

A Hybrid System Approach to Traction Control

F. Borrelli^{*}, A. Bemporad^{†*}, M. Fodor[‡], D. Hrovat[‡]

^{*}Automatic Control Laboratory, ETH-Zurich, CH-8092 Zurich, Switzerland, borrelli@aut.ee.ethz.ch

[†]Dip. Ingegneria dell'Informazione, University of Siena, 53100 Siena, Italy, bemporad@dii.unisi.it

[‡]Ford Research Laboratories, Dearborn, MI 48124, USA, mfodor1,dhrovat@ford.com

Abstract

In this paper we describe a hybrid model and an optimization-based control strategy for solving a traction control problem. The problem is tackled in a systematic way from modeling to control synthesis and implementation. The model is described first in the language HYSDEL (HYbrid Systems DEscription Language) to obtain a mixed-logical dynamical (MLD) hybrid model of the open-loop system. For the resulting MLD model we design a receding horizon finite-time optimal controller. The resulting optimal controller is converted to its equivalent piecewise affine form by employing multiparametric mixed-integer linear programming techniques, and finally experimentally tested on a car prototype. Experiments show that good and robust performance is achieved in a limited development time, by avoiding the design of ad-hoc supervisory and logical constructs usually required by controllers developed according to standard techniques.

Keywords

Hybrid systems, optimal control, traction control, anti-skid systems, model predictive control, multiparametric programming

I. INTRODUCTION

For more than a decade advanced mechatronic systems controlling some aspects of vehicle dynamics have been investigated and implemented in production [21, 25]. 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 [2, 4, 5, 28, 30, 36–38]. 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 design of optimal control strategies for this kind of problem. Recently we proposed a new framework for modeling hybrid systems [10] and an algorithm to synthesize piecewise linear (indeed, piecewise affine) optimal controllers for such systems [7]. In this paper we describe how the hybrid framework [10] and the optimization-based control strategy [7] can be successfully applied for solving the traction control problem in a systematic way. The language HYSDEL (HYbrid Systems DEscription Language) [40] is first used to describe a linear hybrid model of the open-loop system suitable for control design. Such a model is based on a simplified model and a set of parameters provided by Ford Research Laboratories, and involves piecewise linearization techniques of the nonlinear

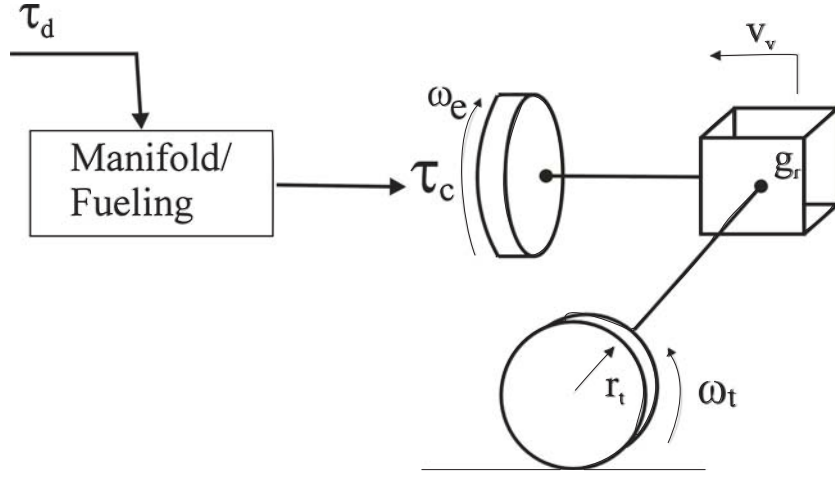


Fig. 1. Simple vehicle model

torque function that are based on hybrid system identification tools [20]. Then, an optimal control law is designed and transformed to an equivalent piecewise affine function of the measurements, that is easily implemented on a car prototype. Experimental results show that good and robust performance is achieved. Preliminary simulation results were reported in the conference paper [13].

A mathematical model of the vehicle/tire system is introduced in Section II. The hybrid modeling and the optimal control strategy are discussed in Sections III and V, respectively. In Section VI we derive the piecewise affine optimal control law for traction control and in Section IX we present the experimental setup and the results obtained.

II. VEHICLE MODEL

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

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

with

$$\tau_c(t) = \tau_d(t - \tau_f) \quad (2)$$

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

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		

TABLE I

PHYSICAL QUANTITIES AND PARAMETERS OF THE VEHICLE MODEL

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, since the actuator will use just the spark advance, the intake manifold dynamics is neglected and the fueling combustion delay is modeled as a pure delay. Both states are indirectly measured through measurements of front and rear wheel speeds: assuming we are modeling a front wheel driven vehicle, ω_e is estimated from the speed of the front wheel, while v_v is estimated from the speed of the rear wheel. The slip $\Delta\omega$ of the car is defined as the difference between the normalized vehicle and engine speeds:

$$\Delta\omega = \frac{v_v}{r_t} - \frac{\omega_e}{g_r}. \quad (3)$$

The frictional torque τ_t is a nonlinear function of the slip $\Delta\omega$ and of the road coefficient of friction μ

$$\tau_t = f_\tau(\Delta\omega, \mu). \quad (4)$$

The road coefficient of friction μ depends on the road-tire conditions, while the function f_τ depends on vehicle parameters such as the mass of the vehicle, the location of the center of gravity and the steering and suspension dynamics [36]. Figure 2(a) shows a typical curve $(\tau_t, \Delta\omega)$ for three different road conditions (ice, snow and pavement).

III. HYBRID SYSTEMS

The mathematical *model* of a system is traditionally associated with differential or difference equations, typically derived from physical laws governing the *dynamics* of the system under consideration. Consequently, most of the control theory and tools have been developed for such systems, in particular for systems whose evolution is described by smooth linear or nonlinear state transition functions. On the other hand, in many applications the system to be controlled is also constituted by parts described by *logic*, such as for instance on/off switches or valves, gears or speed selectors, and evolutions dependent on if-then-else rules. Often, the control of these systems is left to schemes based on heuristic rules inferred from practical plant operation.

Recently, researchers started dealing with *hybrid systems*, namely processes which evolve according to dynamic equations and logic rules, such as hierarchical systems constituted by dynamical components at the lower level, governed by upper level logical/discrete components [1, 14, 15, 29, 34].

The interest in hybrid systems has grown over the last few years not only because of the theoretical challenges, but also because of their impact on applications. Hybrid systems arise in a large number of application areas and are attracting increasing attention in both academic theory-oriented circles as well as in industry, for instance the automotive industry [3, 27]. Our interest is motivated by several clearly discernible trends in the process industries which point toward an extended need for new tools to design control and supervisory schemes for hybrid systems and to analyze their performance.

Several modeling frameworks have been introduced for discrete-time hybrid systems. We will frequently refer to *piecewise affine* (PWA) systems [35]. PWA systems are defined by partitioning the state space into polyhedral regions, and associating with each region a different linear state-update equation

$$\begin{aligned} x(t+1) &= A_i x(t) + B_i u(t) + f_i \\ \text{if } \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} &\in \mathcal{X}_i \triangleq \{ \begin{bmatrix} x \\ u \end{bmatrix} : H_i x + J_i u \leq K_i \}, \end{aligned} \quad (5)$$

where $x \in X \subseteq \mathbb{R}^n$, $u \in U \subseteq \mathbb{R}^m$, $\{\mathcal{X}_i\}_{i=0}^{s-1}$ is a polyhedral partition of the sets of state+input space $\mathbb{R}^{n+m}$. The double definition of the state-update function over common boundaries of sets $\mathcal{X}_i$ (the boundaries will also be referred to as

guardlines) is a technical issue that arises only when the PWA mapping is discontinuous, and can be solved by allowing strict inequalities in the definition of the polyhedral cells in (5). PWA systems can model a large number of physical processes, such as systems with static nonlinearities, and can approximate nonlinear dynamics via multiple linearizations at different operating points.

Furthermore, we mention here linear complementarity (LC) systems [22, 23, 41] and extended linear complementarity (ELC) systems [17], max-min-plus-scaling (MMPS) systems [18], and mixed logical dynamical (MLD) systems [10]. Recently, the equivalence of PWA, LC, ELC, MMPS, and MLD hybrid dynamical systems was proven constructively in [6, 8, 24]. Thus, the theoretical properties and tools can be easily transferred from one class to another. Each modeling framework has its advantages. For instance, stability criteria were formulated for PWA systems [26] and control and verification techniques were proposed for MLD discrete-time hybrid models [12]. In particular, MLD models have proven successful for recasting hybrid dynamical optimization problems into mixed-integer linear and quadratic programs, solvable via branch and bound techniques [32].

MLD systems [10] allow specifying the evolution of continuous variables through linear dynamic equations, of discrete variables through propositional logic statements and automata, and the mutual interaction between the two. Linear dynamics are represented as difference equations $x(t+1) = Ax(t) + Bu(t)$, $x \in \mathbb{R}^n$. Boolean variables are defined from linear-threshold conditions over the continuous variables. The key idea of the approach consists of embedding the logic part in the state equations by transforming Boolean variables into 0-1 integers, and by expressing the relations as mixed-integer linear inequalities [10, 16, 33, 42].

By collecting the equalities and inequalities derived from the representation of the hybrid system we obtain the Mixed Logical Dynamical (MLD) system [10]

$$x(t+1) = Ax(t) + B_1u(t) + B_2\delta(t) + B_3z(t) \quad (6a)$$

$$E_2\delta(t) + E_3z(t) \leq E_1u(t) + E_4x(t) + E_5, \quad (6b)$$

where $x \in \mathbb{R}^{n_c} \times \{0, 1\}^{n_\ell}$ is a vector of continuous and binary states, $u \in \mathbb{R}^{m_c} \times \{0, 1\}^{m_\ell}$ are the inputs, $\delta \in \{0, 1\}^{r_\ell}$, $z \in \mathbb{R}^{r_c}$ represent auxiliary binary and continuous variables respectively, which are introduced when transforming logic relations into mixed-integer linear inequalities, and A , B_{1-3} , E_{1-5} are matrices of suitable dimensions. For instance, the linear-threshold condition

$$[\delta = 1] \leftrightarrow [a'x \leq b] \quad (7)$$

is transformed into

$$a'x - b \leq M(1 - \delta) \quad (8a)$$

$$a'x - b \geq \epsilon + (m - \epsilon)\delta \quad (8b)$$

where M , m are upper and lower bounds, respectively, on the scalar quantity $a'x - b$, and ϵ is a small positive scalar, e.g., the machine precision. Another example is the if-then-else construct

$$\text{IF } [\delta = 1] \text{ THEN } z = a'_1x - b_1 \text{ ELSE } z = a'_2x - b_2 \quad (9)$$

which is transformed into

$$(m_2 - M_1)\delta + z \leq a'_2x - b_2 \quad (10a)$$

$$(m_1 - M_2)\delta - z \leq -a'_2x + b_2 \quad (10b)$$

$$(m_1 - M_2)(1 - \delta) + z \leq a'_1x - b_1 \quad (10c)$$

$$(m_2 - M_1)(1 - \delta) - z \leq -a'_1x + b_1 \quad (10d)$$

where M_i, m_i are upper and lower bounds on $a'_i x - b_i$, $i = 1, 2$. We assume that system (6) is *completely well-posed* [10], which means that for all x, u within a bounded set the variables δ, z are uniquely determined, i.e., there exist functions F, G such that, at each time t , $\delta(t) = F(x(t), u(t))$, $z(t) = G(x(t), u(t))$. This allows assuming that $x(t+1)$ and $y(t)$ are uniquely defined once $x(t)$, $u(t)$ are given, and therefore that x -trajectories exist and are uniquely determined by the initial state $x(0)$ and input signal $u(t)$. It is clear that the well-posedness assumption stated above is usually guaranteed by the procedure used to generate the linear inequalities (6b), and therefore this hypothesis is typically fulfilled by MLD relations derived from modeling real-world plants.

Examples of real-world applications that can be naturally modeled within the MLD framework are reported in [10]. The language HYSDEL (HYbrid Systems DEscription Language) was developed in [40] to obtain MLD models from of a high level textual description of the hybrid dynamics.

IV. DISCRETE-TIME HYBRID MODEL OF THE VEHICLE

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

1. The frictional torque τ_t is approximated as a piecewise affine function of the slip $\Delta\omega$ and of the road coefficient of friction μ by using the approach described in [20]. The algorithm proposed in [20] generates a polyhedral partition of the $(\Delta\omega, \mu)$ -space and the corresponding affine approximation of the torque τ_t in each region.

If the number of regions is limited to two we get:

$$\tau_t(\Delta\omega, \mu) = \begin{cases} k_{11}\Delta\omega + k_{12}\mu + k_{13} & \text{if } 0.21\Delta\omega - 5.37\mu \leq -0.61 \\ k_{21}\Delta\omega + k_{22}\mu + k_{23} & \text{if } 0.21\Delta\omega - 5.37\mu > -0.61 \end{cases} \quad (11)$$

where $k_{11} = 67.53$, $k_{12} = 102.26$, $k_{13} = -31.59$, $k_{21} = -1.85$, $k_{22} = 1858.3$, $k_{23} = -232.51$, as depicted in Figure 2(b).

2. Model (1) is discretized with sampling time $T_s = 20$ ms and the PWA model (11) of the frictional torque is used to obtain the following discrete time PWA model of the vehicle:

$$\tilde{x}(t+1) = \begin{cases} \begin{bmatrix} 0.98316 & 0.78486 \\ 0.00023134 & 0.989220 \end{bmatrix} \tilde{x}(t) + \begin{bmatrix} 0.048368 \\ 5.6688e-6 \end{bmatrix} \tau_c(t) + \begin{bmatrix} -0.35415 \\ 0.0048655 \end{bmatrix} \mu(t) + \begin{bmatrix} 0.10943 \\ -0.0015034 \end{bmatrix} \\ \text{if } 0.21\Delta\omega - 5.37\mu \leq -0.61 \\ \begin{bmatrix} 1.0005 & -0.021835 \\ -6.4359e-6 & 1.00030 \end{bmatrix} \tilde{x}(t) + \begin{bmatrix} 0.048792 \\ -1.5695e-7 \end{bmatrix} \tau_c(t) + \begin{bmatrix} -6.5287 \\ 0.089695 \end{bmatrix} \mu(t) + \begin{bmatrix} 0.81687 \\ -0.011223 \end{bmatrix} \\ \text{if } 0.21\Delta\omega - 5.37\mu > -0.61 \end{cases} \quad (12)$$

where $\tilde{x} = [\frac{\omega_e}{v_v}]$. At this stage τ_c is considered as the control input to the system. The time delay between τ_c and τ_d will be taken into account in the controller design phase detailed in subsection VI-B.

3. The following constraints on the torque, on its variation, and on the slip need to be satisfied:

$$\tau_c \leq 176 \text{ Nm} \quad (13a)$$

$$\tau_c \geq -20 \text{ Nm} \quad (13b)$$

$$\dot{\tau}_c \approx \frac{\tau_c(k) - \tau_c(k-1)}{T_s} \leq 2000 \text{ Nm/s} \quad (13c)$$

$$\Delta\omega \geq 0. \quad (13d)$$

In order to constrain the derivative of the input, the state vector is augmented by including the previous torque $\tau_c(t-1)$. The variation of the combustion torque $\Delta\tau_c(t) = \tau_c(t) - \tau_c(t-1)$ will be the new input variable.

The resulting hybrid discrete-time model has three states ($x_1 = \text{previous } \tau_c$, $x_2 = \omega_e$, $x_3 = v_v$), one control input ($u_1 = \Delta\tau_c$), one uncontrollable input ($u_2 = \mu$), one regulated output ($y = \Delta\omega$), one auxiliary binary variable $\delta \in \{0, 1\}$ indicating the affine region where the system is operating, $[\delta = 0] \iff [0.21\Delta\omega - 5.37\mu \leq -0.61]$, and two auxiliary continuous variables $z \in \mathbb{R}^2$ describing the dynamics in (12), i.e.,

$$z = \begin{cases} A_1\tilde{x} + B_{11}\tau_c + B_{12}\mu + f_1 & \text{if } \delta = 0 \\ A_2\tilde{x} + B_{21}\tau_c + B_{22}\mu + f_2 & \text{otherwise,} \end{cases}$$

where $A_1, A_2, B_{11}, B_{12}, B_{21}, B_{22}, f_1, f_2$ are the matrices in (12). The resulting MLD model is obtained by processing the description list reported in Appendix A through the HYSDEL compiler:

$$x(t+1) = \begin{bmatrix} 0 & 0 & 0 \\ 0.0484 & 0 & 0 \\ 0.0897 & 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 1.0000 & 0 \\ 0.0484 & 0 \\ 0.0897 & 0 \end{bmatrix} \begin{bmatrix} \Delta\tau_c(t) \\ \mu(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \delta(t) + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} z(t) \quad (14a)$$

$$10^3 \begin{bmatrix} 1.1000 \\ 1.1000 \\ -1.1000 \\ -1.1000 \\ 0.1800 \\ 0.1800 \\ -0.1800 \\ -0.1800 \\ -0.4000 \\ 0.4000 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \delta(t) + \begin{bmatrix} -1 & 0 \\ 1 & 0 \\ -1 & 0 \\ 1 & 0 \\ 0 & -1 \\ 0 & 1 \\ 0 & -1 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} z(t) \leq \begin{bmatrix} 0 & 0.3542 \\ 0 & -0.3542 \\ 0 & 6.5287 \\ 0 & -6.5287 \\ 0 & -0.0049 \\ 0 & 0.0049 \\ 0 & -0.0897 \\ 0 & 0.0897 \\ 0 & 0.2125 \\ 0 & -0.2125 \\ -1.0000 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta\tau_c(t) \\ \mu(t) \end{bmatrix} + \begin{bmatrix} 0 & -0.9832 & -0.7849 \\ 0 & 0.9832 & 0.7849 \\ 0 & -1.0005 & 0.0218 \\ 0 & 1.0005 & -0.0218 \\ 0 & -0.0002 & -0.9892 \\ 0 & 0.0002 & 0.9892 \\ 0 & 0.0000 & -1.0003 \\ 0 & -0.0000 & 1.0003 \\ 0 & -0.3872 & 18.0473 \\ 0 & 0.3872 & -18.0473 \\ 0 & 0 & 0 \\ -1.0000 & 0 & 0 \\ 1.0000 & 0 & 0 \\ 0 & 0.0720 & -3.3557 \end{bmatrix} x(t) + 10^3 \begin{bmatrix} 1.1000 \\ 1.1000 \\ -0.0008 \\ 0.0008 \\ 0.1800 \\ 0.1800 \\ 0.0000 \\ -0.0000 \\ 0.0006 \\ 0.3994 \\ 2.0000 \\ 0.1760 \\ 0.0200 \end{bmatrix} \quad (14b)$$

Note that the constraints in (13) are included in the system matrices (14b).

V. CONSTRAINED OPTIMAL CONTROL

Figure 3 depicts the lateral and longitudinal traction torque as a function of the wheel slip. It is clear that if the wheel slip increases beyond a certain value, the longitudinal and lateral driving forces on the tire decrease considerably and the vehicle cannot speed up and steer as desired.

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. The overall control scheme is depicted in Figure 4 and is composed of two parts: a device that estimates the road surface condition μ and consequently generates a desired wheel slip $\Delta\omega_d$, and a traction controller that regulates the wheel slip at the desired value $\Delta\omega_d$. In this paper we only focus on the second part, as the first one is already available from previous projects at Ford Research Laboratories.

The control system receives the desired wheel slip $\Delta\omega_d$, the estimated road coefficient adhesion μ , the measured front and rear wheel speeds as input and generates the desired engine torque τ_c (the time delay between τ_c and τ_d will be compensated a posteriori as described in Section VI-B).

In the sequel we describe how a Model Predictive Controller (MPC) can be designed for the posed traction control problem. The main idea of MPC is to use the *model* of the plant to *predict* the future evolution of the system [31]. 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:

$$\min_V \sum_{k=0}^{T-1} |Q(\Delta\omega_k - \Delta\omega_d(t))| + |R\Delta\tau_{c,k}| \quad (15)$$

$$\text{subj. to } \begin{cases} x_{k+1} = Ax_k + B_1 \left[\frac{\Delta\tau_{c,k}}{\mu(t)} \right] + B_2\delta_k + B_3z_k \\ E_2\delta_k + E_3z_k \leq E_1 \left[\frac{\Delta\tau_{c,k}}{\mu(t)} \right] + E_4x_k + E_5 \\ x_0 = x(t) \\ \Delta\tau_{c,k} = \tau_{c,k} - \tau_{c,k-1}, \quad k = 0, \dots, T-1, \quad \tau_{c,-1} \triangleq \tau_c(t-1) \end{cases} \quad (16)$$

where matrices $A, B_1, B_2, B_3, E_2, E_3, E_1, E_4, E_5$ are given in (14), and $V \triangleq [\Delta\tau_{c,0}, \dots, \Delta\tau_{c,T-1}]'$ is the optimization vector. Note that the optimization variables are the torque variations $\Delta\tau_{ck} = \tau_{c,k} - \tau_{c,k-1}$, and that the set point $\Delta\omega_d(t)$ and the current road coefficient of adhesion $\mu(t)$ are considered constant over the prediction horizon T .

Problem (15)-(16) can be translated into a mixed integer linear program (MILP) (the minimization of a linear cost function subject to linear constraints with binary and continuous variables) of the form:

$$\min_{\mathcal{E}, V, Z, \delta} \sum_{k=0}^{T-1} \varepsilon_k^w + \varepsilon_k^u \quad (17a)$$

$$\text{subj. to} \quad G^\varepsilon \mathcal{E} + G^u V + G^Z z + G^\delta \delta \leq S + F \begin{bmatrix} \omega_e(t) \\ v_v(t) \\ \tau_c(t-1) \\ \mu(t) \\ \Delta\omega_d(t) \end{bmatrix}, \quad (17b)$$

where $Z = [z'_0, \dots, z'_{T-1}]' \in \mathbb{R}^{2T}$, $\delta = [\delta_0, \dots, \delta_{T-1}]' \in \{0, 1\}^T$ and $\mathcal{E} = [\varepsilon_0^w, \dots, \varepsilon_{T-1}^w, \varepsilon_0^u, \dots, \varepsilon_{T-1}^u]' \in \mathbb{R}^{2T}$ is a vector of additional slack variables satisfying $\varepsilon_k^w \geq \pm Q(\Delta\omega_k - \Delta\omega_d(t))$, $\varepsilon_k^u \geq \pm R\Delta\tau_{c,k}$, $k = 0, 1, \dots, T-1$ introduced in order to translate the cost function (15) into the linear cost function (17a). Matrices $G^\varepsilon, G^u, G^Z, G^\delta, S, F$ are matrices of suitable dimension that, as described in [7], can be constructed from Q, R, T and $A, B_1, B_2, B_3, E_2, E_3, E_1, E_4, E_5$. The resulting control law is

$$\tau_c(t) = \tau_c(t-1) + \Delta\tau_{c,0}^*, \quad (18)$$

where $V^* \triangleq [\Delta\tau_{c,0}^*, \dots, \Delta\tau_{c,T-1}^*]'$ denotes the sequence of optimal input increments computed at time t by solving (17) for the current measurements $\omega_e(t)$, $v_v(t)$, set point $\Delta\omega_d(t)$, and estimate of the road coefficient $\mu(t)$.

VI. CONTROLLER DESIGN

The design of the controller is performed in two steps. First, the MPC controller (15)-(18) based on model (14) is tuned in simulation until the desired performance is achieved. The MPC controller is not directly implementable, as it would require the MILP (17) to be solved on-line, which is clearly prohibitive on standard automotive control hardware. Therefore, for implementation, in the second phase the explicit piecewise affine form of the MPC law is computed off-line by using the multiparametric mixed integer programming (mp-MILP) solver presented in [19]. According to the approach of [7], the resulting control law has the piecewise affine form

$$\tau_c(t) = F_i\theta(t) + g_i \quad \text{if} \quad H_i\theta(t) \leq k_i, \quad i = 1, \dots, n_r, \quad (19)$$

where $\theta(t) = [\omega_e(t) \ v_v(t) \ \tau_c(t-1) \ \mu(t) \ \Delta\omega_d(t)]'$. Therefore, the set of states + references is partitioned into n_r polyhedral cells, and an affine control law is defined in each one of them. Rather than solving the MILP (17) *on-line* for the given $\theta(t)$, the idea is to use the mp-MILP solver to compute *off-line* the solution of the MILP (17) for all the parameters $\theta(t)$ within a given polyhedral set. Although the resulting piecewise affine 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 affine function. The control law can be implemented on-line in the following simple way: (i) determine the i -th region that contains the current vector $\theta(t)$ (current measurements and references); (ii) compute $u(t) = F_i\theta(t) + g_i$ according to the corresponding i -th control law. A more efficient way of evaluating the piecewise affine control law based on the organization of the controller gains on a balanced search tree is reported in [39].

A. Tuning

A good hybrid MPC design strategy consists of tuning the MPC controller using simulation and on-line optimization, and then converting the controller to its piecewise affine explicit form (19). The piecewise affine controller will behave in exactly the same way, but with a lower computation cost. The parameters of the controller (15)-(18) to be tuned are the horizon length T and the weights Q and R . By increasing the prediction horizon T the controller performance improves, but at the same time the number of constraints in (16) increases. As in general the complexity of the final piecewise affine controller increases dramatically with the number of constraints in (16) (see [11] for the case of linear systems), tuning T amounts to finding the smallest T which leads to a satisfactory closed-loop behaviour. Simulations were carried out to test the controller against changes to model parameters.

A satisfactory performance was achieved with $T = 4, Q = 50, R = 1$ which corresponds to an explicit controller consisting of $n_r = 137$ regions (Hybrid Controller 1). By using $T = 5, Q = 50, R = 1$ one obtains a slightly better performance at the price of a higher number of regions in the explicit controller, that is $n_r = 504$. We will refer to this controller as Hybrid Controller 2.

B. Combustion Torque Delay

The vehicle model in Section II is affected by a time delay of $\sigma = \frac{T_f}{T_s} = 12$ sampling intervals between the desired commanded torque τ_d and the combustion torque τ_c . To avoid the introduction of σ auxiliary states in the hybrid model (6), we take such a delay into account only during implementation of the control law.

Let the current time be $t \geq \sigma$ and let the explicit optimal control law in (19) be denoted as $\tau_c(t) = f_{PWA}(\theta(t))$. Then, we compensate for the delay by setting

$$\tau_d(t) = \tau_c(t + \sigma) = f_{PWA}(\hat{\theta}(t + \sigma)), \quad (20)$$

where $\hat{\theta}(t + \sigma)$ is the σ -step ahead predictor of $\theta(t)$. Since at time t , the inputs $\tau_d(t - i)$, $i = 1, \dots, \sigma$ and therefore $\tau_c(t - i + \sigma)$, $i = 1, \dots, \sigma$ are available, $\hat{\theta}(t + \sigma)$ can be computed from $\omega_e(t)$, $v_v(t)$ by iterating the PWA model (12), by assuming $\mu(t + \sigma) = \mu(t)$ and $\Delta\omega_d(t + \sigma) = \Delta\omega_d(t)$.

In order to motivate the assumption in (20), in Appendix B we show that for the related setting of linear quadratic regulation (LQR) of linear time invariant models with delays (quadratic performance indices, infinite prediction horizon), such an assumption is exact, that is, the LQR gain for the delayed system is obtained by combining the LQR gain for the delay-free system with a σ -steps ahead predictor. Through tedious algebraic manipulations, it is possible to prove that this is true also in the present hybrid finite-horizon context.

VII. MOTIVATION FOR HYBRID CONTROL

There are several reasons that led us to solve the traction control problem by using a hybrid approach. First of all the nonlinear frictional torque τ_t in (4) has a clear piecewise-linear behaviour [37]: The traction torque increases almost linearly for low values of the slip, until it reaches a certain peak after which it start decreasing. For various road conditions the curves have different peaks and slopes. By including such a piecewise linearity in the model we obtained a single control law that is able to achieve the control task for a wide range of road conditions. Moreover, the design flow has the following advantages: (i) From the definition of the control objective to its solution, the problem is tackled in a systematic way by using the HYSDEL compiler and multiparametric programming algorithms; (ii) Constraints are embedded in the control problem in a natural and effective way; (iii) The resulting control law is piecewise affine and requires much less supervision by logical constructs than controllers developed with traditional techniques (e.g. PID control); (iv) It is easy to extend the design to handle more accurate models and include additional constraints without changing the design flow. For example, one can use a better piecewise-linear approximation of the traction torque, a more detailed model of the dynamics and include logic constructs in the model such as an hysteresis for the controller activation as a function of the slip.

In terms of performance, the results obtained with our approach are comparable with a well-tuned PID controller used at Ford Motor Company. The experiments will show that a good performance is achieved despite the limited development time compared to the time needed for the design of the PID controller. Moreover, the hybrid approach proposed in this paper provides an insight on the achievable limits of control performance, as discussed next.

VIII. SIMULATION RESULTS

Extensive simulations were carried out before testing the hybrid controller on a passenger vehicle. In particular, different controllers based on linear/affine models of the vehicle were considered and their performance was compared to the hybrid controller performance. In this section we present a summary of the simulation results obtained by using standard linear Model Predictive Control (MPC) synthesis techniques for different linear models of the vehicle.

By combining equation (1) and equation (11) we obtain two linear models for the vehicle dynamics of dimension two. We will refer to this models as affine model 1 and affine model 2.

Affine model 1:

$$\begin{pmatrix} \dot{\omega}_e \\ \dot{v}_v \end{pmatrix} = \begin{pmatrix} -\frac{b_e}{J'_e} - \frac{k_{11}}{J'_e g_r^2} & \frac{k_{11}}{J'_e} g_r r_t \\ \frac{k_{11}}{m_v r_t g_r} & -\frac{k_{11}}{m_v r_t^2} \end{pmatrix} \begin{pmatrix} \omega_e \\ v_v \end{pmatrix} + \begin{pmatrix} \frac{1}{J'_e} \\ 0 \end{pmatrix} \tau_c + \begin{pmatrix} -\frac{k_{12}}{J'_e g_r} \\ \frac{k_{12}}{m_v r_t} \end{pmatrix} \mu + \begin{pmatrix} -\frac{k_{13}}{J'_e g_r} \\ \frac{k_{13}}{m_v r_t} \end{pmatrix} \quad (21)$$

Affine model 2:

$$\begin{pmatrix} \dot{\omega}_e \\ \dot{v}_v \end{pmatrix} = \begin{pmatrix} -\frac{b_e}{J'_e} - \frac{k_{21}}{J'_e g_r^2} & \frac{k_{21}}{J'_e} g_r r_t \\ \frac{k_{21}}{m_v r_t g_r} & -\frac{k_{21}}{m_v r_t^2} \end{pmatrix} \begin{pmatrix} \omega_e \\ v_v \end{pmatrix} + \begin{pmatrix} \frac{1}{J'_e} \\ 0 \end{pmatrix} \tau_c + \begin{pmatrix} -\frac{k_{22}}{J'_e g_r} \\ \frac{k_{22}}{m_v r_t} \end{pmatrix} \mu + \begin{pmatrix} -\frac{k_{23}}{J'_e g_r} \\ \frac{k_{23}}{m_v r_t} \end{pmatrix} \quad (22)$$

By combining equation (1) and the derivative of τ_t obtained by differentiating equation (11), we obtain two linear models for the vehicle dynamics of dimension three. We will assume that μ is constant over the prediction horizon and will denote this two models as linear model 3 and linear model 4.

Linear model 3:

$$\begin{pmatrix} \dot{\omega}_e \\ \dot{v}_v \\ \dot{\tau}_t \end{pmatrix} = \begin{pmatrix} -\frac{b_e}{J'_e} & 0 & -\frac{1}{J'_e g_r} \\ 0 & 0 & \frac{1}{m_v r_t} \\ -\frac{k_{11} J'_e}{g_r b_e} & 0 & -k_{11} J'_e - k_{11} m_v \end{pmatrix} \begin{pmatrix} \omega_e \\ v_v \\ \tau_t \end{pmatrix} + \begin{pmatrix} \frac{1}{J'_e} \\ 0 \\ 0 \end{pmatrix} \tau_c \quad (23)$$

Linear model 4:

$$\begin{pmatrix} \dot{\omega}_e \\ \dot{v}_v \\ \dot{\tau}_t \end{pmatrix} = \begin{pmatrix} -\frac{b_e}{J'_e} & 0 & -\frac{1}{J'_e g_r} \\ 0 & 0 & \frac{1}{m_v r_t} \\ -\frac{k_{21} J'_e}{g_r b_e} & 0 & -k_{21} J'_e - k_{21} m_v \end{pmatrix} \begin{pmatrix} \omega_e \\ v_v \\ \tau_t \end{pmatrix} + \begin{pmatrix} \frac{1}{J'_e} \\ 0 \\ 0 \end{pmatrix} \tau_c \quad (24)$$

These four models were used to design four linear Model Predictive Controllers subject to constraints (13), where the delay in (2) was compensated as described in Section VI-B. The four linear traction controllers were simulated by using a nonlinear model of the vehicle driving on a polished ice surface ($\mu = 0.2$) with $\omega_e(0) = 180.6$ rad/s and $v_v(0) = 0$ m/s (which represent the vehicle standing initially still with the wheels slipping). We compared the performance of the linear controllers to the one obtained by using a hybrid controller.

Linear MPC based on model 1. The performance is in general very bad independently of the tuning of the MPC. Figure 5 depicts a simulation of one of the best tuned MPC based entirely on linear model 1. The explanation for such poor behavior can be mainly found in the large model mismatch, due to the large difference in tire slope characteristics between the two model regions. This poor performance may be improved by adding a Kalman filter or, perhaps better, by adding additional states (for instance, by extending the linear two-dimensional model with the integral of the output in order to obtain an integral action, as was done in the hybrid context in [9]). The benefits of using Kalman filtering and of augmenting the linear model with additional states are clear when linear model 3 is used.

Linear MPC based on model 2. The performance improves compared to the affine model 1. However, a small model mismatch generates a steady state offset as can be seen in Figure 6. Such a steady state error can be removed with the introduction of additional states and Kalman filtering, as described earlier. The advantage of using Kalman filtering and an extended model is apparent from the performance achieved by using linear model 4.

Linear MPC based on models 3 and 4. These two cases lead to similar performance, which is, in general, good, as can be seen from Figure 7 for the case of model 4. In fact, the vehicle model is very sensitive to the traction torque model τ_t . In models 3 and 4 the traction torque τ_t is a state that can be estimated from the measurements by using a Kalman filter. Despite a model mismatch, the estimation of τ_t is relatively good and this justifies the good performance of such controllers. However, the model mismatch in this case leads to about 21% larger initial spin, which can be seen by

comparing Figure 7 with the corresponding simulation results for the hybrid case shown in Figure 8. Note in particular, that the model 4 leads to an additional engine torque pulse in the initial phase of slip control, which in turn results in an additional “glitch” in the initial slip curve and overall more excessive initial spin.

The experimental results obtained with the linear MPC controller based on model 4 will be presented in the next section. We want to point out that the optimal hybrid controller presented in this paper quantifies the best performance achievable in the control problem at hand, therefore providing a measurement unit for the degree of performance achieved by the linear MPC controllers, which is unknown a priori.

IX. EXPERIMENTAL SETUP AND RESULTS

The hybrid traction controller was tested in a small (1390 kg) front-wheel-drive passenger vehicle with manual transmission. The explicit controller was run with a 20 ms timebase in a 266 MHz Pentium II-based laptop. Vehicle wheel speeds were measured directly by the laptop computer, and the calculated engine torque command was passed to the powertrain control module through a serial bus. Torque control in the engine was accomplished through spark retard, cylinder air/fuel ratio adjustment, and cylinder fuel shutoff where needed. The overall system latency from issuance of the torque command to production of the actual torque by the engine by the engine was relatively large (0.25 sec), which is in part attributed to computational and implementation delays. The vehicle was tested on a polished ice surface (indoor ice arena, $\mu \simeq 0.2$) with a variety of ramp, step, and sinusoidal tracking reference signals. Control intervention was initiated when the average driven wheel speed exceeded the reference wheel speed for the first time.

As indicated above, the experiments were conducted on a uniform ice surface in an ice arena that provided suitable test and development facilities during the warmer periods of the year. Due to the obvious space limitations, only limited speed tests were possible, which can still display key characteristics of a given traction control system. In particular, the tests were done for aggressive, wide-open throttle (“full gas”) tip-ins from a standstill condition in first gear, where brakes were typically applied prior to the tip-in. This large initial disturbance and subsequent “pedal-to-the-metal” operation creates some of the most demanding conditions for the traction controller. Note that the target slip is initially step-changed to about 10 rad/s and then gradually lowered as the engine speed is kept constant during the vehicle launch acceleration. Once the vehicle speed reaches the synchronous level with the corresponding engine speed (around time 10 sec), the clutch is fully engaged and the slip target is kept to a constant value of 2 rad/s.

Figure 9 shows test results for the case of a linear MPC based on model 4, and Figure 10 for the hybrid control case. Comparing the two, it can be seen that the hybrid control on the average results in circa 20% lower initial slip peak and significantly faster containment of the first spin. As explained in Section VIII, this is due to an additional engine torque hesitation pulse that can be seen in Figure 9. At the same time, the hybrid case shows somewhat noisier behavior, which could be further treated through appropriate filtering if needed. In addition, Figure 10b shows the target tracking capability around time 8 sec, where the effects of relatively large delay of 0.25 sec can also be seen. A comparison between the test results of Figures 9 and 10 and corresponding simulation results of Figures 7 and 8 reveals similar trends, including the above hesitation pulse effects and significantly smaller slip overshoots and better overall slip target tracking achieved with the hybrid controller (relatively large initial engine torque of circa 220 Nm is not present in the simulation results, which used an effective net torque between the engine and brakes). Good qualitative and quantitative correlation between the simulation and experimental results confirms that the above simple model was appropriate for the present study.

X. CONCLUDING REMARKS

In this paper we described a hybrid model and an optimization-based control strategy for a vehicle traction control problem. We showed, through experiments carried on at Ford Research Laboratories, that good and robust performance is achieved on polished ice, which represents some of the most challenging road surfaces since it requires the largest amount of torque reduction and precise control in the least favorable (small S/N) region of vehicle operations. The performance was relatively robust w.r.t. manual transmission clutch modes of application, which represents a challenging disturbance that is characteristic for manual power trains with their inherent event-to-event and driver-to-driver variability. It was also shown, through a comparison between simulation and actual vehicle test results, that the simple vehicle model used for this study was well suited for MPC and hybrid control designs and related performance predictions. Furthermore, the resulting optimal piecewise affine control law was easily implemented on low cost hardware.

The simulation and test results demonstrated that the l_1 -optimal hybrid controller used in the present problem can lead to about 20% reduction in peak slip amplitudes and corresponding spin duration when compared to best-case linear MPC counterparts. At the same time, the hybrid controller provided a systematic way to create a benchmark of optimal possible performance against which many other controllers - classical as well as modern - could be compared.

It should be pointed out that the present work was based on a very coarse approximation of tire characteristic curves. Further improvements are possible by more granular resolution of these characteristics. For these more complex piecewise affine 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. We are also currently working to extend the results of this paper to MPC formulation based on 2-norm.

ACKNOWLEDGMENTS

We thank Prof. M. Morari for his helpful comments on a preliminary version of the manuscript.

APPENDIX

I. HYSDEL HYBRID MODEL

Below we report the description list in HYSDEL of the traction control model described in Section IV.

```

SYSTEM FordCar {

INTERFACE {
/* Description of variables and constants */

STATE {
    REAL taotold ;
    REAL we      ;
    REAL vv      ;
}

INPUT { REAL deltataot; REAL mu;
}

PARAMETER {

    /* Region of the PWA linearization */
    /* ar * mu + br * deltaw <= cr */

```

```

REAL ar = -5.3781;
REAL br = 53.127/250;
REAL cr = -0.61532;

/* Other parameters */
REAL deltaxmax = 400;
REAL deltaxmin = -400;
REAL zwemax = 1000;
REAL zwemin = -100;
REAL zvvmax = 80;
REAL zvvmin = -100;
REAL gr = 13.89;
REAL rt = 0.298;
REAL e = 1e-6;

/* Dynamic behavior of the model (Matlab generated) */
REAL a11a = 0.98316      ;
REAL a12a = 0.78486      ;
REAL a21a = 0.00023134   ;
REAL a22a = 0.989220     ;
REAL b11a = 0.048368     ;
REAL b12a = -0.35415     ;
REAL b21a = 0.089695     ;
REAL b22a = 0.0048655    ;
REAL f1a = 0.048792      ;
REAL f2a = -1.5695e-007  ;

REAL a11b = 1.0005       ;
REAL a12b = -0.021835    ;
REAL a21b = -6.4359e-006 ;
REAL a22b = 1.00030      ;
REAL b11b = 0.048792     ;
REAL b12b = -6.5287      ;
REAL b21b = -1.5695e-007 ;
REAL b22b = 0.089695     ;
REAL f1b = 0.81687       ;
REAL f2b = -0.011223     ;
}

}

IMPLEMENTATION {
    AUX {
        REAL zwe, zvv;
        BOOL region;
    }

    AD {
        /* PWA Domain */
        region = ar * ((we / gr) - (vv / rt)) + br * mu - cr <= 0
        /* region= ((we / gr) - (vv / rt))-1.6 <= 0 */
        [deltamax,deltamin,e];
    }
}

```

```

}

DA {
    zwe = { IF region THEN a11a * we + a12a * vv + b12a* mu+f1a    [zwemax, zwemin, e]
            ELSE a11b * we + a12b * vv + b12b* mu+f1b    [zwemax, zwemin, e]
    };
    zvv = { IF region THEN a21a * we + a22a * vv + b22a* mu +f2a [zvvmax, zvvmin, e]
            ELSE a21b * we + a22b * vv + b22b* mu+f2b    [zvvmax, zvvmin, e]
    };
}

CONTINUOUS {
    taotold    = deltataot;
    we         = zwe+b11a* taotold+b11a* deltataot ;
    vv         = zvv+b21a* taotold+b21a* deltataot;
}

MUST { deltataot <= 2000;
        taotold <=176;
        -taotold <=20;
        (we/gr)-(vv/rt)>=0;
    }
}
}

```

II. OPTIMAL CONTROL OF LINEAR SYSTEMS WITH DELAY

Consider discrete time linear system

$$x(t+1) = Ax(t) + Bu(t - \Delta) \quad (25)$$

where $x(t) \in \mathbb{R}^n$ and $u(t) \in \mathbb{R}^m$ are the state and input vectors, respectively, and the corresponding augmented system

$$\tilde{x}(t+1) = \tilde{A}\tilde{x}(t) + \tilde{B}u(t) \quad (26)$$

where

$$\tilde{x}(t) = \begin{bmatrix} x(t) \\ u(t - \Delta) \\ u(t - \Delta + 1) \\ \vdots \\ u(t - 1) \end{bmatrix} \quad (27)$$

and

$$\tilde{A} = \begin{bmatrix} A & B & \mathbf{0}_{n \times m} & \cdots & \mathbf{0}_{n \times m} \\ \mathbf{0}_{m \times n} & \mathbf{0}_{m \times m} & \mathbf{I}_{m \times m} & \cdots & \mathbf{0}_{m \times m} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_{m \times n} & \mathbf{0}_{m \times m} & \mathbf{0}_{m \times m} & \cdots & \mathbf{I}_{m \times m} \\ \mathbf{0}_{m \times n} & \mathbf{0}_{m \times m} & \mathbf{0}_{m \times m} & \cdots & \mathbf{0}_{m \times m} \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} \mathbf{0}_{n \times m} \\ \mathbf{0}_{m \times m} \\ \vdots \\ \mathbf{0}_{m \times m} \\ \mathbf{I}_{m \times m} \end{bmatrix}, \quad (28)$$

where $\mathbf{I}_{i \times i} \in \mathbb{R}^{i \times i}$ is the identity matrix and $\mathbf{0}_{i \times j} \in \mathbb{R}^{i \times j}$ is a matrix with all the elements equal to zero.

Theorem 1: Let $\tilde{K}$ be the LQR gain of system (26) with $\tilde{Q} = \text{blkdiag}(Q, \mathbf{0}_{\Delta m \times \Delta m})$ and $\tilde{R} = R$ being the state and input weighting matrices, respectively. Then,

$$\tilde{K} = K \begin{bmatrix} A^{\Delta-1} & A^{\Delta-2}B & A^{\Delta-3}B & \cdots & B \end{bmatrix} \quad (29)$$

where K is the LQR gain for system (26) with $\Delta = 0$ and weighing matrices Q and R , i.e.,

$$K = -(B^T S B + R)^{-1} B^T S A \quad (30)$$

$$S = A(S - S B (B^T S B + R)^{-1} B^T S) A + Q \quad (31)$$

Proof:

The optimal control law for system (26) is

$$u(k) = \tilde{K} x(k), \quad k \geq \Delta \quad (32)$$

where

$$\tilde{K} = -(\tilde{B}^T \tilde{S} \tilde{B} + \tilde{R})^{-1} \tilde{B}^T \tilde{S} \tilde{A} \quad (33)$$

$$\tilde{S} = \tilde{A}(\tilde{S} - \tilde{S} \tilde{B} (\tilde{B}^T \tilde{S} \tilde{B} + \tilde{R})^{-1} \tilde{B}^T \tilde{S}) \tilde{A} + \tilde{Q} \quad (34)$$

Partition the matrix $\tilde{S}$ according to the structure of the matrix $\tilde{A}$

$$\tilde{S} = \begin{bmatrix} S_{11} & S_{12} & S_{13} & \cdots & S_{1\Delta} \\ S_{21} & S_{22} & S_{23} & \cdots & S_{2\Delta} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ S_{\Delta 1} & S_{\Delta 2} & S_{\Delta 3} & \cdots & S_{\Delta \Delta} \end{bmatrix} \quad (35)$$

where $S_{11} \in \mathbb{R}^{n \times n}$, $S_{1j} \in \mathbb{R}^{n \times m}$, $j = 2, \dots, \Delta$, $S_{i1} \in \mathbb{R}^{m \times n}$, $i = 2, \dots, \Delta$, $S_{ij} \in \mathbb{R}^{m \times m}$, $i, j = 2, \dots, \Delta$. By using the form (35) of $\tilde{S}$ and equations (28), the LQR gain $\tilde{K}$ in (33) can be written as:

$$\tilde{K} = (S_{\Delta\Delta} + R)^{-1} \begin{bmatrix} S_{\Delta 1} A & S_{\Delta 1} B & S_{\Delta 2} B & \cdots & S_{\Delta(\Delta-1)} B \end{bmatrix} \quad (36)$$

It is immediate to prove by substitution that the Riccati equation (34) of the augmented system is solved by the following equations

$$S_{1,1} = A^T A^{\Delta-1} S A^{\Delta-1} + \sum_{k=1}^{\Delta-1} A^T A^{\Delta-1-k} Q A^{\Delta-1-k} + Q \quad (37)$$

$$S_{i,1} = B^T A^T A^{\Delta-i} S A^{\Delta-1} + B^T \sum_{k=1}^{\Delta-i} A^T A^{\Delta-i-k} Q A^{\Delta-1-k}, \quad i = 2, \dots, \Delta \quad (38)$$

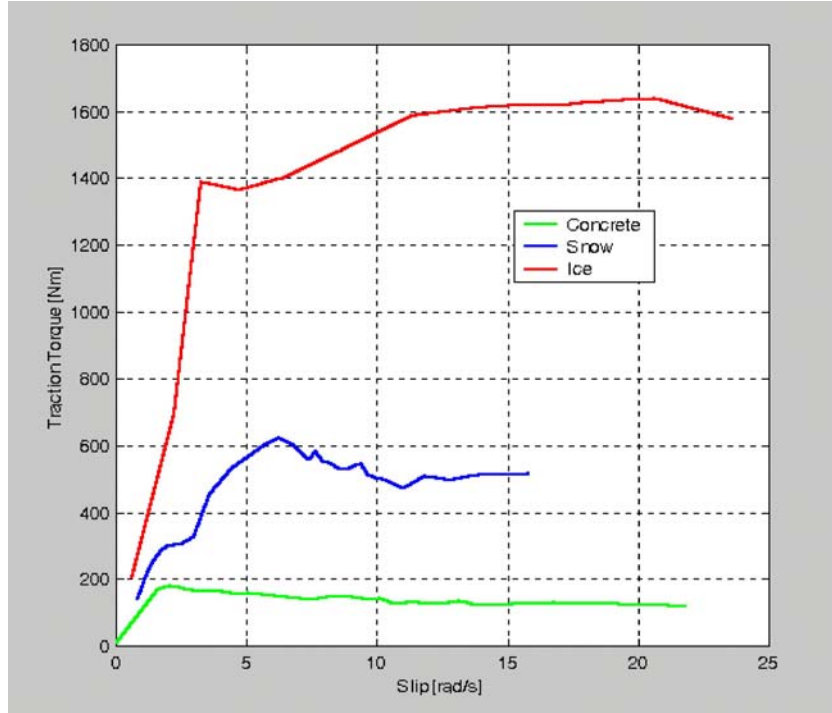
$$S_{i,j} = B^T A^T A^{\Delta-i} S A^{\Delta-1} B + B^T \left(\sum_{k=1}^{\Delta-i} A^T A^{\Delta-i-k} Q A^{\Delta-j-k} \right) B, \quad i = 2, \dots, \Delta, \quad j \leq i. \quad (39)$$

Equation (36) together with equations (37)-(39) prove the theorem. $\square$

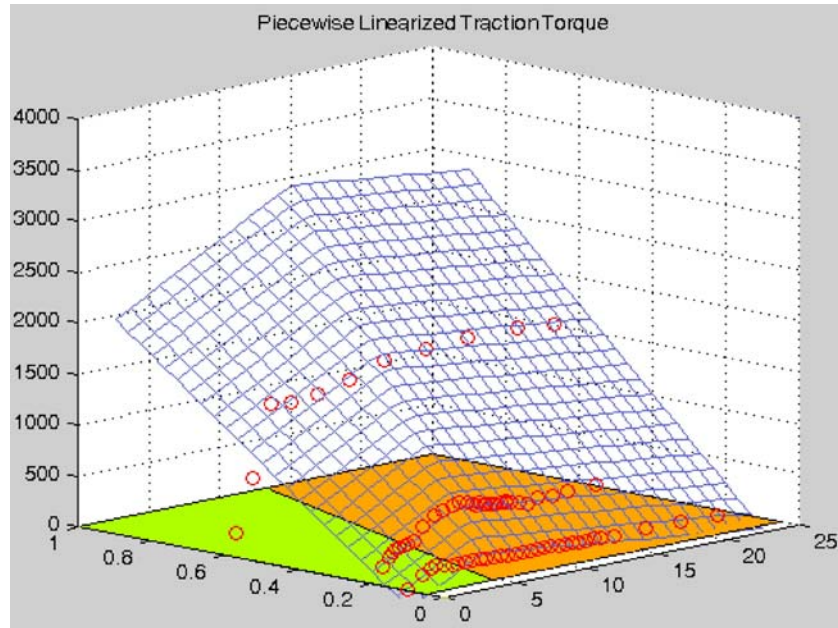
REFERENCES

- [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.
- [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. Torcini. 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. An efficient technique for translating mixed logical dynamical systems into piecewise affine systems. In *Proc. 41th IEEE Conf. on Decision and Control*, 2002.
- [7] A. Bemporad, F. Borrelli, and M. Morari. Piecewise linear optimal controllers for hybrid systems. In *Proc. American Contr. Conf.*, pages 1190–1194, Chicago, IL, June 2000.
- [8] A. Bemporad, G. Ferrari-Trecate, and M. Morari. Observability and controllability of piecewise affine and hybrid systems. *IEEE Trans. Automatic Control*, 45(10):1864–1876, 2000.
- [9] A. Bemporad, N. Giorgetti, I.V. Kolmanovsky, and D. Hrovat. A hybrid system approach to modeling and optimal control of DISC engines. In *Proc. 41th IEEE Conf. on Decision and Control*, 2002.
- [10] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, March 1999.
- [11] A. Bemporad, M. Morari, V. Dua, and E.N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.
- [12] A. Bemporad, F.D. Torrisi, and M. Morari. Optimization-based verification and stability characterization of piecewise affine and hybrid systems. In B. Krogh and N. Lynch, editors, *Hybrid Systems: Computation and Control*, volume 1790 of *Lecture Notes in Computer Science*, pages 45–58. Springer Verlag, 2000.
- [13] F. Borrelli, A. Bemporad, M. Fodor, and D. Hrovat. A hybrid approach to traction control. In A. Sangiovanni-Vincentelli and M.D. Di Benedetto, editors, *Hybrid Systems: Computation and Control*, Lecture Notes in Computer Science. Springer Verlag, 2001.
- [14] M.S. Branicky. *Studies in hybrid systems: modeling, analysis, and control*. PhD thesis, LIDS-TH 2304, Massachusetts Institute of Technology, Cambridge, MA, 1995.
- [15] M.S. Branicky, V.S. Borkar, and S.K. Mitter. A unified framework for hybrid control: model and optimal control theory. *IEEE Trans. Automatic Control*, 43(1):31–45, 1998.
- [16] T.M. Cavalier, P.M. Pardalos, and A.L. Soyster. Modeling and integer programming techniques applied to propositional calculus. *Computers Opns Res.*, 17(6):561–570, 1990.
- [17] B. De Schutter and B. De Moor. The extended linear complementarity problem and the modeling and analysis of hybrid systems. In P. Antsaklis, W. Kohn, M. Lemmon, A. Nerode, and S. Sastry, editors, *Hybrid Systems V*, volume 1567 of *Lecture Notes in Computer Science*, pages 70–85. Springer, 1999.
- [18] B. De Schutter and T. van den Boom. Model predictive control for max-plus-linear systems. In *Proc. American Contr. Conf.*, pages 4046–4050, 2000.
- [19] V. Dua and E.N. Pistikopoulos. An algorithm for the solution of multiparametric mixed integer linear programming problems. *Annals of Operations Research*, pages 123–139, 2000.
- [20] G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari. A clustering technique for the identification of piecewise affine systems. In M. Di Benedetto and A. Sangiovanni-Vincentelli, editors, *Proc. 4th International Workshop on Hybrid Systems: Computation and Control*, volume 2034 of *Lecture Notes in Computer Science*, pages 218–231. Springer-Verlag, 2001.
- [21] M. Fodor, J. Yester, and D. Hrovat. Active control of vehicle dynamics. In *Proceedings of the 17th Digital Avionics Systems Conference*, Seattle, Washington, November 1998.
- [22] W.P.M.H. Heemels. *Linear complementarity systems: a study in hybrid dynamics*. PhD thesis, Dept. of Electrical Engineering, Eindhoven University of Technology, The Netherlands, 1999.
- [23] W.P.M.H. Heemels, J.M. Schumacher, and S. Weiland. Linear complementarity systems. *SIAM Journal on Applied Mathematics*, 60(4):1234–1269, 2000.
- [24] W.P.M.H. Heemels, B. De Schutter, and A. Bemporad. Equivalence of hybrid dynamical models. *Automatica*, 2001. to appear.

- [25] D. Hrovat, J. Asgari, and M. Fodor. Automotive mechatronic systems. In Cornelius T. Leondes, editor, *Chapter 1 of "Mechatronic Systems Techniques and Applications Volume 2 - Transportation and Vehicular Systems"*. Gordon and Breach Science Publishers, 2000.
- [26] M. Johansson and A. Rantzer. Computation of piece-wise quadratic Lyapunov functions for hybrid systems. *IEEE Trans. Automatic Control*, 43(4):555–559, 1998.
- [27] T.A. Johansen, J. Kalkkuhl, J. Lüdemann, and I. Petersen. Hybrid control strategies in ABS. In *Proc. American Contr. Conf.*, Arlington, Virginia, June 2001.
- [28] 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.
- [29] G. Labinaz, M.M. Bayoumi, and K. Rudie. A survey of modeling and control of hybrid systems. *Annual Reviews of Control*, 21:79–92, 1997.
- [30] G. F. Mauer. A fuzzy logic controller for an ABS braking system. *IEEE Transaction on Fuzzy Systems*, November 1995.
- [31] D.Q. Mayne, J.B. Rawlings, C.V. Rao, and P.O.M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, June 2000.
- [32] G.L. Nemhauser and L.A. Wolsey. *Integer and Combinatorial Optimization*. Wiley, 1988.
- [33] R. Raman and I.E. Grossmann. Relation between MILP modeling and logical inference for chemical process synthesis. *Computers Chem. Engng.*, 15(2):73–84, 1991.
- [34] R.L. Grossmann, A. Nerode, A.P. Ravn, and H. Rischel (Eds.). *Hybrid Systems*. Springer Verlag, New York, 1993. no. 736 in LCNS.
- [35] E.D. Sontag. Nonlinear regulation: The piecewise linear approach. *IEEE Trans. Automatic Control*, 26(2):346–358, April 1981.
- [36] H. S. Tan. *Adaptive and Robust Controls with Application to Vehicle Traction Control*. PhD thesis, Univ. of California, Berkeley, 1988.
- [37] H. S. Tan and M. Tomizuka. Discrete time controller design for robust vehicle traction. *IEEE Control System Magazine*, 10(3):107–113, April 1990.
- [38] H.S. Tan and M. Tomizuka. An adaptive sliding mode vehicle traction controller design. In *Proc. American Contr. Conf.*, volume 2, pages 1856–1861, 1990.
- [39] P. Tøndel, T.A. Johansen, and A. Bemporad. Evaluation of piecewise affine control via binary search tree. *Automatica*, 2002. To appear.
- [40] F.D. Torrisi and A. Bemporad. HYSDEL — A tool for generating computational hybrid models. Technical Report AUT02-03, ETH Zurich, Submitted for publication on IEEE Transactions on Control Systems Technology, 2002. <http://control.ethz.ch/~hybrid/hysdel>.
- [41] A.J. van der Schaft and J.M. Schumacher. Complementarity modelling of hybrid systems. *IEEE Trans. Automatic Control*, 43:483–490, 1998.
- [42] H.P. Williams. *Model Building in Mathematical Programming*. John Wiley & Sons, Third Edition, 1993.



(a) Measured tire torque τ_t for three different road conditions



(b) Piecewise affine model of the tire torque τ_t

Fig. 2. Nonlinear behavior and its piecewise-linear approximation of the traction torque τ_t as a function of the slip $\Delta\omega$ and road coefficient adhesion μ

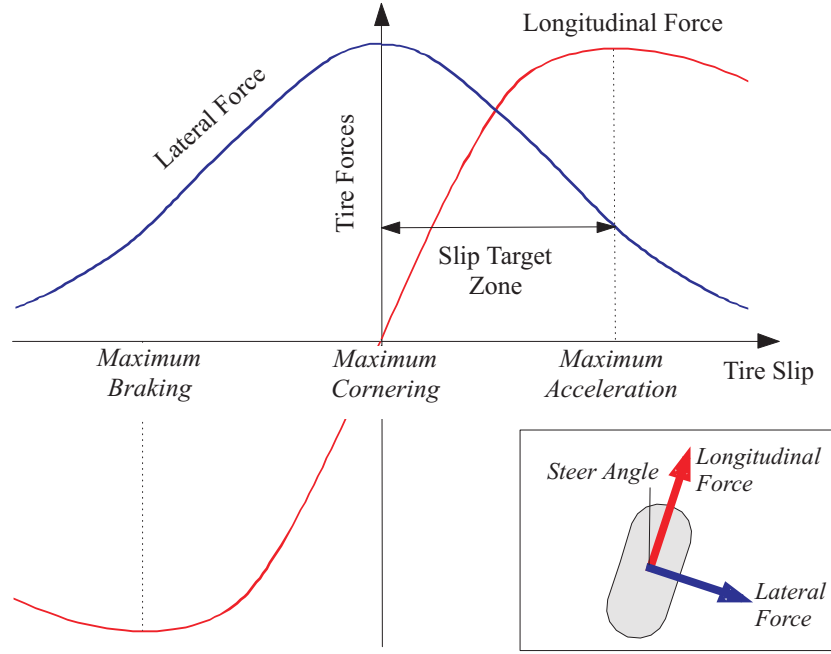


Fig. 3. Typical behaviour of lateral and longitudinal tire forces

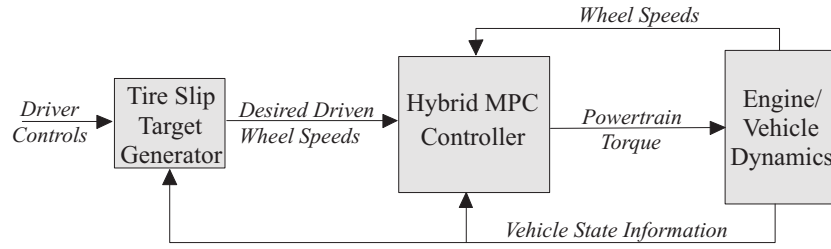


Fig. 4. Overall traction control scheme. In this paper we focus on the design of the hybrid MPC controller

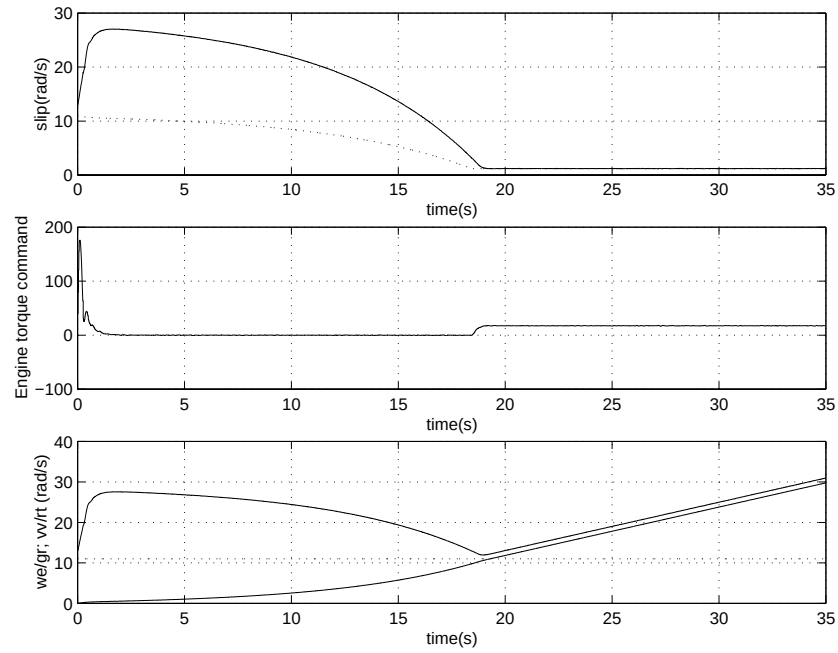


Fig. 5. Simulation results: Linear MPC based on affine model 1, $\Delta\omega_d = 2$ rad/s

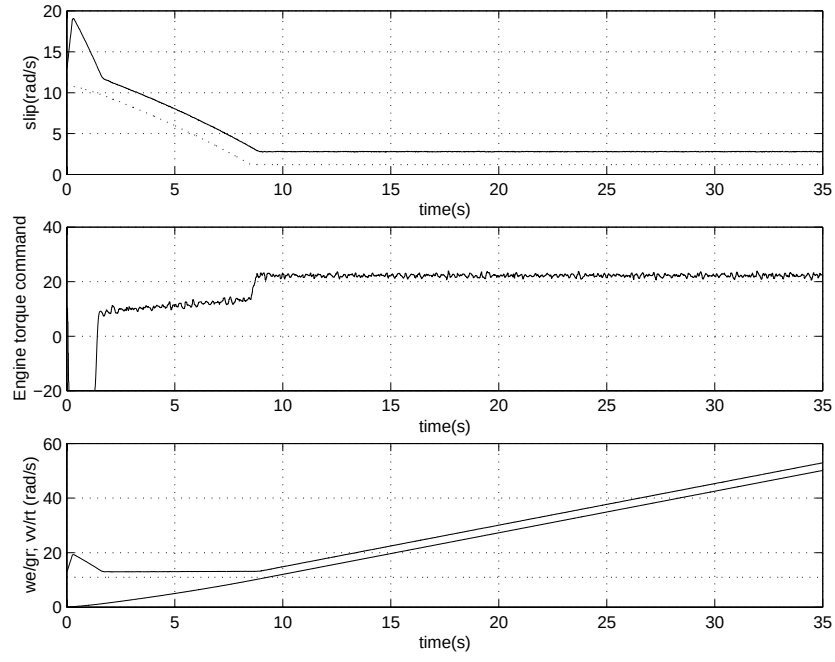


Fig. 6. Simulation results: Linear MPC based on affine model 2, $\Delta\omega_d = 2$ rad/s

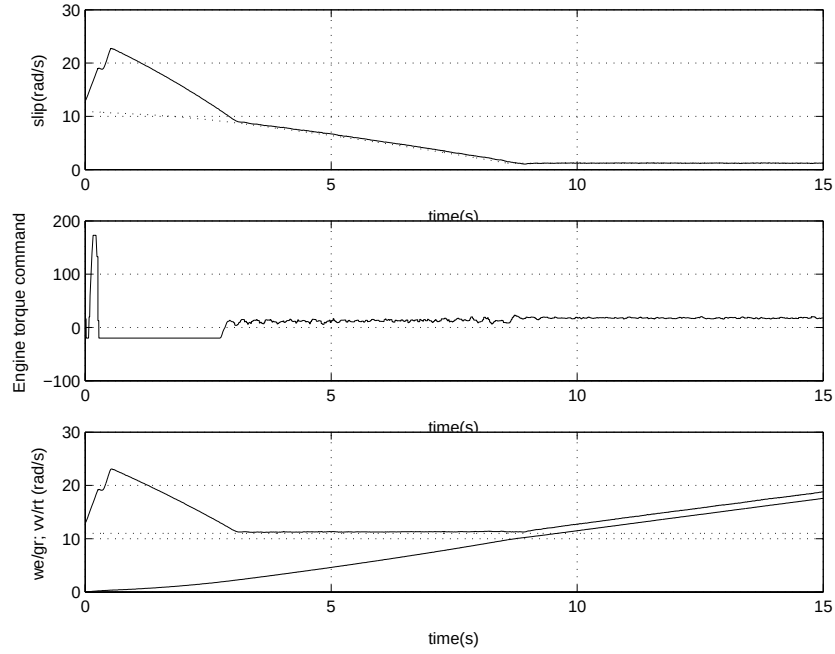


Fig. 7. Simulation results: Linear MPC based on linear model 4, $\Delta\omega_d = 2$ rad/s

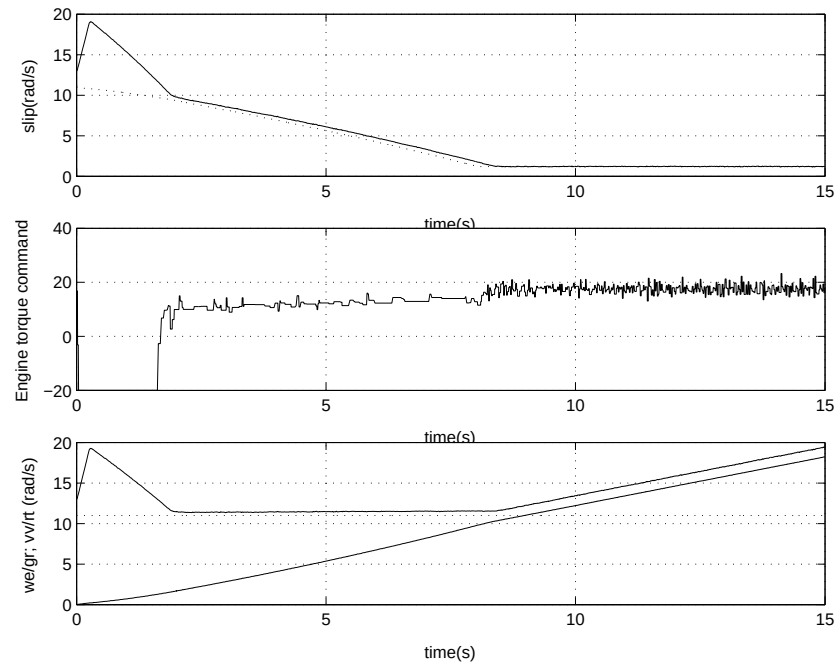
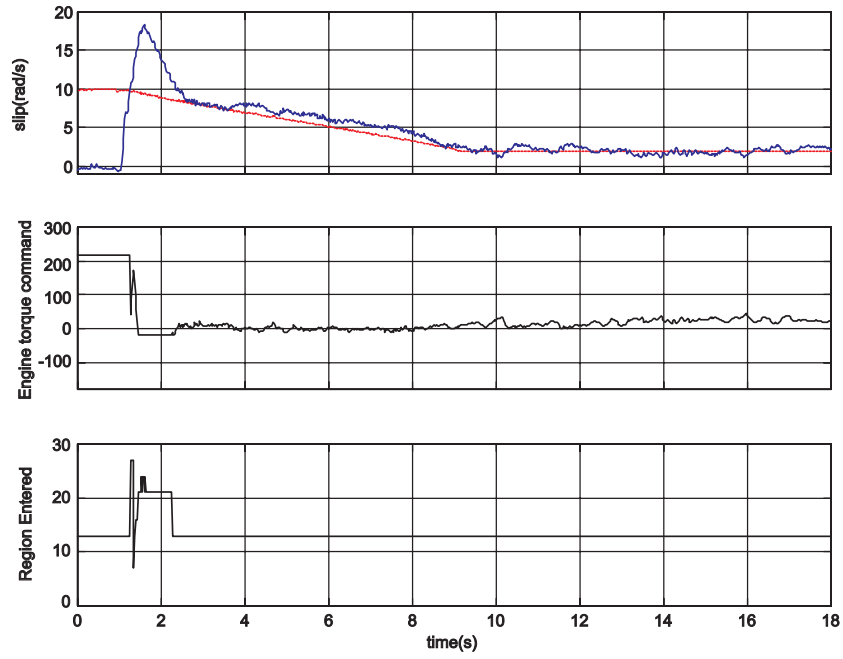
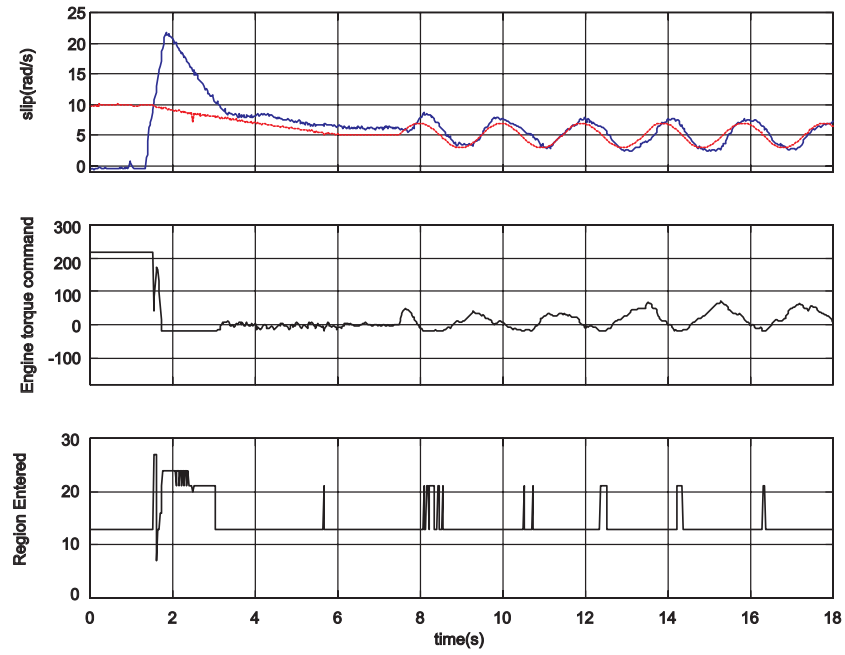


Fig. 8. Simulation results: MPC based on hybrid model , $\Delta\omega_d = 2$ rad/s

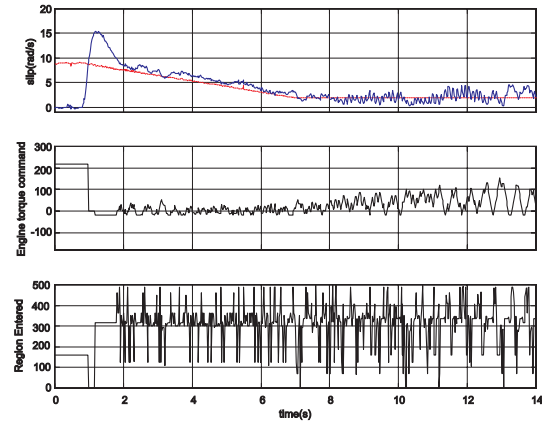


(a) Ramp and step slip reference

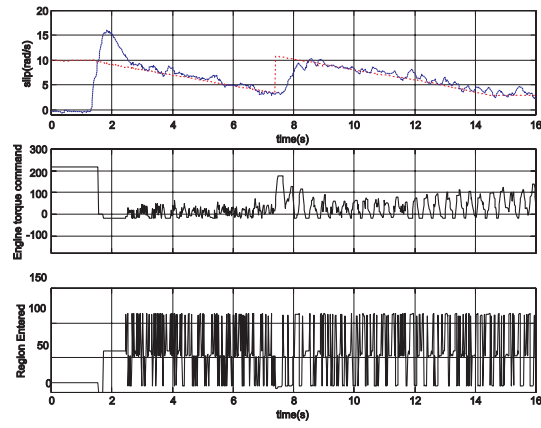


(b) Sinusoidal slip reference

Fig. 9. Simulation results: Linear MPC based on linear model 4



(a) Ramp and step slip reference, Hybrid Controller 1



(b) Ramp and step slip reference, Hybrid Controller 2

Fig. 10. Hybrid controllers