A Hybrid Approach to Traction Control

Francesco Borrelli[1], Alberto Bemporad[1,3], Michael Fodor[2], and Davor Hrovat[2]

[1] Automatic Control Laboratory, ETH, CH-8092 Zurich, Switzerland
Phone: +41 1 632-4158, Fax: +41 1 632-1211
{borrelli,bemporad}@aut.ee.ethz.ch
[2] Ford Research Laboratories,
Dearborn, MI 48124
Phone: +1 313 594-2958, Fax: +1 313 322-5562
{mfodor1,dhrovat}@ford.com
[3] Dip. Ingegneria dell'Informazione, Università di Siena
Phone: +39 0577 234-631, Fax: +39 0577 234-632
bemporad@dii.unisi.it

Abstract. In this paper we describe a hybrid model and an optimization-based control strategy for solving a traction control problem currently under investigation at Ford Research Laboratories. We show through simulations on a model and a realistic set of parameters that good and robust performance is achieved. Furthermore, the resulting optimal controller is a piecewise linear function of the measurements that can be implemented on low cost control hardware.

1 Introduction

For more than a decade advanced mechatronic systems controlling some aspects of vehicle dynamics have been investigated and implemented in production [13]. Among them, the class of traction control problems is one of the most studied. Traction controllers are used to improve a driver's ability to control a vehicle under adverse external conditions such as wet or icy roads. By maximizing the tractive force between the vehicle's tire and the road, a traction controller prevents the wheel from slipping and at the same time improves vehicle stability and steerability. In most control schemes the wheel slip, i.e., the difference between the normalized vehicle speed and the speed of the wheel is chosen as the controlled variable. The objective of the controller is to maximize the tractive torque while preserving the stability of the system. The relation between the tractive force and the wheel slip is nonlinear and is a function of the road condition [2]. Therefore, the overall control scheme is composed of two parts: a device that estimates the road surface condition, and a traction controller that regulates the wheel slip at any desired value. Regarding the second part, several control strategies have been proposed in the literature mainly based on sliding-mode controllers, fuzzy logic and adaptive schemes [5, 14, 4, 19, 20, 17, 2, 18]. Such control schemes are motivated by the fact that the system is nonlinear and uncertain.

The presence of nonlinearities and constraints on one hand, and the simplicity needed for real-time implementation on the other, have discouraged the

M.D. Di Benedetto, A. Sangiovanni-Vincentelli (Eds.): HSCC 2001, LNCS 2034, pp. 162–174, 2001.

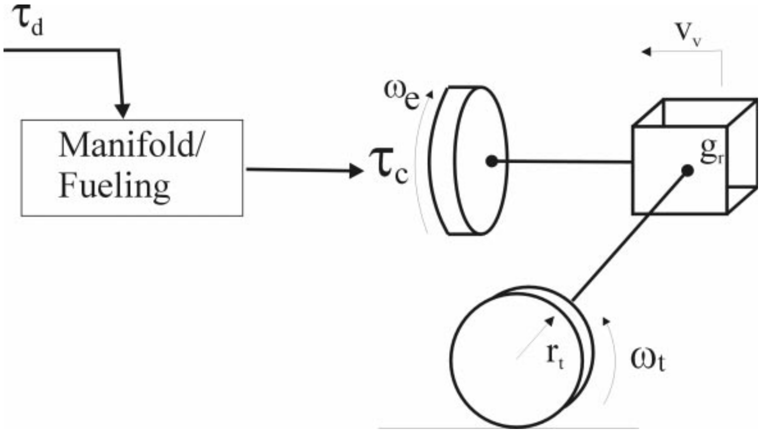


Fig. 1. Simple vehicle model

design of optimal control strategies for this kind of problem. Recently we proposed a new framework for modeling hybrid systems [8] and an algorithm to synthesize piecewise linear optimal controllers for such systems [6]. In this paper we describe how the hybrid framework [8] and the optimization-based control strategy [6] can be successfully applied for solving the traction control problem in a systematic way. We show, through simulations on a simplified model and for a set of parameters provided by Ford Research Laboratories, that good and robust performance can be achieved. Furthermore, the resulting optimal controller consists of a piecewise linear function of the measurements, that can be easily implemented.

A mathematical model of the vehicle/tire system is introduced in Section 2. The hybrid modeling and the optimal control strategy are discussed in Sections 2.1 and 3, respectively. In Section 4 we derive the piecewise affine optimal control law for traction control and present some simulation results.

2 Vehicle Model

The model of the vehicle used for the design of the traction controller is depicted in Figure 1, and consists of the equations

$$\begin{pmatrix} \dot{\omega}_e \\ \dot{v}_v \end{pmatrix} = \begin{pmatrix} -\frac{b_e}{J'_e} & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \omega_e \\ v_v \end{pmatrix} + \begin{pmatrix} \frac{1}{J'_e} \\ 0 \end{pmatrix} \tau_c + \begin{pmatrix} -\frac{1}{J'_e g_r} \\ -\frac{1}{m_v r_t} \end{pmatrix} \tau_t \tag{1}$$

with

$$\dot{\tau}_c(t) = -k_i \tau_c(t) + k_i \tau_d(t - \tau_f) \tag{2}$$

where the involved physical quantities and parameters are described in Table 1.

Table 1. Physical quantities and parameters of the vehicle model

ω_e	Engine speed	r_t	Tire radius
v_v	Vehicle speed	τ_c	Actual combustion torque
J_e'	Combined engine/wheel inertia	τ_d	Desired combustion torque
b_e	Engine damping	τ_t	Frictional torque on the tire
g_r	Total driveline gear ratio between ω_e and v_v	μ	Road coefficient of friction
m_v	Vehicle mass	τ_f	Fueling to combustion pure delay period
$\Delta\omega$	Wheel slip		

The frictional torque τ_t is approximated as a piecewise linear function of the slip $\Delta\omega$ and of the road coefficient of friction μ

$$\tau_t(\Delta\omega, \mu) = \begin{cases} k_1^i \Delta\omega \text{ if } \Delta\omega \leq \Delta\omega_b^i \\ k_2^i \Delta\omega \text{ if } \Delta\omega > \Delta\omega_b^i \end{cases} \text{for} \quad \mu_i \leq \mu \leq \mu_{i+1} \quad i = 0, \ldots, N \qquad (3)$$

as depicted in Figure 2(a).

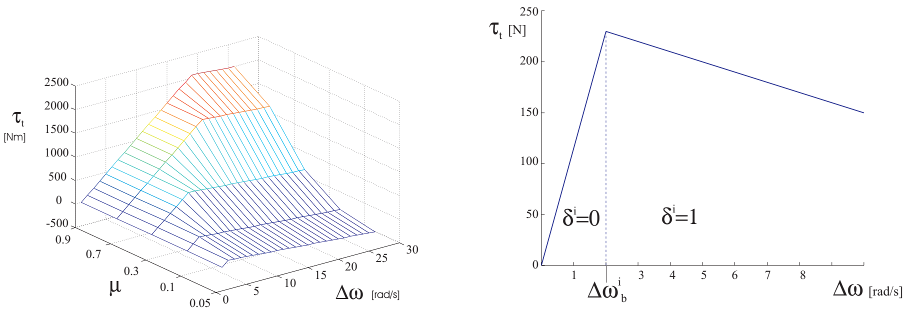
Model (1) contains two states for the mechanical system downstream of the manifold/fueling dynamics. The first equation represents the wheel dynamics under the effect of the combustion torque and of the traction torque, while the second one describes the longitudinal motion dynamics of the vehicle. In addition to the mechanical equations (1) the air intake and fueling model (2) also contributes to the dynamic behaviour of the overall system. For simplicity, the intake manifold dynamics is modeled as a first order filter and the fueling combustion delay is modeled as a pure delay.

2.1 Discrete-Time Hybrid Model

Hybrid systems provide a unified framework for describing processes evolving according to continuous dynamics, discrete dynamics, and logic rules [1,16,10,3]. The interest in hybrid systems is mainly motivated by the large variety of practical situations, for instance real-time systems, where physical processes interact with digital controllers. Several modeling formalisms have been developed to describe hybrid systems [12,15], among them the class of Mixed Logical Dynamical (MLD) systems introduced by Bemporad and Morari [8]. Examples of real-world applications that can be naturally modeled within the MLD framework are reported in [7,8,9]. The language HYSDEL (HYbrid Systems DEscription Language) was developed in [21] to obtain MLD models from of a high level textual description of the hybrid dynamics.

The model obtained in Section 2 is transformed into an equivalent discrete-time MLD model through the following steps:

1. Discretize the model (1)–(3) with sampling time $T_s = 20$ ms;
2. Introduce an auxiliary logic variable δ^i for each interval $[\mu_i, \mu_{i+1}]$ whose value can be 1 or 0 depending on the value of the slip $\Delta\omega$, as shown in Figure 2(b).



(a) Full model

(b) Piecewise linear model of the tire torque τ_t with $\mu \in (\mu_i, \mu_{i+1})$

Fig. 2. Model of the tire torque τ_t as a function of the slip $\Delta\omega$ and road coefficient adhesion μ

Remark 1. In the sequel we will use a simplified model where the slopes $k_1^1 = k_1^2 = \ldots = k_1^N$ and $k_2^1 = k_2^2 = \ldots = k_2^N$, while the breakpoints ω_b^i in (3) are allowed to be different. In this case the number of auxiliary logic variables δ^i reduces from $\log_2 N$ to 1, at the price of a "rougher" model of the nonlinearity.

The resultant MLD system is the following[1]:

$$x(t+1) = Ax(t) + B_1 u(t) + B_2 \delta(t) + B_3 z(t) \tag{4a}$$

$$y(t) = Cx(t) + D_1 u(t) + D_2 \delta(t) + D_3 z(t) \tag{4b}$$

$$E_2 \delta(t) + E_3 z(t) \leq E_1 u(t) + E_4 x(t) + E_5 \tag{4c}$$

where $x \in \mathbb{R}^5$, $(x_1 = \Delta\omega_d,\ x_2 = \omega_e,\ x_3 = v_v,\ x_4 = \tau_t,\ x_5 = \tau_c)$, $u \in \mathbb{R}$, $(u = \tau_d)$, $y \in \mathbb{R}$ $(y = \Delta\omega)$, $\delta \in \{0, 1\}$ and $z \in \mathbb{R}^3$. The variables δ and z are auxiliary variables whose value is determined uniquely by the inequalities (4c) once $x(t)$ and $u(t)$ are fixed [8].

In Figure 4 we compare the evolution of the discrete-time MLD model (4) with the evolution of the continuous time model (1)–(3), depicted in Figure 3, when $\mu = .1$, $\Delta\omega_b = 2$ rad/s and a pulse torque $\tau_d = 50$ Nm is applied to the system. The MLD model (4) captures in discrete time the hybrid behavior of the system satisfactorily.

3 Optimal Control

It is clear from Figure 2(b) that if the slip increases beyond Δw_b, the driving force on the tire decreases considerably and the vehicle cannot speed up as desired. By

[1] The numerical values of the matrices in (4) are reported in the Appendix.

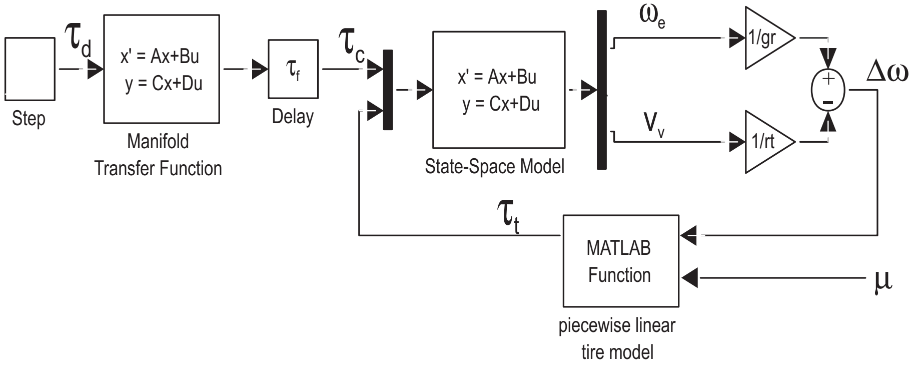


Fig. 3. Simulink scheme of the vehicle model

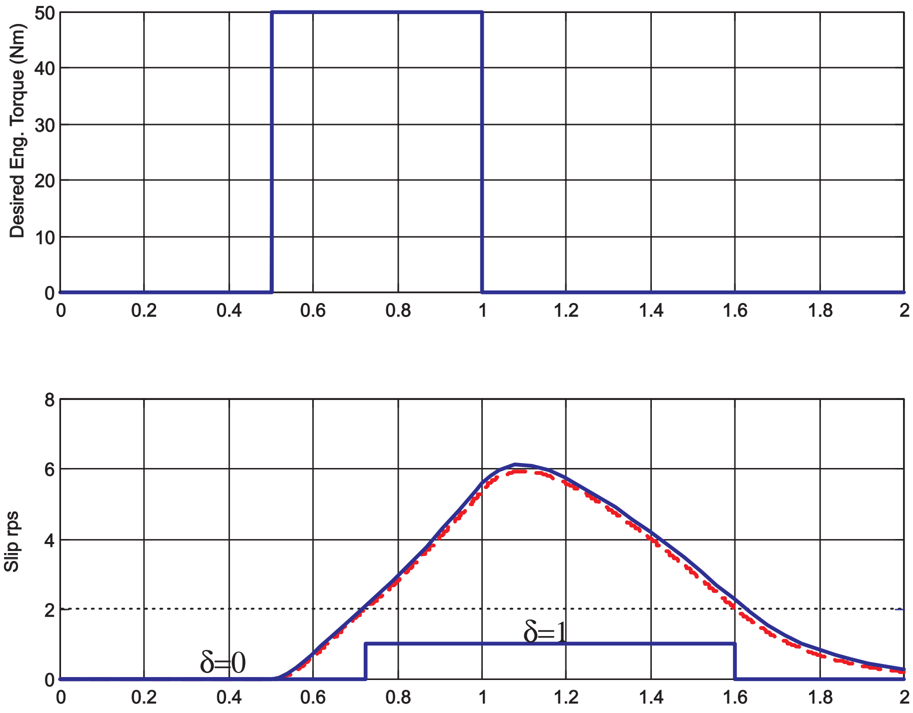


Fig. 4. Continuous time simulation of the Matlab-Simulink block in Figure 3 (solid line), discrete-time simulation of the MLD model (dashed line)

maximizing the tractive force between the vehicle's tire and the road, a traction controller prevents the wheel from slipping and at the same time improves vehicle stability and steerability. The overall control scheme is composed of two parts: a device that estimates the road surface condition, and a traction controller that regulates the wheel slip at any desired value. In this paper we will focus only on the second part, as the first one is available at Ford Research Laboratories.

Once the road coefficient of adhesion μ has been estimated, a desired wheel sleep $\Delta\omega_d^k$ is chosen corresponding to the breakpoints $\Delta\omega_b^k$ in model (3), $\mu \in [\mu_{k-1}, \mu_k]$, where the frictional torque $\tau_t(\Delta\omega)$ on the tire is maximized. Alternatively, to increase the safety of the controller [18] we could avoid operating in the region where the slope of the curve $\tau_t(\Delta\omega)$ is negative, see Figure 2(b), by simply choosing $\Delta\omega_d^k(\mu) < \Delta\omega_b^k \;\; (\mu \in [\mu_{k-1}, \mu_k])$. The control system takes the desired wheel slip $\Delta\omega_d^k$ and measured wheel speed as input and generates the desired engine torque. The following constraints on the torque and its variation need to be satisfied:

$$-20 \text{ Nm} \leq \tau_d \leq 176 \text{ Nm} \tag{5}$$

$$\dot{\tau}_d(t) \leq 2000 \text{ Nm/s} \tag{6}$$

In the sequel we describe how a Model Predictive Controller (MPC) can be designed for the posed traction control problem described. The main idea of MPC is to use the *model* of the plant to *predict* the future evolution of the system. Based on this prediction, at each time step t a certain performance index is optimized under operating constraints with respect to a sequence of future input moves. The first of such optimal moves is the *control* action applied to the plant at time t. At time $t+1$, a new optimization is solved over a shifted prediction horizon. For the traction control problem, at each time step t the following finite horizon optimal control problem is solved:

$$\min_{\{\Delta u_0^{T-1}\}} \sum_{k=0}^{T-1} |\Delta\omega(t+k|t) - \Delta\omega_d(t)| \tag{7}$$

$$\text{subj. to} \begin{cases} \text{MLD dynamics (4)} \\ \tau_{min} \leq u(t+k) \leq \tau_{max}, \;\; k = 0, 1, \ldots, T-1 \\ \Delta\tau_{min} \leq \delta u(t+k) \leq \Delta\tau_{max}, \;\; k = 0, 1, \ldots, T-1 \\ x_{min} \leq x(t+k|t) \leq x_{max}, \;\; k = 1, \ldots, T-1 \end{cases} \tag{8}$$

where $\Delta u_0^{T-1} = \{\delta u(t), \ldots, \delta u(t+T-1)\}$, and "$(t+k|t)$" denotes the predicted value at time $t+k$ based on the state information available at time t. Note that the optimization variables are not the future inputs u_{t+k}, but the variation $\delta u(t+k) = u(t+k) - u(t+k-1)$, which makes it necessary to increase the dimension of the state vector by one to include the previous torque $\tau_d(t-1)$ as a an additional state $x_6(t) = \tau_d(t-1)$.

Problem (7)-(8) can be translated into a mixed integer linear program (MILP) (the minimization of a linear cost function subject to linear constraints where variables can be binary and/or continuous) of the form:

$$\begin{aligned} &\min_{z=\{z_c, z_d\}} f_c^T z_c + f_d^T z_d \\ &\text{subj. to } \; G_c z_c + G_c z_d \leq S + Fx(t) \end{aligned} \tag{9}$$

where $z_c \in \mathbb{R}^l$ and $z_d \in \{0, 1\}^m$.

Given the measurement of the state $x(t)$, problem (9) is solved at each time step, but only the first optimal input $u^*(t) = \tau_d(t-1) + \delta u_0^*$ is implemented as the new command torque $\tau_d(t)$. At the next time step the procedure is repeated starting with the new measurement of the state.

The design of the controller is performed in two steps. First, the MPC controller (7)-(8) based on model (4) is tuned in simulation until the desired performance is achieved. The MPC controller is not directly implementable, as it would require the MILP (9) to be solved on-line, which is clearly prohibitive on standard automotive control hardware. Therefore, for implementation, in the second phase the explicit piecewise linear form of the MPC law (see Section 4.2) is computed off-line by using the multi-parametric mixed integer programming solver presented in [11]. Although the resulting piecewise linear control action is *identical* to the MPC designed in the first phase, the on-line complexity is reduced to the simple evaluation of a piecewise linear function.

4 Controller Design

The only parameter of the controller (7)-(8) to be tuned is the horizon length T. By increasing the prediction horizon the controller performance improves, but at the same time the number of constraints in (8) increases. As will be explained in Section 4.2 the complexity of the final piecewise linear controller increases with the number of constraints in (8). Therefore, tuning T amounts to finding the smallest T which leads to a satisfactory closed-loop behaviour.

4.1 Simulations

We simulate the closed-loop composed of the traction controller (7)-(8) and model (1)-(2), where the piecewise linear function modeling the frictional torque on the tire τ_t (3) is replaced by a more accurate nonlinear model provided by Ford, see Figure 5. The actual combustion torque τ_c is estimated from the two measurements ω_e and v_v by using an extended Kalman Filter designed for the PWA model.

The controlled system is simulated with an initial vehicle speed of zero. The intake manifold state τ_c is set to a large torque value, namely $\tau_c(0) = 100$ Nm, in order to approximate a wide-open throttle launch from a standstill. In Figure 6 we simulate a straight-ahead driving with a transition at time $t^* = 2$ s from a high coefficient of friction $\mu = 0.9$, and $\Delta\omega_d = 18$ rad/s (cement pavement) to a low one $\mu = 0.1$, $\Delta\omega_d = 2$ rad/s (dry ice). The simulations show the good performance of the controller despite the large mismatch between the nonlinear model of the frictional torque model and the piecewise linearized one.

The following controllers are simulated:

- Controller 1 (Figure 6(a)): T=3;
- Controller 2 (Figure 6(b)): T=9;

The Simulink control diagram used for simulation is shown in Figure 5.

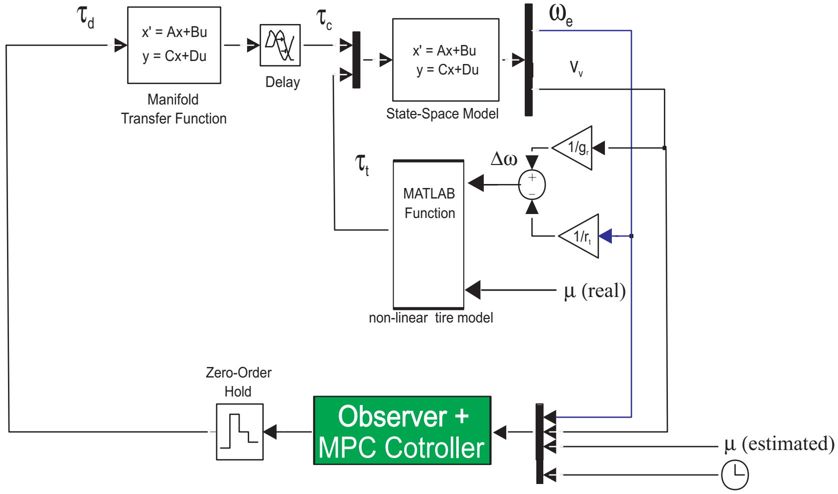


Fig. 5. Simulink diagram of the closed-loop control system

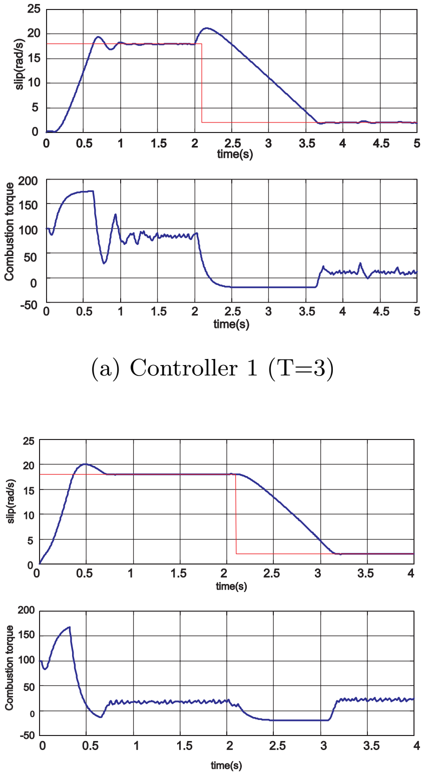
4.2 Explicit Controller

Once the controller has been tuned, the explicit piecewise linear form of the MPC law is computed off-line by using a multiparametric mixed integer linear programming (mp-MILP) solver, according to the approach of [6]. Rather than solving the MILP (9) *on-line* for the given current state $x(t)$, the idea is to use the mp-MILP solver to compute *off-line* the solution of the MILP (9) for all the states $x(t)$ within a given polyhedral set.

As shown in [6], the explicit solution $z^*(x(t))$ of (9) is a piecewise affine function of $x(t)$. Therefore, the model predictive controller is also available explicitly, as the optimal input $\delta u(t)$ consists simply of a component of $z^*(x(t))$. As a result, the state space is partitioned into polyhedral sets, where an affine control law is defined in each polyhedron.

We remark that for any given state $x(t)$ the on-line solution of MPC and the explicit off-line solution provide the same result. Therefore, a good design strategy consists of tuning the MPC controller using simulation and on-line optimization, and then to convert the controller to its piecewise affine explicit form. The explicit controller will behave in exactly the same way at much lower computation cost.

The result of the mp-MILP solver is a list of N records. The i-th record contains the constraints defining the i-th polyhedral region $H(i)x \leq K(i)$, $H(i) \in \mathbb{R}^{m_i \times n}$, and the corresponding i-th gain $\delta u = F(i)x + G(i)$. The control law can be implemented on-line in the following simple way: (1) determine the i-th region that contains the actual vector state $x(t)$ (measured and/or estimated); (2) compute $\delta u(t) = F(i)x(t) + G(i)$, according to the corresponding i-th control law.



(a) Controller 1 (T=3)

(b) Controller 2 (T=9)

Fig. 6. Closed-loop simulation of Controller 1 and Controller 2. Straight-ahead driving with a transition at time $t^* = 2$ s from a high coefficient of friction $\mu = 0.9$, and $\Delta\omega_d = 18$ rad/s (cement pavement) to a low one $\mu = 0.1$, $\Delta\omega_d = 2$ rad/s (dry ice)

In Figure 6(a) we report the performance achieved with two explicit MPC controllers, obtained by solving the mp-MILP problem for the box $X_{min} \leq x(t) \leq X_{max}$, $X_{min} = [0, 0, 0, -20, -20, -40]$ and $X_{max} = [20, 150, 10, 100, 300, 40]$:

- Controller 1 : T=3, Number of regions $N = 76$, maximum number of constraints per region $M = \max_{i=1,...,N} m_i = 13$;
- Controller 2: T=9, Number of regions $N = 243$, maximum number of constraints per region $M = 25$.

As an example, we report only the first and last region of Controller 1:

$$\delta u = \left\{ \begin{array}{l} -\,40.0000 \\ \\ \text{if} \begin{bmatrix} 0.0 & 0.0 & -0.0 & 0.0 & 0.0 & -0.05 \\ 0.0 & 0.01 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.01 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.02 \\ -1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & -1.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & -1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & -0.05 & 0.0 & 0.0 \\ -6.13 & 0.47 & -22.70 & -0.02 & 0.14 & 0.02 \\ 0.0 & 0.0 & -0.0 & 0.0 & 0.0 & -0.05 \\ 15.83 & -1.22 & 58.65 & 0.03 & -0.24 & -0.03 \\ -8.0 & 1.23 & -59.26 & -0.01 & 0.07 & 0.0 \\ -31.57 & 2.43 & -116.94 & -0.05 & 0.47 & 0.00 \\ 12.25 & -0.94 & 45.38 & 0.03 & -0.26 & -0.02 \end{bmatrix} x \leq \begin{bmatrix} -1.0 \\ 1.0 \\ 1.0 \\ 1.0 \\ 1.0 \\ 0.0 \\ 0.0 \\ 1.0 \\ 1.0 \\ -1.0 \\ 1.0 \\ 0.0 \\ -1.0 \\ -1.0 \end{bmatrix} \\ \text{(Region \#1)} \\ \\ \vdots \\ \\ \begin{bmatrix} 368.11 \\ -28.34 \\ 1363.38 \\ 0.85 \\ -7.13 \\ -1.00 \end{bmatrix}^T x + 11.59 \\ \\ \text{if} \begin{bmatrix} 43.7456 & -3.3676 & 162.0209 & 0.1011 & -0.8468 & -0.0 \\ -7.8681 & 0.6057 & -29.1410 & -0.0153 & 0.1391 & 0.0178 \\ -6.1306 & 0.4719 & -22.7058 & -0.0180 & 0.1364 & 0.0215 \\ -8.0 & 1.2317 & -59.2593 & -0.0068 & 0.0697 & -0.0 \\ 0.0 & 0.0067 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0100 & 0.0 & 0.0 \\ -1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & -1.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & -1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & -0.0500 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & -0.05 \end{bmatrix} x \leq \begin{bmatrix} 1.0 \\ -1.0 \\ 1.0 \\ 0.0 \\ 1.0 \\ 1.0 \\ 1.0 \\ 0.0 \\ 0.0 \\ 1.0 \\ 1.0 \end{bmatrix} \\ \text{(Region \#76)} \end{array} \right. \tag{10}$$

In Figure 7 a zoomed section of the control law associated with Controller 1 is shown. The section is obtained by fixing the torque $\tau_c = 20$, the desired slip $\varDelta\omega_d = 2$, the friction torque $\tau_t = 80$, and the previous input $\tau_d(t-1) = 20$. Note that the southeast corner is not feasible because it corresponds to a negative slip.

5 Conclusion

In this paper we described a hybrid model and an optimization-based control strategy for a traction control problem. We showed, through simulations on a model and a realistic set of parameters from Ford Research Laboratories, that good and robust performance is achieved. Furthermore, the resulting optimal controller is a piecewise linear function of the measurements that can be easily implemented on low cost hardware. In order to ease the implementation of the controller, the number of regions in the piecewise linear law should be reduced. One possible way is to exploit reachability analysis for hybrid systems in order to remove regions which are never entered, for all the operating conditions within a realistic set. At the same time, for complex piecewise linear partitions, we are developing efficient forms of implementation that greatly reduce the number of regions to be stored by exploiting properties of multiparametric linear programming.

Acknowledgments. We thank Manfred Morari for fruitful discussions and his helpful comments on the original manuscript.

6 Appendix

Below we report the numerical values of the matrices in (4) obtained by using the tool HYSDEL. See `http://www.aut.ee.ethz.ch/~hybrid/FordExample.html`

$$A = \begin{bmatrix} 1&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0&0\\ 0&0&0&0&0.819&0.181\\ 0&0&0&0&0&1 \end{bmatrix},\ B_1 = \begin{bmatrix} 0\\0\\0\\0\\0.18127\\1 \end{bmatrix},\ B_2 = \begin{bmatrix} 0\\0\\0\\0\\0\\0 \end{bmatrix},\ B_3 = \begin{bmatrix} 0&0&0\\ 1&0&0\\ 0&1&0\\ 0&0&1\\ 0&0&0\\ 0&0&0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0&0&0&1&0&0 \end{bmatrix},\ D_1 = \begin{bmatrix} 0 \end{bmatrix},\ D_2 = \begin{bmatrix} 0 \end{bmatrix},\ D_3 = \begin{bmatrix} 0&0&0 \end{bmatrix}$$

$$E_1 = \begin{bmatrix} 0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0 \end{bmatrix},\ E_2 = \begin{bmatrix} 100\\-700\\1400\\200\\6000\\1400\\200\\6000\\-1400\\-200\\-6000\\-1400\\-200\\-6000 \end{bmatrix},\ E_3 = \begin{bmatrix} 0&0&0\\ 0&0&0\\ -1&0&0\\ 0&-1&0\\ 0&0&-1\\ 1&0&0\\ 0&1&0\\ 0&0&1\\ -1&0&0\\ 0&-1&0\\ 0&0&-1\\ 1&0&0\\ 0&1&0\\ 0&0&1 \end{bmatrix} \tag{11}$$

$$E_4 = \begin{bmatrix} -1&0&-4&0&0&0\\ 1&0&4&0&0&0\\ 0&-1&0&0&0&0\\ 0&0&-1&0&0&0\\ 0&0&0&-1&0&0\\ 0&1&0&0&0&0\\ 0&0&1&0&0&0\\ 0&0&0&1&0&0\\ 0&-1&0&0&0&0\\ 0&0&-1&0&0&0\\ 0&0&0&-1&-1&0\\ 0&1&0&0&0&0\\ 0&0&1&0&0&0\\ 0&0&0&1&1&0 \end{bmatrix},\ E_5 = \begin{bmatrix} 100\\-0.0001\\1400\\200\\6000\\200\\6000\\0\\0\\0\\0\\0\\0 \end{bmatrix}$$

References

1. P.J. Antsaklis. A brief introduction to the theory and applications of hybrid systems. *Proc. IEEE, Special Issue on Hybrid Systems: Theory and Applications*, 88(7):879–886, July 2000.
2. R. Balakrishna and A. Ghosal. Modeling of slip for wheeled mobile robots. *IEEE Trans. Robotics and automation*, 11(1):349–370, February 1995.
3. A. Balluchi, L. Benvenuti, M. Di Benedetto, C. Pinello, and A. Sangiovanni-Vincentelli. Automotive engine control and hybrid systems: Challenges and opportunities. *Proc. IEEE*, 88(7):888–912, 2000.

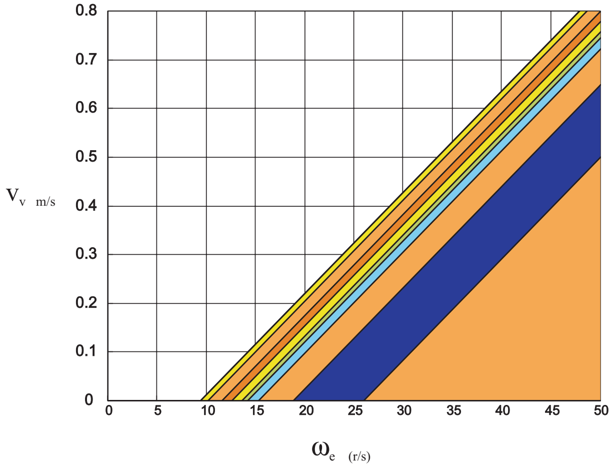


Fig. 7. Section of the explicit solution of Controller 1 for $x_1 = \Delta\omega_d = 2,\ x_4 = \tau_t = 70,\ x_5 = \tau_c = 70,\ x_6 = \tau_d(t-1) = 20$

4. M. Bauer and M. Tomizuka. Fuzzy logic traction controllers and their effect on longitudinal vehicle platoon. *Vehicle System Dynamics*, 25(4):277–303, April 1996.
5. A. Bellini, A. Bemporad, E. Franchi, N. Manaresi, R. Rovatti, and G. Torrini. Analog fuzzy implementation of a vehicle traction sliding-mode control. In *Proc. ISATA 29th International Symposium on Automotive Technology and Automation. Automotive Autom, Croydon, UK*, pages 275–282, 1996.
6. A. Bemporad, F. Borrelli, and M. Morari. Piecewise linear optimal controllers for hybrid systems. In *Proc. American Control Conf.*, Chicago, IL, 2000.
7. A. Bemporad, D. Mignone, and M. Morari. Moving horizon estimation for hybrid systems and fault detection. In *Proc. American Control Conf.*, 1999.
8. A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, March 1999.
9. A. Bemporad and M. Morari. Verification of hybrid systems via mathematical programming. In F.W. Vaandrager and J.H. van Schuppen, editors, *Hybrid Systems: Computation and Control*, volume 1569 of *Lecture Notes in Computer Science*, pages 31–45. Springer Verlag, 1999.
10. M.S. Branicky. *Studies in hybrid systems: modeling, analysis, and control.* PhD thesis, LIDS-TH 2304, Massachusetts Institute of Technology, Cambridge, MA, 1995.
11. V. Dua and E. N. Pistikopoulos. An algorithm for the solution of multiparametric mixed integer linear programming problems. *Annals of Operations Research*, to appear.
12. W.P.M.H. Heemels, B. De Schutter, and A. Bemporad. Equivalence of hybrid dynamical models. *Automatica*, to appear.
13. D. Hrovat. Automotive mechatronic systems. In Cornelius T. Leondes, editor, *Mechatronic Systems Techniques and Applications: Volume 2 - Transportation and Vehicular Systems.* Gordon and Breach Science Publishers, 2000.

14. P. Kachroo and M. Tomizuka. An adaptive sliding mode vehicle traction controller design. In *Proc. IEEE International Conference on Systems, Man and Cybernetics. Intelligent Systems for the 21st Century*, volume 1, pages 777–782, 1995.
15. G. Labinaz, M.M. Bayoumi, and K. Rudie. A survey of modeling and control of hybrid systems. In *13th IFAC World Congress 1996*, 1996.
16. J. Lygeros, C. Tomlin, and S. Sastry. Controllers for reachability specifications for hybrid systems. *Automatica*, 35(3):349–370, 1999.
17. G. F. Mauer. A fuzzy logic controller for an ABS braking system. *IEEE Transaction on Fuzzy Systems*, November 1995.
18. H. S. Tan. *Adaptive and Robust Controls with Application to Vehicle Traction Control*. PhD thesis, Univ. of California, Berkeley, 1988.
19. H. S. Tan and M. Tomizuka. Discrete time controller design for robust vehicle traction. *IEEE Control System Magazine*, 10(3):107–113, April 1990.
20. H.S. Tan and M. Tomizuka. An adaptive sliding mode vehicle traction controller design. In *Proc. American Control Conf.*, volume 2, pages 1856–1861, 1990.
21. F.D. Torrisi, A. Bemporad, and D. Mignone. HYSDEL - A language for describing hybrid systems. Technical Report AUT00-03, ETH Zurich, 2000. `http://control.ethz.ch/~hybrid/hysdel`.