} \check{a}_{\ell})$  in (16)

decision variables is determined. Implementing only the first of these inputs, the procedure is repeated in a moving horizon fashion for a simulation time horizon with a simulation time step  $T$  and control time step  $T_c$ .

Amongst the possible optimization goals for traffic networks are the maximization of traffic flow, spreading traffic density, and minimizing the variation in control variables [17]. We chose as our objective function the following linear combination of terms:

$$J(k_c) = c_1 \frac{J_{\text{TTS}}^{\text{MPC}}(k_c)}{\text{TTS}_{\text{norm}}} + \sum_{\gamma \in \Gamma} c_{2,\gamma} \frac{J_{\gamma, \text{TEFC}}^{\text{MPC}}(k_c)}{\text{TEFC}_{\gamma, \text{norm}}} + c_3 \frac{J_{\text{pen}}^{\text{MPC}}(k_c)}{\text{pen}_{\text{norm}}}, \quad (17)$$

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, 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  $\text{TTS}_{\text{norm}}$ ,  $\text{TEFC}_{\gamma, \text{norm}}$ , and  $\text{pen}_{\text{norm}}$ , which are obtained by considering the uncontrolled system.

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

$$J_{\text{TTS}}^{\text{MPC}}(k_c) = T_s \sum_{k \in \mathcal{K}(k_c, k_c + N_p)} \left( \sum_{(m,i) \in I_{\text{all}}} L_m \lambda_m \rho_{m,i}(k) + \sum_{o \in O_{\text{all}}} w_o(k) \right). \quad (18)$$

Here,  $I_{\text{all}}$  denotes the set of index pairs  $(m, i)$  of all links and segments in the network, and  $O_{\text{all}}$  denotes the set of indices of all origins. Further,  $\mathcal{K}(k_c, k_c + N_p) = \{Mk_c, Mk_c + 1, \dots, M(k_c + N_p) - 1\}$  where  $M$  is such that  $T_c = MT$ . Note that 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 introduced in (16) are captured in the linear expression

$$J_{\gamma, \text{TEFC}}^{\text{MPC}}(k_c) = \sum_{k \in \mathcal{K}(k_c, k_c + N_p)} J_{\gamma, \text{TEFC}}(k). \quad (19)$$

Hence, if a linearized expression for (16) is found, the emission and fuel consumption factors enter the objective function linearly.

Finally, the penalty term on deviations of the decision variables can be written:

$$J_{\text{pen}}^{\text{MPC}}(k_c) = \sum_{j=1}^{N_c-1} \left\{ \sum_{o \in O_{\text{all}}} |r_o(k_c + j) - r_o(k_c + j - 1)| + a_{\text{speed}} \sum_{(m,i) \in C_{\text{all}}} |v_{\text{ctrl},m,i}(k_c + j) - v_{\text{ctrl},m,i}(k_c + j - 1)| \right\}, \quad (20)$$

where  $a_{\text{speed}}$  is a nonnegative weighting coefficient and where  $C_{\text{all}}$  is the set of all pairs of indices  $(m, i)$  of links and segments in which a variable speed limit is applied. In order to reduce the number of decision variables often a control horizon  $N_c < N_p$  is introduced and from control step  $k_c + N_c - 1$  onwards the control signals are taken to be constant. Note that instead of a 1-norm, alternatively a quadratic penalty term could be adopted. This penalty term can be transformed into a linear form, such that the overall objective function is linear and convex. This transformation, for which some additional real-valued auxiliary variables need to be introduced, will be explained in Section V.

All in all, given a linear expression for (16), the above MPC objective is linear and convex, yet the underlying METANET prediction model is nonlinear and nonconvex. As a result, the optimization problem based on the original models is intractable for real-life implementation. Hence, in order to reduce the computation time, in the following, a PWA approximation of the METANET model is proposed.

#### IV. PWA APPROXIMATION

A function  $f: \Omega \rightarrow \mathbb{R}^m$  is PWA if there exists a polyhedral partition  $\{\Omega_i\}_{i \in \mathcal{I}}$  of  $\Omega \subseteq \mathbb{R}^n$  such that  $f$  is affine on each polyhedron  $\Omega_i$ , i.e.,

$$f(x) = A_i x + b_i \quad \forall x \in \Omega_i, \forall i \in \mathcal{I}, \quad (21)$$

with  $A_i, b_i$  constants. Here, a polyhedral partition of  $\Omega$  represents a finite number of nonempty polyhedra  $\{\Omega_i\}_{i \in \mathcal{I}}: \cup_{i \in \mathcal{I}} \Omega_i = \Omega, \Omega_i \cap \Omega_j = \emptyset \forall i \neq j$ . For a continuous PWA function we only require that  $\text{int}(\Omega_i) \cap \text{int}(\Omega_j) = \emptyset \forall i \neq j$ . For more information on general PWA theory, see [20].

One can approximate a nonlinear function in a PWA manner with arbitrary accuracy, i.e., by considering a sufficiently large number of regions. However, as will be pointed out in Section V, this comes at the cost of a larger computational burden if the approximated model equation is to be applied in the eventual optimization approach. Therefore, as a main consideration in the PWA approximation, the number of affine pieces should be kept small, while safeguarding a close match to the original traffic model. In return for a less accurate model, one then arrives at a PWA model description that is faster to deal with in optimizations and which solution could alternatively be used as an initial starting point for control when using the nonlinear model.

In the remainder of this section a selection is provided of possible methods to arrive at a PWA approximation. These methods will subsequently be used for the approximation of the nonlinear METANET equations in Section IV-B.

#### A. PWA Approximation Methods

Four approaches for PWA approximation of nonlinear functions are least-squares optimization, PWA identification, (partially) piecewise constant approximation, and PWA approximation of a multivariate function by reduction to separable quadratic terms. Here it should be noted that there is a large difference in complexity between the approximation of single and multi-variable functions, both of which occur in the METANET setting we consider in this paper. More information on the various available methods for PWA function approximations can be found in [21] and [22].

1) *Least-squares optimization*: A well-known optimization-based approximation approach comprises the minimization of the squared error or difference between the original function and the approximation curve. For single-variate nonlinear functions this method is easy to apply and accurate. After one specifies the desired number of regions or intervals of the PWA function, both the optimal intervals and parameters of the affine functions are determined using least-squares optimization. Additionally, one may add positive weights to parts of the function that especially require a high accuracy. E.g., the following PWA problem may be solved in a least-squares manner – here given for an approximation of a function  $f$  defined on an interval  $[x_{\min}, x_{\max}]$  by a continuous PWA function  $f_{\text{PWA}}(x)$  with three intervals:

$$\min_{\alpha, \beta, \gamma, \delta, \epsilon, \zeta} \int_{x_{\min}}^{x_{\max}} w(x) (f_{\text{PWA}}(x) - f(x))^2 dx \quad (22)$$

s.t.

$$f_{\text{PWA}}(x) = \begin{cases} \gamma + \frac{x - x_{\min}}{\alpha - x_{\min}} (\delta - \gamma) & \text{for } x_{\min} \leq x < \alpha \\ \delta + \frac{x - \alpha}{\beta - \alpha} (\epsilon - \delta) & \text{for } \alpha \leq x < \beta \\ \epsilon + \frac{x - \beta}{x_{\max} - \beta} (\zeta - \epsilon) & \text{for } \beta \leq x \leq x_{\max}, \end{cases} \quad (23)$$

where  $w$  denotes a weighting function. Least-squares optimization can be solved using e.g., a multi-start Gauss-Newton or Levenberg-Marquardt approach [23].

2) *Piecewise-affine identification*: An alternative approach is hybrid or piecewise-affine identification. Differently from the previous method, PWA identification is a clustering algorithm that returns a PWA approximation based on a set of data points or discretized version of a model. Therefore, this approach is especially useful for complex, multi-variate functions. In general, the three available methods described next create local data sets after which the clustering algorithm creates local affine models by classifying the points. Similar models are again grouped into clusters, depending on the number of regions required [21].

Amongst the available methods for identification, the most precise for bivariate identification is the algorithm Multicategory Robust Linear Programming (MRLP) [24]. However, this method is computationally expensive and generally works for the identification of up to three polytopes based on up to 200 data points. This is due to the fact that one linear program is solved to find boundaries of all regions simultaneously. Alternatively, the clustering algorithms Support Vector Classification (SVC) and Proximal Support Vector Classification

(PSVC) can be used, yet for multivariate estimation the original domain of the variables may then not be completely covered by the union of computed subregions. In contrast to MRLP the SVC approach [25] solves several quadratic programs in order to sequentially find boundaries between two regions or half-spaces at a time. PSVC [26] is the most time-efficient algorithm of the three and only requires a single system of linear equations. Compared to the non-proximal version, it assigns data points to the closest of two parallel half-planes that are maximally separated, leading to a strongly convex objective.

These algorithms are implemented in the Hybrid Identification Toolbox (HIT) [27], a platform embedded within the Multi-Parametric Toolbox for Matlab [28].

3) *Partially piecewise-constant approximation*: An approximation approach for bivariate functions that uses relatively few auxiliary variables is by segmentation of the domain of one of the variables, where in each region or subdomain that variable is assigned a constant value. In general, a bivariate function  $f(x, y)$  can be approximated as follows. Assume that based on the relative ranges  $\frac{x_{\max} - x_{\min}}{x_{\max}}$  and  $\frac{y_{\max} - y_{\min}}{y_{\max}}$  (in case  $x_{\max} = 0$  or  $y_{\max} = 0$ , only the numerator applies) or the magnitude of the partial derivatives, variable  $x$  is selected to be taken constant in each region. For a selection of  $N$  consecutive intervals  $[x_i, x_{i+1}]$  for  $i = 1, \dots, N - 1$  and with  $x_1 = x_{\min}$ ,  $x_N = x_{\max}$ , we can set e.g.,

$$f(x, y) \approx f\left(\frac{x_i + x_{i+1}}{2}, y\right) \quad \text{for } x \in [x_i, x_{i+1}]. \quad (24)$$

Now, if  $f$  is linear in  $y$  (as will be the case for several functions appearing in the METANET model), this approach results in a PWA approximation of  $f$ . Alternatively the least-squares optimization approach discussed in Section IV-A1 can be applied for each function  $f\left(\frac{x_i + x_{i+1}}{2}, y\right)$ .

Substitution of one of the variables can be seen as a specific case of PWA approximation where instead of an affine piece, a constant value is associated with each region of the domain. Nonetheless, this piecewise-constant approach can deliver adequate approximation results for some functions. In the particular case that the approximated variable coincides with the mean value of the region, no approximation error applies, which is generally not the case for an approximation derived from PWA identification. For single-variate functions this partially piecewise-constant approach may also be applied, but in general it is not very difficult to obtain a more accurate PWA formulation for this class of functions.

4) *PWA approximation by reduction*: Finally, as proposed in [29], [30], a bivariate term of the structure  $x \cdot y$ , with  $x, y \in \mathbb{R}$  can be recast without approximation in the equivalent form  $\frac{1}{4}[z_+^2 - z_-^2]$ , where two new real variables are introduced, i.e.,  $z_+ := x + y$ ,  $z_- := x - y$ . The PWA approximation now reduces to replacing both quadratic terms  $z_+^2, z_-^2$  in a PWA manner, for which any of the previous methods can be adopted. Since these terms can be approximated independently, this approach results in a relatively small number of binary variables, also if it would be extended to factors of higher order.

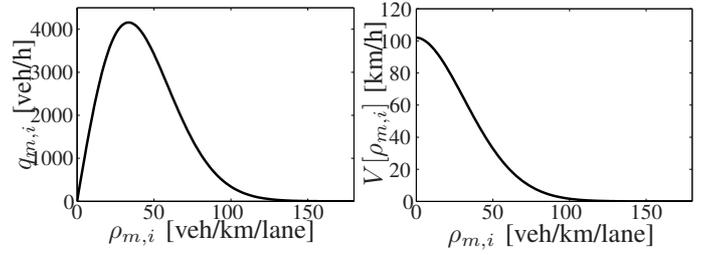


Fig. 2. Fundamental diagrams of traffic flow

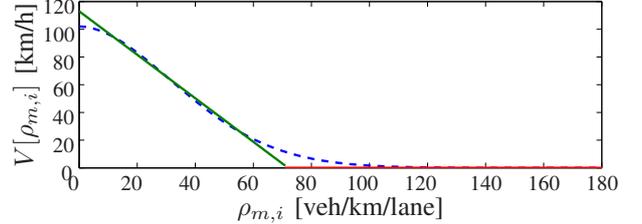


Fig. 3. PWA approximation of the fundamental diagram in two pieces

## B. PWA Approximation of METANET and VT-macro

In this section the nonlinear elements of the METANET model are dealt with. In general, application-specific knowledge could be applied in the approximation of model equations. In other words, information from other model equations can facilitate the approximation, e.g., by weighting areas that require a close match. When applied to this specific case, it indeed pays off to make use of physical information, e.g., the fundamental diagram of traffic flow depicted in Fig. 2 will be used also in the approximation of other functions of the METANET model.

First, note that (2), (5), and (6) do not need to be approximated as the first two functions are already linear and the latter equation is PWA. The remaining nonlinear terms can be divided into three main groups:

1) *Fundamental diagram*: The first nonlinear term of (4) corresponds to one of the fundamental diagrams of traffic flow depicted in Fig. 2, which represent the equilibrium relations between speed, flow, and density in a homogeneous part of a freeway [31]. For this single-variate nonlinear term, a least-squares approach is adopted as shown in Fig. 3. An approximation in two affine pieces was applied, as this yielded a good accuracy at a better efficiency as compared to a more accurate division in three pieces. Since the second term in (4) is a linear expression, a PWA fundamental diagram combined with a variable speed limit leads to a PWA expression of the desired speed equation.

2) *Speed equation (3)*: Several variables that give rise to the nonlinear terms in (3) are kept constant at a value determined by historical data or equal to the currently measured value for predictions in receding horizon. The accuracy can be improved by adopting a sequence of different, predicted values that varies for each simulation step, i.e., by using the values of the state variables that result from a simulation of the traffic prediction model, in which the optimal control sequence consisting of  $N_c$  control inputs is considered. Alternatively, a more exact approximation could be obtained using PWA

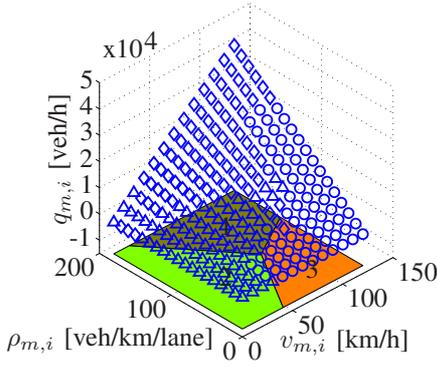


Fig. 4. PWA approximation of flow equation (1) by hybrid identification using the Hybrid Identification Toolbox [27]

identification or a piecewise-constant approach. However, the improvement in approximation accuracy may not justify the increase in computational complexity that this causes.

- $v_{m,i}(k)[v_{m,i-1}(k) - v_{m,i}(k)]$ . Here, the first velocity variable  $v_{m,i-1}(k)$  is substituted by a constant value. Note that the absolute approximation error caused by this method is relatively seen decreased due to the multiplication of the replaced velocity by the relatively small term  $T_s/L_m$  ( $2.78 \cdot 10^{-3}$  h/km: refer to Table I for typical values of the parameters used).
- $\frac{\rho_{m,i+1}(k) - \rho_{m,i}(k)}{\rho_{m,i}(k) + \kappa}$ . As in the previous item, the density term in the denominator is kept constant at a historically-based value, according to measurements, or it is determined based on a sequence of predictions using the computed control variables. Note that the multiplication factor  $\eta T_s / \tau L_m$  in the numerator is rather large (33.33 km/h), as is the addition of  $\kappa = 40$  veh/km/lane to the approximated variable in the denominator. Both aspects cause a reduction to the effect of the error of the approximated denominator w.r.t. the speed variable  $v_{m,i}(k)$ .
- Subtraction of the term (7). Final adaptations are made to this speed-drop term by substituting the density variable in the denominator by a constant value, combined with the substitution of  $q_0 \cdot v_{m,1}$  as in the PWA approximation of the flow equation (1).

3) *Flow equation (1)*: The bivariate equation modeling traffic outflow can be approximated by PWA identification, reduction to separable quadratic terms, and by a piecewise-constant approach for one of the variables. When adopting the latter method we choose to substitute the velocity variable  $v_{m,i}(k)$ , having the smallest domain, by the mean value of each subdomain, transforming (1) into:

$$q_{m,i}(k) = \lambda_m \rho_{m,i}(k) \frac{v_j + v_{j+1}}{2} \text{ for } v_{m,i}(k) \in [v_j, v_{j+1}]. \quad (25)$$

Here, the intervals  $[v_j, v_{j+1}]$  can be chosen individually by taking into consideration the shape of the function or determined in a more sophisticated way by using optimization.

In the approximation of (1) it is further important to take into account the shape of the fundamental diagram shown in Fig. 2(a) and (b). To be more precise, in order to increase

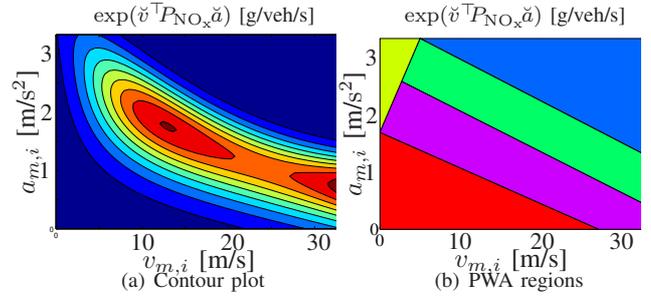


Fig. 5. Domain and proposed regions of the PWA approximation of  $\text{NO}_x$

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. Therefore, the focus should be on a good match in the area around the function values as determined by the fundamental diagram. The final PWA approximation by PWA identification can be seen in Fig. 4.

Here it should be noted that in both approximation methods, the relative approximation error of the flow variable  $q_{m,i}(k)$  is of the same order as the approximation error in  $\rho_{m,i}(k)v_{m,i}(k)$  (PWA identification) or in  $v_{m,i}(k)$  (piecewise-constant approach). To further put the error in perspective, it should be noted that the flow variable occurs in the density function (2), where the difference  $q_{m,i-1}(k) - q_{m,i}(k)$  between consecutive segments is multiplied by the relatively small constant  $T_s / \lambda_m L_m$  ( $1.39 \cdot 10^{-3}$  h/km for  $\lambda_m = 2$ ). Hence, approximation errors of (1) are relatively seen reduced when considering the complete METANET model. On the other hand, it should not be forgotten that errors are re-used in different segments of the model and that by the iterative nature of MPC, errors re-appear also in variables for later time steps of the prediction horizon.

Finally, note that the on-ramp flow equation (6) is already PWA, which means that together with the originally linear equations (2), (5), (12), and (13) we now have a system of only linear and PWA model equations. However, since the term  $J_{\gamma, \text{TEFC}}(k)$  from (16) causes the optimization objective to become a nonlinear nonconvex function, a final PWA approximation should be made.

4) *Total emission and fuel consumption (16)*: In order to arrive at a PWA expression for (16), the exponential term  $\exp(\tilde{v}_\ell^T(k) P_\gamma \tilde{a}_\ell(k))$  is replaced by a PWA approximation using PWA identification for each of the emission and fuel consumption elements. In Fig. 5(a) a filled contour plot of this term for  $\text{NO}_x$  as it was plotted in Fig. 1(c) is provided, together with a division of the domain into regions, each of which is associated with an affine expression that we obtained by hybrid identification as explained in Section IV-A2. Further, to reduce the number of additional variables and therefore the computational complexity, in the original VT-macro model it is suggested to substitute the variable  $n_\ell(k)$  by a constant value [11]. Hence,  $n_\ell(k)$  can again be taken as a constant equal to the currently measured or predicted values, as explained for the approximation of (3).

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

## V. THE PWA-MPC PROBLEM

We now have gathered all ingredients for an implementation of model predictive traffic control using the integrated PWA METANET and VT-macro prediction models. As will be elaborated upon in this section, a full description of the PWA traffic system still leads to an intractable problem formulation when written as a mixed-integer linear program (MILP). Therefore we propose a mixed-logical dynamic formulation instead, which can as well be solved as an MILP when incorporated with MPC, yet in a more efficient manner.

### 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 linear and PWA model equations of METANET and VT-macro and to rewrite them into one coherent PWA description of the entire traffic network. A discrete-time PWA dynamical system in state space notation can be described as follows:

$$\begin{aligned} x(k+1) &= A_i x(k) + B_i u(k) + f_i \\ y(k) &= C_i x(k) + D_i u(k) + g_i \\ [x^T(k) \ u^T(k)]^T &\in \Omega_i, \ i \in \mathcal{I}, \end{aligned} \quad (26)$$

where  $x(k) \in \mathbb{R}^{n_x}$ ,  $u(k) \in \mathbb{R}^{n_u}$ , and  $y \in \mathbb{R}^{n_y}$  denote respectively the state, input, and output vector, and where  $\Omega_i, \cup_{i \in \mathcal{I}} \Omega_i = \mathbb{R}^{n_x+n_u}$  is a convex polyhedron,  $\text{int}(\Omega_i) \cap \text{int}(\Omega_j) = \emptyset, \forall i \neq j \in \mathcal{I}$ . For each region  $\Omega_i$ ,  $A_i \in \mathbb{R}^{n_x \times n_x}$ ,  $B_i \in \mathbb{R}^{n_x \times n_u}$ ,  $C_i \in \mathbb{R}^{n_y \times n_x}$ ,  $D_i \in \mathbb{R}^{n_y \times n_u}$ , and  $f_i \in \mathbb{R}^{n_x}$ ,  $g_i \in \mathbb{R}^{n_y}$  represent constant system matrices.

Furthermore, recall that the state variables refer to the mean velocities  $v_{m,i}(k)$ , densities  $\rho_{m,i}(k)$ , and flows  $q_{m,i}(k)$  of vehicles, together with the flows  $q_o(k)$  and queue lengths  $w_o(k)$  at the origins, and the (spatial)-temporal components of the space-mean accelerations  $a_\ell(k)$  and numbers of vehicles  $n_\ell(k)$ . The input variables refer to the variable speed limits  $v_{\text{ctrl},m,i}(k)$  and ramp metering rates  $r_o(k_c)$ .

To obtain the above PWA system description, the individual PWA model equations should be combined for each link, segment, node, and origin 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 this results in, the composition of the full PWA traffic model is already inefficient. Moreover, when using MPC as explained in Section III, this PWA model has to be evaluated over several future time steps, which causes this PWA-MPC approach for the METANET model (where an MILP is used for optimization [32]) 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 equivalent MLD description:

$$\begin{aligned} x(k+1) &= Ax(k) + B_1 u(k) + B_2 \delta(k) + B_3 z(k) + f \\ y(k) &= Cx(k) + D_1 u(k) + D_2 \delta(k) + D_3 z(k) + g \\ E_1 x(k) + E_2 u(k) + E_3 \delta(k) + E_4 z(k) &\leq h, \end{aligned} \quad (27)$$

where  $\delta(k) \in \{0, 1\}^{n_b}$  denotes a vector of binary variables and  $z(k) \in \mathbb{R}^{n_z}$  represents the auxiliary variables resulting from the procedure discussed next [32]. Similarly, the constraints defined through system matrices  $E$  and constant vector  $h$  arise along with the composition of the MLD model. As in the PWA system (26)  $x(k) \in \mathbb{R}^{n_x}$ ,  $u(k) \in \mathbb{R}^{n_u}$ , and  $y(k) \in \mathbb{R}^{n_y}$  denote respectively the state, input, and output vector.

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 size that grows linearly.

In order to arrive at a directly solvable optimization problem, an iteration of the MPC method based on the MLD model can be written as an MILP where some of the decision variables belong to an integer domain (in this case solely binary) and some to a real domain. The following statements summarize the conversion (adapted from [30], [32]):

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

$$\begin{cases} f(x) \leq c + (M - c)(1 - \delta) \\ f(x) \geq c(1 - \delta) + \epsilon + (m - \epsilon)\delta, \end{cases} \quad (28)$$

$$\delta = \delta_1 \delta_2 \Leftrightarrow \begin{cases} -\delta_1 + \delta \leq 0 \\ -\delta_2 + \delta \leq 0 \\ \delta_1 + \delta_2 - \delta \leq 1, \end{cases} \quad (29)$$

$$z = \delta f(x) \Leftrightarrow \begin{cases} z \leq M\delta \\ z \geq m\delta \\ z \leq f(x) - m(1 - \delta) \\ z \geq f(x) - M(1 - \delta). \end{cases} \quad (30)$$

Here, binary dummy variables (denoted by  $\delta \in \{0, 1\}$ ) are introduced to indicate whether a certain region applies that is associated with one of the affine pieces of the PWA function. The constants  $m, M$  denote respectively a lower and upper bound of a function  $f(x)$  over a set of variables  $x$  that is bounded in case of an affine function, where we can assume that  $x(k), u(k)$ , and  $y(k)$  are bounded. Finally,  $c$  denotes an arbitrary constant and the small constant  $\epsilon$  denotes the machine precision (used to turn a strict inequality into a non-strict inequality that fits the MILP framework).

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  $f_1 \leq f_2$  and  $\delta = 0$  otherwise, according to the constraints (28). The latter expression of  $f$  should

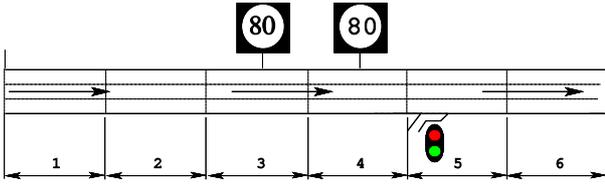


Fig. 6. Set-up of the case study

again be written  $f = z_1 + f_2 - z_2$  with auxiliary variables  $z_i = f_i \delta_i$ ,  $i = 1, 2$ , according to the constraints (30). If in addition a third term  $f_3$  arises in the PWA equation, such that  $f = f_1 \delta_1 + f_2 \delta_2 + f_3(1 - \delta_1)(1 - \delta_2)$ , where

$$\begin{aligned} \delta_1 = 1 &\Leftrightarrow f_1 \leq f_2 \text{ and } f_1 \leq f_3 \\ \delta_2 = 1 &\Leftrightarrow f_2 \leq f_1 \text{ and } f_2 \leq f_3, \end{aligned} \quad (31)$$

this gives rise to a multiplication of binary variables,  $\delta_1 \delta_2$ , for which the constraints in (29) are needed.

Finally, as pointed at in Section III, it is needed to transform the norm  $\|s\|_1$ ,  $s \in \mathbb{R}^n$  in a similar manner. Here, an optimization objective  $\min_{s \in \mathbb{R}^n} \|s\|_1 = \sum_{i=1}^n |s_i|$  can be substituted by the following linear expressions, as can be easily verified:

$$\min_{s, z \in \mathbb{R}^n} \sum_{i=1}^n z_i \text{ subject to } -z \leq s \leq z. \quad (32)$$

Likewise, an  $\infty$ -norm can be transformed into a linear problem, and alternatively a 2-norm in the penalty term (20) can be substituted by a quadratic objective.

All in all, we now end up with an MILP or MIQP formulation, which belongs to the class of NP-hard problems, i.e., it is generally accepted that no polynomial time algorithm exists that solves the problem to optimality [14]. However, for relatively small problem instances efficient solvers are available that are based on e.g., column generation techniques, branch and bound, or cutting plane methods [33], [34]. Concerning the implications of this MILP conversion for the ease of computation of the final MPC problem, it thus remains of interest to keep the number of regions and therefore the number of additional binary variables small. However, a division using fewer regions may further increase the approximation error. Therefore, an appropriate balance between computational speed and approximation accuracy has to be found, which has already been considered in the approximation of the individual model equations in Section IV-B and which will be evaluated next.

## VI. CASE STUDY

In this section, a benchmark study from the literature [17] is adopted in order to facilitate the comparison of results. Here, simulation results from the original nonlinear METANET model with emissions are compared with the MLD-MPC approach based on the PWA models. Both computation time and performance w.r.t. the optimization objective are analyzed. In addition, a more diverse demand profile is considered in order to evaluate the performance of the MLD-MPC approach when experiencing larger variations in the traffic states.

### A. Set-up

The freeway setting used in the simulations is depicted in Fig. 6: 6 segments are considered of which segments 3 and 4 are subject to variable speed limits and a metered on-ramp is placed between segments 4 and 5. Additionally, an upper bound is added constraining the queue length  $w_{o_2}(k)$  to 100 veh. The demand profile is depicted in Fig. 7, where the mainstream origin and on-ramp have a capacity of 4000 and 2000 veh/h respectively. Parameter values can be found in Table I. Further, the temporal aspect of the vehicular emissions and fuel consumption is considered (see Section II-B). Finally, conform the analysis in [17], a prediction and control horizon of respectively  $N_p = 7$ ,  $N_c = 5$  is found to lead to the best results for demand profile 1. We simulate the freeway dynamics for a simulation horizon corresponding to 2.5 h with the controller sampling time  $T_c = 1$  min.

### B. Results

Table II shows the TTS, total emission values, and mean CPU times obtained for three scenarios, i.e., minimizing (S1) TTS only ( $c_1 = 1, c_2 = c_3 = 0$ ) (S2) TTS and HC emissions ( $c_1 = 1, c_2 = 0.25, c_3 = 0$ ) (S3) TTS and  $\text{NO}_x$  emissions as well as fuel consumption (FC) ( $c_1 = 1, c_2 = 0.4, c_3 = 0$ ). These scenarios were chosen to evaluate our approach for different problem sizes. Additionally, the standard deviation ( $\sigma$ ) and the minimum and maximum CPU times are provided. The values obtained in the uncontrolled case are used to normalize the performance criteria. Further, the percentage difference is given between the objective function values obtained under nonlinear control or while using MLD-MPC.

As can be seen from Table II, for scenario S1 one run of MLD-MPC optimization took on average over the 150 simulation time steps 6 s., as compared to approximately 47 s, which is the time required for 10 iterations of the nonlinear solver<sup>1</sup>. Further, it should be noted that whereas the MILP is solved to optimality at once, nonlinear MPC requires an a-priori unknown number of iterations before convergence is reached. Therefore, 10 iterations should be seen as a lower bound: in scenario S2 and S3 at least 15 runs were required to obtain good results, where the number of required runs is based on the variance in the solution returned by consecutive optimization runs for different random, initial starting points [17]. A certain number of runs may also be required in order to obtain a feasible solution in case of constraints. Finally, it can be seen that the computational advantage of applying MLD-MPC is of similar order for all three cases.

As for the deviations of the objective function values when comparing MLD-MPC to nonlinear MPC, the absolute emission values can be said to differ relatively little. On the other hand, the values of the TTS have a deviation of roughly 10% when compared to nonlinear MPC. This reduction in accuracy results from the trade-off with the improvements in computational efficiency. Further, as expected, the TTS values increase in both the nonlinear and MLD-MPC approaches

<sup>1</sup>The CPU times were obtained adopting the Tomlab CPLEX and fmincon environment within 32-bit Matlab 7.9.0 (R2009b) on a Linux PC with a 3GHz Intel Core Duo processor and 3.7Gb RAM.

TABLE II  
COMPARISON OF THE DIFFERENT SCENARIOS FOR DEMAND PROFILE 1

(S1) TTS	TTS (veh-h)	TNO <sub>x</sub> (kg)	TCO (kg)	THC (kg)	TFC (l)	CPU (s) [σ, min, max]
Uncontrolled	1.463·10 <sup>3</sup>	6.683	59.49	4.103	4.654·10 <sup>3</sup>	–
Nonlinear MPC	1.268·10 <sup>3</sup>	6.988	60.45	3.916	4.348·10 <sup>3</sup>	46.71 (10 runs) [25.0, 9.56, 131]
MLD-MPC (% diff.)	1.392·10 <sup>3</sup> (9.8%)	6.814 (-2.5%)	59.92 (-0.9%)	4.122 (5.3%)	4.526·10 <sup>3</sup> (4.1%)	5.829 (-87.5%) [16.2, 0.2137, 186]

(S2) TTS + HC	TTS (veh-h)	THC (kg)	CPU (s) [σ, min, max]
Uncontrolled	1.463·10 <sup>3</sup>	4.103	–
Nonlinear MPC	1.287·10 <sup>3</sup>	3.857	62.05 (15 runs) [29.7, 30.3, 302]
MLD-MPC (% diff.)	1.403·10 <sup>3</sup> (9.1%)	4.045 (4.9%)	14.5 (-76.7%) [29, 1.7, 231]

(S3) TTS + TNO <sub>x</sub> + TFC	TTS (veh-h)	TNO <sub>x</sub> (kg)	TFC (l)	CPU (s) [σ, min, max]
Uncontrolled	1.463·10 <sup>3</sup>	6.683	4.539·10 <sup>3</sup>	–
Nonlinear MPC	1.380·10 <sup>3</sup>	6.959	4.353·10 <sup>3</sup>	80.44 (15 runs) [56.5, 24.2, 248]
MLD-MPC (% diff.)	1.441·10 <sup>3</sup> (4.4%)	6.805 (-2.2%)	4.498·10 <sup>3</sup> (3.3%)	29.34 (-63%) [29.7, 0.221, 183]

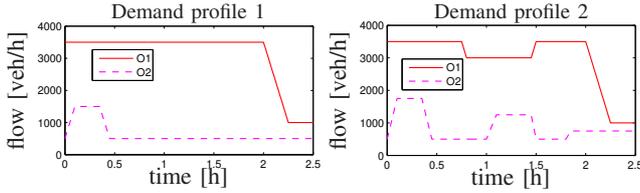


Fig. 7. On-ramp and mainstream demand profiles

when emission control is applied, where the largest increase is in the most comprehensive scenario S3. For this scenario it should be noted that the total NO<sub>x</sub> emissions were not reduced w.r.t. the uncontrolled case for both the MLD and nonlinear MPC approach. However, this value as well as the value for TFC did get reduced in comparison to the values that were obtained under the TTS minimization objective of S1. In order to further decrease the value, the weights for the NO<sub>x</sub> reduction in the optimization objective could be adjusted. Here it should be kept into account that due to the multi-objective nature of some terms may have opposing effects on the minimization of other elements in the objective function. In particular, it should be noted that in scenario S2 and S3, the approximated emissions are directly incorporated in the objective functions in case of MLD-MPC.

Further, also under the more diverse, second demand profile depicted in Fig. 7, a reduction in computational speed can be observed, while the improvement in TTS is very close to the reduction in TTS while adopting nonlinear MPC. Here, we only applied ramp metering for both the MLD-MPC and nonlinear MPC approach. When taking into account the behavior of the traffic network while using the PWA approximated model, as depicted in Fig. 9, it can be said that the behavior of the traffic flow, density and speed over the simulation horizon is similar with the nonlinear case in Fig. 8. The uncontrolled behavior is plotted in Fig. 10, where two large peaks in the queue length at origin 1 can be observed. In the controlled case, these peaks are reduced to a smaller peak of around 100 vehicles around 45 minutes time, and a larger peak around 2 hours. The queue at origin 2 is at its upper bound in both cases, while in the MLD-MPC case, more fluctuations in the queue lengths can be observed. Overall, there is more fluctuation in

TABLE III  
COMPARISON OF TTS AND CPU TIME AND THEIR RELATIVE DIFFERENCES – DEMAND PROFILE 2

(1) TTS	TTS (veh-h)	CPU (s) [σ, min, max]
Uncontrolled	1.716·10 <sup>3</sup>	–
Nonlinear MPC	1.641·10 <sup>3</sup>	8.98 (25 runs) [6.1, 3.4, 43]
MLD-MPC (% diff.)	1.657·10 <sup>3</sup> (1.0%)	1.92 (-78.6%) [2.8, 0.15, 14]

the different variables for MLD-MPC, which is smoothed in the nonlinear MPC case. This difference could be explained by the use of smooth versus nonsmooth, PWA model equations.

Recall that this paper presents a proof-of-concept. To show the performance of the presented method in real-life traffic scenarios, it should be applied to diverse case studies based on real-life situations in which varying demand profiles could be investigated. There, in order to yield the best overall accuracy, one should consider fine-tuning the individual function approximations as well as calibrating the model parameters in the MLD-MPC approach for these real-life scenarios. These will be topics for future research. Finally, it is important to note that instead of taking the optimal decision variables resulting from MLD-MPC as a final control input, these results could also be used as a starting point for nonlinear optimization, which would still yield a faster solution given the gains in computation times while using MLD-MPC.

### C. Computational Efficiency

Concerning the computational efficiency of the MLD method it should be noted that the specific problem structure could be exploited to speed-up the optimization, i.e., computational advantages could result from tuning the solver to the particular problem like by changing the structure of the constraint matrices [35]. An overview of computational efficiency in the solving of MILPs when using different solvers can be found in [36]. Also, as an alternative to nonlinear control, the feasible-direction method proposed in [3] could be used to compare the computational requirements. In particular, results of the designated Advanced Motorway Optimal Control (AMOC) toolbox for the feasible-direction implementation of ramp metering can be found in [37], while integrated ramp metering and speed control is considered in [18], where the

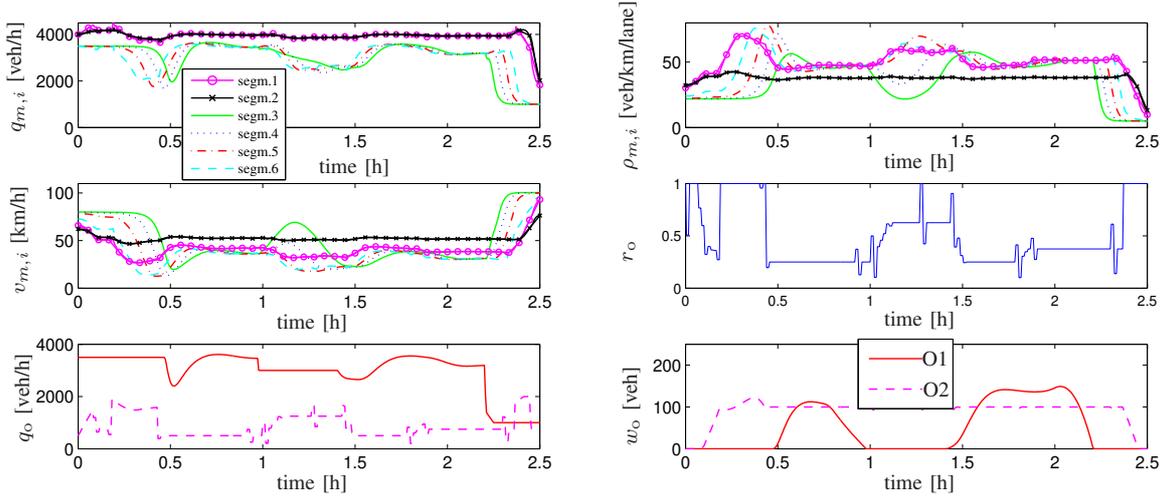


Fig. 8. Simulation results for demand profile 2 - nonlinear MPC

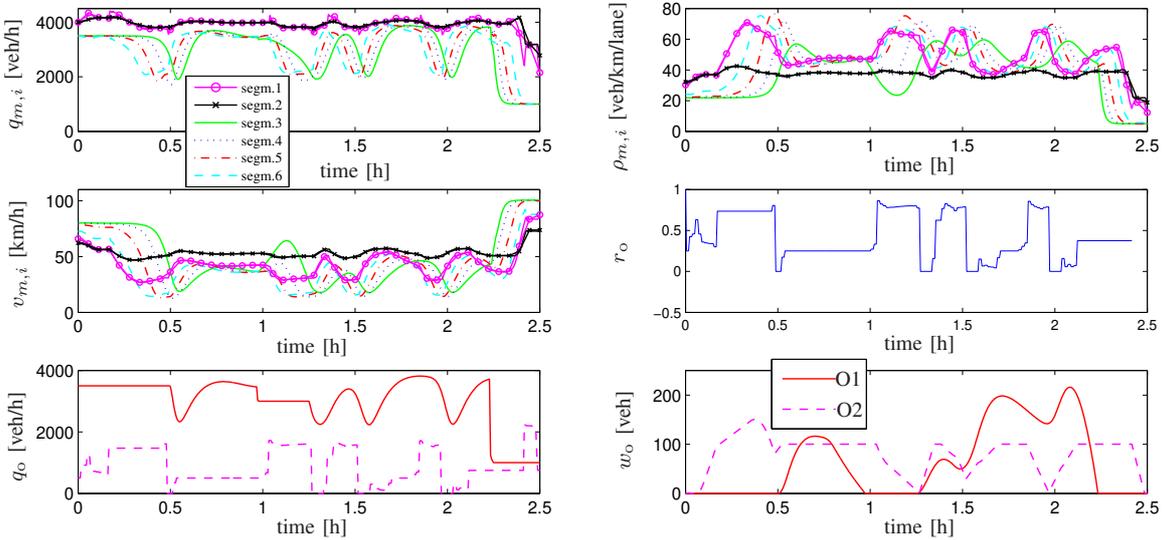


Fig. 9. Simulation results for demand profile 2 - MLD-MPC

method is found to be fast enough for real-life implementation. Here, it should be noted that the constraints on state variables as considered in our case study for the maximum queue length can result in an increase in the required computational time when implemented as a hard constraint; in [18], this constraint is incorporated in the penalty function as a soft constraint. The latter approach however does not in general yield the same traffic behavior; in instances where a rather large penalty is required to keep the queue length below the limit, the effect on the control variables will be substantial. Hence, it would be fruitful to perform an analysis on the impact of optimal control actions on the traffic behavior as well as on computational efficiency of both methods under the same conditions.

## VII. CONCLUSIONS AND FURTHER RESEARCH

A piecewise-affine (PWA) approximation of the nonlinear traffic flow model METANET integrated with the VT-macro model for vehicular emissions and fuel consumption has been proposed in order to deal with the computational complexity of the original nonlinear nonconvex model-based traffic control approach, which is currently hindered from application in real-life traffic networks due to the required computation time. Several methods to approximate the nonlinear functions appearing in the models have been discussed and the transformation to a ready-to-implement mixed-integer linear optimization programming (MILP) problem has been provided. This method has been tested in a case study with hard state constraints comparing the performance of the mixed-logical dynamic (MLD)-MPC method with the original nonlinear-programming

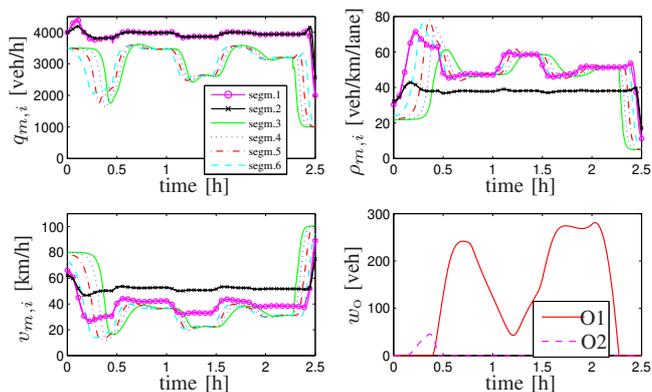


Fig. 10. Uncontrolled behavior under demand profile 2

MPC application w.r.t. the trade-off between computational speed and accuracy, The simulation results when considering two hypothetical demand profiles, showed that MLD-MPC can indeed be applied at an improved computational efficiency at the cost of some deterioration of the control performance.

As further steps to bring the MLD-MPC approach to be applicable in real life, more elaborate case studies could be performed in order to investigate in detail the trade-off between approximation accuracy (i.e., the number of affine pieces or regions) and computational efficiency, in a setting that more closely resembles real-life traffic conditions. Here, an extensive analysis of the effect of the MLD-MPC method on the traffic behavior could be studied under different traffic scenarios. Finally, the MLD-MPC approach could also be applied to different traffic models like higher-order macroscopic models, the discretized Payne model and the cell and link transmission models.

#### ACKNOWLEDGMENTS

This research is supported by the European 7th Framework Network of Excellence “Highly-complex and networked control systems (HYCON2)”.

#### 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] A. Kotsialos, M. Papageorgiou, C. Diakaki, Y. Pavlis, and F. Middelham, “Traffic flow modeling of large-scale motorway networks using the macroscopic modeling tool METANET,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 3, no. 4, pp. 282–292, Dec. 2002.
- [4] S. Hoogendoorn and P. Bovy, “State-of-the-art of vehicular traffic flow modelling,” *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 215, no. 4, pp. 283–303, Aug. 2001.
- [5] S. Zegeye, B. De Schutter, H. Hellendoorn, and E. Breunese, “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, USA, Sep. 2009, pp. 149–154.
- [6] J. Maciejowski, *Predictive Control with Constraints*. Harlow, England: Prentice Hall, 2002.
- [7] J. Rawlings and D. Mayne, *Model Predictive Control: Theory and Design*. Madison, WI, USA: Nob Hill Publishing, 2009.
- [8] 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.
- [9] 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.
- [10] 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.
- [11] 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, USA, Jun.–Jul. 2010, pp. 2284–2289.
- [12] S. Lin, B. De Schutter, Y. Xi, and H. Hellendoorn, “Fast model predictive control for urban road networks via MILP,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 3, pp. 846–856, Sep. 2011.
- [13] A. Ghods, L. Fu, and A. Rahimi-Kian, “An efficient optimization approach to real-time coordinated and integrated freeway traffic control,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 4, pp. 873–884, Dec. 2010.
- [14] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. San Francisco, CA, USA: W.H. Freeman and Company, 1979.
- [15] N. Groot, B. De Schutter, S. Zegeye, and H. Hellendoorn, “Model-based predictive traffic and emission control using PWA models: A mixed-logical dynamic approach,” in *Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems (ITSC 2011)*, Washington D.C., DC, USA, Oct. 2011, pp. –.
- [16] —, “Model-based predictive traffic control: A piecewise-affine approach based on METANET,” in *Proceedings of the 18th IFAC World Congress*, Milano, Italy, Aug. 2011, pp. 10709–10714.
- [17] 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.
- [18] R. Carlson, I. Papamichail, M. Papageorgiou, and A. Messmer, “Optimal mainstream traffic flow control of large-scale motorway networks,” *Transportation Research Part C*, vol. 18, no. 2, pp. 193–212, Apr. 2010.
- [19] K. Ahn, A. Trani, H. Rakha, and M. Van Aerde, “Microscopic fuel consumption and energy emission models,” in *Proceedings of the 78th Annual Meeting of the Transportation Research Board*, Washington DC, USA, January 1999.
- [20] E. Sontag, “Nonlinear regulation: The piecewise linear approach,” *IEEE Transactions on Automatic Control*, vol. 26, no. 2, pp. 346–357, 1981.
- [21] 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.
- [22] S. Azuma, J. Imura, and T. Sugie, “Lebesgue piecewise affine approximation of nonlinear systems,” *Nonlinear Analysis: Hybrid Systems*, vol. 4, no. 1, pp. 92–102, 2010.
- [23] R. Fletcher, *Practical Methods of Optimization, Volume 1: Unconstrained Optimization*. Chichester, UK: John Wiley & Sons, 1980.
- [24] E. Bredensteiner and K. Bennett, “Multicategory classification by support vector machines,” *Computational Optimizations and Applications*, vol. 12, no. 1–3, pp. 53–79, 1999.
- [25] V. Vapnik, *Statistical Learning Theory*. New York, NY, USA: John Wiley and Sons, 1998.
- [26] G. Fung and O. Mangasarian, “Proximal support vector machine classifiers,” in *Proceedings of KDD-2001: Knowledge Discovery and Data Mining*, F. Provost and R. Srikant, Eds. San Francisco, CA, USA: Association for Computing Machinery, 2001, pp. 77–86.
- [27] G. Ferrari-Trecate, “Hybrid identification toolbox (HIT),” [http://www-roq.inria.fr/who/Giancarlo.Ferrari-Trecate/HIT\\_toolbox.html](http://www-roq.inria.fr/who/Giancarlo.Ferrari-Trecate/HIT_toolbox.html), 2005.
- [28] M. Kvasnica, P. Grieder, and M. Baotić, “Multi-parametric toolbox (mpt),” <http://control.ee.ethz.ch/~mpt/>, 2004.
- [29] M. Kvasnica, A. Szűcs, and M. Fikar, “Automatic derivation of optimal piecewise affine approximations of nonlinear systems,” in *Proceedings of the 18th IFAC World Congress*, Milano, Italy, Aug.–Sep. 2011, pp. 8675–8680.
- [30] H. Williams, *Model Building in Mathematical Programming*, 3rd ed. New York, NY, USA: Wiley, 1993.

- [31] A. May, *Traffic Flow Fundamentals*. Englewood Cliffs, New Jersey: Prentice-Hall, 1990.
- [32] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407–427, 1999.
- [33] A. Atamtürk and M. Savelsbergh, "Integer-programming software systems," *Annals of Operations Research*, vol. 140, no. 1, pp. 67–124, Nov. 2005.
- [34] 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.
- [35] O. Saeki and K. Tsuji, "A method for solving a class of mixed integer linear programming problem with block angular structure," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, Hammamet, Tunisia, October 2002, pp. 308–313.
- [36] R. Bixby, M. Fenelon, Z. Gu, E. Rothberg, and R. Wunderling, "MIP: Theory and practice - Closing the gap," in *System Modelling and Optimization: Methods, Theory, and Applications*, ser. IFIP Conference Proceedings, M. J. D. Powell and S. Scholtes, Eds., vol. 174. Cambridge, UK: Kluwer Academic Publishers, 2000, pp. 19–50.
- [37] A. Kotsialos and M. Papageorgiou, "Efficiency and equity properties of freeway network-wide ramp metering with AMOC," *Transportation Research Part C*, vol. 12, no. 6, pp. 401–420, 2004.