MODEL PREDICTIVE CONTROL OF NONLINEAR MECHATRONIC SYSTEMS: AN APPLICATION TO A MAGNETICALLY ACTUATED MASS SPRING DAMPER [1]

Stefano Di Cairano * Alberto Bemporad *
Ilya Kolmanovsky ** Davor Hrovat **

* *Dip. Ingegneria dell'Informazione, Università di Siena, Italy. Email:* dicairano,bemporad@dii.unisi.it.
** *Ford Motor Company, Dearborn, Michigan, USA. Email:* ikolmano,dhrovat@ford.com.

Abstract: Mechatronic systems in the automotive applications are characterized by significant nonlinearities and tight performance specifications further exacerbated by state and input constraints. Model Predictive Control (MPC) in conjunction with hybrid modeling can be an attractive and systematic methodology to handle these challenging control problems. In this paper, we focus on a mass spring damper system actuated by an electromagnet, which is one of the most common elements in the automotive actuators, with fuel injectors representing a concrete example. We present two designs which are based, respectively, on a linear MPC approach in cascade with a nonlinear state-dependent saturation, and on a hybrid MPC approach. The performance and the complexity of the two MPC controllers are compared.

Keywords: Model Predictive Control, Automotive Applications, Hybrid systems.

1. INTRODUCTION

Automotive actuators, such as fuel injectors, are examples of mechatronic systems (Hrovat *et al.*, 2000; Barron and Powers, 1996) that are characterized by tight operating requirements (such as high precision, low power consumption, fast transition time), significant nonlinearities, as well as input and state constraints which need to be enforced during the system operation. On the other hand, their dynamics may often be characterized by relatively low-dimensional models.

Model Predictive Control (MPC) (Qin and Badgwell, 2003) is a systematic feedback control design technique which determines the control input via receding horizon optimal control. Its main appeal is in being able to enforce pointwise-in-time constraints while providing the control designer with direct capability to shape the transient response by adjusting the weights in the objective function being minimized. MPC controllers can handle continuous-valued and discrete-valued control inputs, accommodate system parameter changes or subsystem faults, as long as they are reflected in the model used for on-line optimization.

Automotive actuators can often be adequately characterized by low dimensional models, and in this case an explicit implementation of the MPC controller becomes possible (see e.g. (Giorgetti *et al.*, 2005)), whereby the solution is pre-computed off-line and its representation is stored for on-

[1] Work (partially) done in the framework of the HYCON Network of Excellence, contract number FP6-IST-511368.

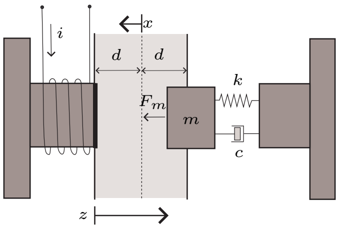


Fig. 1. The schematics of a magnetically actuated mass spring damper system.

line application. The on-line optimization is not required and the computational effort can be reduced to the point where the implementation of these control algorithms becomes feasible within stringent memory and chronometric constraints of automotive micro-controllers.

In this paper, we discuss and illustrate this approach in more detail in application to an electromagnetically actuated mass-spring damper system. Such a system arises very frequently in automotive actuation mechanisms (see (Hrovat *et al.*, 2000; Guzzella and Sciarretta, 2005) and references therein), including fuel injectors. In an actuation system of this kind, there is typically a moving mass which operates against a spring and a damper, while being controlled by a magnetic force from an electromagnetic coil. The force from the coil is unidirectional (i.e., the coil can only attract but not repel the mass), and this force decays inversely proportionally to the square of the distance between the mass and the coil. This force is also proportional to the square of the current, which is controlled to the desired value by an inner loop controller. By neglecting the much faster electrical dynamics, in this paper we consider a second order mechanical system with the magnetic force being the control input. The effect of electrical dynamics is incorporated in an extension of this work, see (Di Cairano *et al.*, 2006).

We present two controller designs, a linear MPC, where a nonlinear constraint on the magnetic force is neglected in the design and subsequently enforced by cascading a nonlinear state-dependent saturation, and a hybrid MPC, which considers also the nonlinear force constraints through a piecewise affine approximation.

2. PHYSICAL MODEL AND CONSTRAINTS

We consider a linear model for the moving mass dynamics in the form

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\dfrac{k}{m} & -\dfrac{c}{m} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \dfrac{1}{m} \end{bmatrix} u(t), \tag{1}$$

where the states are, respectively, the position [m] of the moving mass x_1, and the velocity of the moving mass x_2 [m/sec]. The magnetic force [N] acting on the moving mass is denoted by u. The neutral position of the spring corresponds to $x_1 = 0$ while the coil is located at $x_1 = x_{1c} = 4 \cdot 10^{-3}$ m. The magnetic force is given by

$$u(t) = k_a \frac{i^2(t)}{(z(t) + k_b)^2},$$

where $z = x_{1c} - x_1$ is the distance between the moving mass and the coil, i is the current [A] through the coil and k_a, k_b are constant parameters.

In this paper we consider the case when the inner-loop controller is capable of controlling the current to the desired set-point on a faster time-scale than the mechanical dynamics of the system, so that we may view $u(t)$ as the control input in (1). In order to realize a given $u(t)$ by the current, $i(t)$, $u(t)$ must satisfy the following constraints

$$u(t) \geq 0, \tag{2a}$$

$$u(t) \leq k_a \frac{i_{max}^2}{(z(t) + k_b)^2} \tag{2b}$$

where $i_{max} = 10$ A denotes the maximum current through the coil. The first of these constraints reflects the fact that the magnetic force is unidirectional (i.e., the electromagnet can only attract and not repel the moving mass). The second constraint is due to the limitations of the current which can be delivered by the power electronic circuits in the system. Note that (2) defines a non-convex set in the input+state space, being the intersection of a halfspace and of the hypograph of a convex function.

Besides (2), additional constraints are introduced to bound the moving mass position between the coil and a symmetric stop on the other end, i.e., at $-4 \cdot 10^{-3}$ m,

$$-4 \cdot 10^{-3} \leq x_1(t) \leq 4 \cdot 10^{-3}. \tag{3}$$

Even though the moving mass cannot penetrate into the coil or into the symmetric stop on the other end, these constraints have to be imposed explicitly to preserve the validity of model (1). If this is not done, undesirable moving mass bouncing can create noise and increase wear of the parts.

In a number of practical applications it is actually desirable to control the moving mass so that it is positioned against the coil with $x_1 = x_{1c}$. As the moving mass approaches the coil, its velocity needs to be carefully controlled to avoid high collision velocities (this is called *soft-landing*). In addition, maintaining the velocity of approach relatively low reduces the disturbance to the current control loop. In this paper we use the constraint

$$-\varepsilon - \beta(x_{1c} - x) \leq \dot{x} \leq \varepsilon + \beta(x_{1c} - x) \tag{4}$$

to limit the velocity of approach, where $\beta = 2500\ \mathrm{s}^{-1}$ and $\varepsilon = 0.2$ m/s. The maximum allowed absolute value of the moving mass velocity at the neutral position (i.e., at $x_1 = 0$) is $v_{max} = 10.2$ m/s and it is 0.2 m/s at the contact position with the coil (i.e., at $x_1 = x_{1c}$).

To facilitate the application of Model Predictive Control, system model (1) is translated into the discrete-time model

$$\begin{bmatrix} x_1(t+1) \\ x_2(t+1) \end{bmatrix} = \begin{bmatrix} 0.89 & 4.7\cdot10^{-4} \\ -42.77 & 0.85 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 7\cdot10^{-7} \\ 2.85\cdot10^{-3} \end{bmatrix} u(t), \quad (5)$$

where we have considered the sampling period $T_s = 0.5$ ms. With the discrete-time approach, constraints (2), (3) and (4) are enforced only at the sampling instants kT_s, $k \in \mathbb{N}$, for system (5).

3. LINEAR MODEL PREDICTIVE CONTROL

Model Predictive Control (Qin and Badgwell, 2003) is an optimization-based closed-loop control strategy in which pointwise-in-time design constraints on system's state, input and output can be explicitly embedded into the controller and, at the same time, it is a closed-loop strategy, since at each time instant the optimization is repeated using the most recent measurements.

The MPC strategy is based on the solution of the optimal control problem

$$\begin{aligned} \min_{\{y_k,u_k\}_{k=0}^{N-1}} \quad & \sum_{k=0}^{N_J-1} (y_k - r_y(t))' Q_y (y_k - r_y(t)) + \\ & \Delta u_k' Q_{\Delta u} \Delta u_k \\ \text{subject to} \quad & y_{\min} \le y_k \le y_{\max},\ k = 1, ..., N_C \\ & u_{\min} \le u_k \le u_{\max},\ k = 0, ..., N_U \\ & \Delta u_{\min} \le \Delta u_k \le \Delta u_{\max},\ k = 0, ..., N_U \\ & \Delta u_k = 0,\ k \ge N_U \\ & x_{k+1} = Ax_k + Bu_k \\ & y_k = Cx_k + Du_k,\ k = 0, \ldots, N_J - 1 \end{aligned} \quad (6)$$

where $\Delta u_k = u_k - u_{k-1}$, $u_{-1} = u(t-1)$ is the previous input, and $r_y(t)$ is the output reference at time step t. N_J is the prediction horizon along which performance is computed, N_C is the horizon along which the output constraints are enforced, and N_U is the number of free control actions, so that $N_U \le N_J$ and $u_k = u_{N_U}$, $\forall k = N_u + 1, \ldots, N_J$ [2].

The MPC algorithm can be summarized as follows: at each sampling instant t

(1) Set $x_0 = x(t)$.
(2) Solve (6) obtaining $u^*(x(t)) = [u_0^*, \ldots, u_{N-1}^*]$.
(3) Apply the input $u(t) = u_0^*$ and discard the remaining elements of $u^*(x(t))$.

The complexity of the MPC algorithm clearly depends on the structure of the optimization problem. In particular, if the system dynamics and the design constraints are linear and Problem (6) involves only continuous variables, the MPC algorithm requires, at each time step t, the solution of a Quadratic Program (QP), for which solution algorithms of polynomial complexity exist. On the other hand if some variables in Problem (6) are integer-valued, which is the case when the system model in (6) is a hybrid model, mixed-integer programming (MIP) techniques are required, which have combinatorial complexity.

When designing model-based control systems, there is a natural trade-off between model complexity and computation required. In particular, the more complex (and presumably accurate) is the model, the more complex Problem (6) may become. In view of this trade-off, in the sequel we first design an MPC controller, disregarding constraint (2b). Since dynamics (5) are linear and constraints (2a), (3), and (4) are also linear, the MPC algorithm requires the solution of QP only. If constraint (2b) is almost never active, the resulting MPC controller, cascaded by a state-dependent input-saturation, may be sufficient for adequately controlling the system, and the controller based on linear MPC solution can be simple and fast. On the other hand, if constraint (2b) is often active, the predicted trajectory will largely differ from the actual one, because of the unmodeled state-dependent input saturation. In the latter case, the system performance will most likely be degraded.

For the electromagnetically actuated mass-spring-damper system, the linear-MPC controller was designed using the Hybrid Toolbox (Bemporad, 2003). To make the moving mass position track a given reference signal and to enforce the constraints in (4) as output constraints, we define the output equation

$$y(t) = \begin{bmatrix} 1 & 0 \\ 2500 & 1 \\ -2500 & 1 \end{bmatrix} x(t) \quad (7)$$

Accordingly, we set

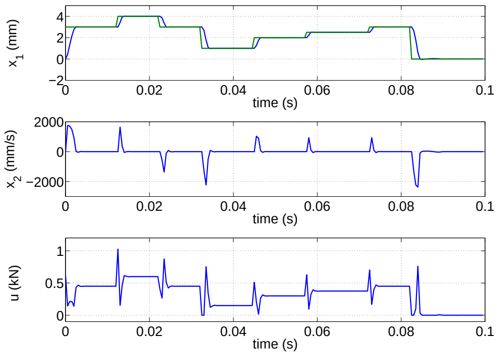
$$\begin{aligned} & Q_y = \begin{bmatrix} 10^4 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix},\ Q_u = 10^{-10}, \\ & y_{\min} = \begin{bmatrix} -4\cdot10^{-3} \\ -\infty \\ -10.2 \end{bmatrix},\ y_{\max} = \begin{bmatrix} 4\cdot10^{-3} \\ 10.2 \\ +\infty \end{bmatrix}, \\ & u_{\min} = 0,\ u_{\max} = 10^4,\ \Delta u_{\min} = -\Delta u_{\max} = \infty, \\ & N_J = 30,\ N_C = 5,\ N_U = 3, \end{aligned}$$

and use the linear dynamic model (5), (7) as prediction model.

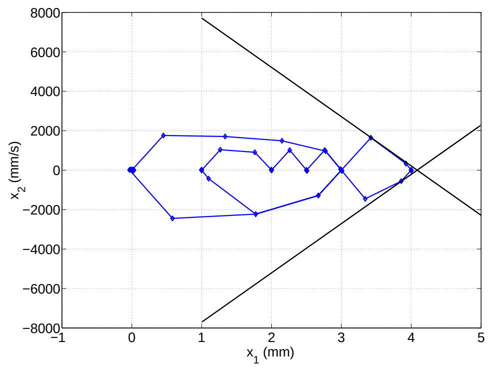
Figure 2 shows the behavior of closed-loop formed by the linear model and the MPC controller when tracking a desired reference profile over a

[2] In the MPC literature and in many MPC algorithms usually $N_C = N_J = N_U = N$.

simulation time interval of 0.1 seconds from the initial state $x(0) = \left[\begin{smallmatrix}0\\0\end{smallmatrix}\right]$. Figure 2(a) reports the position, velocity and input profiles with respect to time, and Figure 2(b) reports the phase plane in which satisfaction of velocity constraint (4) is shown. This velocity constraint only becomes active near the contact position x_{1c}. Because the controller cannot provide quick decelerations due to unidirectionality of the magnetic force, it keeps the constraint (4) inactive in parts of the trajectory away from the contact point.



(a) State and input trajectories.



(b) Phase-plane trajectories.

Fig. 2. Ideal linear MPC simulation.

Figure 3 shows the behavior of the closed-loop system when a position-dependent saturation block, enforcing constraint (2b), is cascaded with the MPC controller. The performance clearly degrades, especially when the reference is decreasing. The reason for such degradation is that the linear MPC controller does not recognize that braking the mass at large distances away from the coil is impossible because of the state-dependent input saturation. This is in fact seen from the input plot in Figure 3 where the dashed line corresponds to the output of the MPC controller, while the solid line corresponds to the output of the saturation block.

To avoid wide oscillations and long settling periods, the saturation constraint (2b) should be taken into account in the MPC setup. Unfortunately, (2b) is a nonconvex constraint that cannot

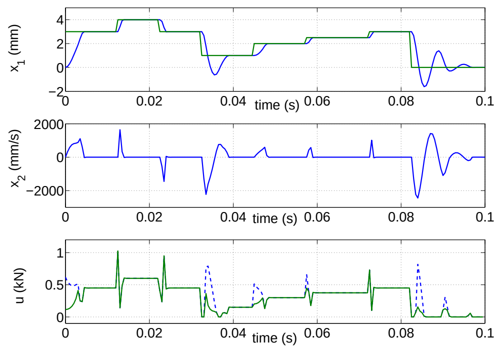


Fig. 3. Effects of saturation (2b).

be handled by standard linear MPC. Next section shows how such a constraint can be handled by a hybrid MPC approach.

4. HYBRID MODEL PREDICTIVE CONTROL

We approximate constraint (2b) by a (continuous) piecewise linear function $f(x_1) = r_i x_1 + q_i$, if $x_1 \in [\bar{x}_i, \bar{x}_{i+1})$, $i = 0, \dots \ell$ where $\{\bar{x}_i\}_{i=1}^{\ell-1}$ are the breakpoints of the function profile. In this paper we consider a piecewise linear approximation with three segments ($\ell = 3$), where the breakpoints are $\bar{x}_0 = -x_{1c}$, $\bar{x}_1 = 1.58 \cdot 10^{-4}$ m, $\bar{x}_2 = 1.82 \cdot 10^{-3}$ m and $\bar{x}_3 = x_{1c}$. Next, we introduce two binary variables $\delta_1, \delta_2 \in \{0, 1\}$ defined by the logical conditions

$$\begin{array}{l}[\delta_1 = 1] \leftrightarrow [x_1 \leq \bar{x}_1] \\ [\delta_2 = 1] \leftrightarrow [x_1 \leq \bar{x}_2]\end{array} \tag{8}$$

and two continuous variables $z_1, z_2 \in \mathbb{R}$ defined by the logical conditions

$$z_1 = \begin{cases} (r_1 - r_2)x_1 + (q_1 - q_2) & \text{if } \delta_1 = 1 \\ 0 & \text{otherwise} \end{cases} \tag{9a}$$

$$z_2 = \begin{cases} r_2 x_1 + q_2 & \text{if } \delta_2 = 1 \\ r_3 x_1 + q_3 & \text{otherwise,} \end{cases} \tag{9b}$$

and impose that

$$u \leq z_1 + z_2, \tag{10}$$

where clearly $z_1 + z_2 = f(x_1)$. Constraints (2a), (3), (4), (8), (9), (10), together with (5), are easily modeled in HYSDEL (Torrisi and Bemporad, 2004), and the equivalent Mixed Logical Dynamical (MLD) hybrid model (Bemporad and Morari, 1999)

$$x(k+1) = Ax(k) + B_1 u(k) + B_2 \delta(k) + B_3 z(k), \tag{11a}$$

$$E_2 \delta(k) + E_3 z(k) \leq E_1 u(k) + E_4 x(k) + E_5, \tag{11b}$$

corresponding to the saturated magnetic actuator is obtained, where the matrices A, B_i, $i = 1 \dots 3$, E_j, $j = 1, \dots 5$, are generated automatically using the Hybrid Toolbox (Bemporad, 2003).

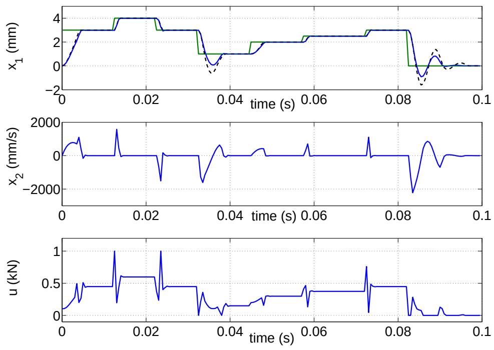
The hybrid MPC optimization problem is formulated as

$$\min_{\{u_k\}_{k=0}^{N-1}} \Delta x_N^T Q_N \Delta x_N + \sum_{k=0}^{N-1} \Delta x_k^T Q_x \Delta x_k + u_k Q_u u_k \tag{12a}$$

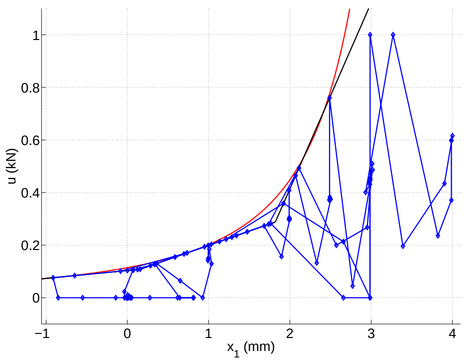
$$\text{subject to MLD dynamics (11),} \tag{12b}$$

where $Q_x = Q_N = \left[\begin{smallmatrix} 2\cdot 10^6 & 0 \\ 0 & 0 \end{smallmatrix}\right]$, $Q_u = 10^{-7}$, $N = 3$, and $\Delta x_1(k) = x_1(k) - r_y$. Because of the binary variables δ, the hybrid MPC strategy (12) requires the solution of mixed-integer quadratic programs. Note that only two binary variables are considered for each prediction step, so that the resulting optimization problem is of very small size.

The resulting closed-loop trajectories for the simulation scenario described in Section 3 when the hybrid MPC is applied are reported in Figure 4.



(a) State and input trajectories.



(b) Nonlinear force constraint, its PWL approximation, and input values generated by hybrid MPC.

Fig. 4. Closed-loop system response using the hybrid MPC controller (12).

Note that the PWL approximation (8), (9) is a lower bound to the maximum force profile, so that the force generated by the hybrid MPC algorithm never exceeds the saturation limits. With respect to the simulation of the linear MPC cascade by the saturation block (reported as dashed line in the position trajectory plot in Figure 4) we note that the system reacts a little slower when starting from the neutral position $x_1(0) = 0$, with null velocity $x_2(0) = 0$. This is the effect of the conservative approximation of the force constraint. While such negative effect can be eliminated by introducing a more refined approximation, the positive effects of the hybrid MPC controller are clear when the reference decreases. Both the overshoot and the settling period are reduced, because the controller is now aware of the limited available force and it provides the braking action in the region where a larger magnetic force is available.

MPC controller	Cumulative position error (mm^2)	Input energy (kN^2)
Linear (ideal)	51.4679	29.1918
Linear saturated	97.8608	26.6314
Hybrid	83.1005	26.6588

Table 1. Comparison of the three MPC scenarios

Table 1 compares the cumulative square position error, $\sum_k (x_1(k) - r(k))^2$, and cumulative square inputs (=actuator's energy), $\sum_k u(k)^2$, for the different MPC control scenarios. The tracking performance clearly degrades from the linear MPC controller in the ideal case of no force saturation, to the saturating one, while the hybrid controller has better performance (15%) with respect to the linear-saturated one, despite the slightly conservative approximation of constraint (2b). Moreover one must consider that a certain component of the tracking error is due to the one-step delay in reacting to reference changes, due to the non-anticipative implementation of the MPC algorithms. Such an error, that with respect to data in Table 1 has a value of 25.5, is independent of the controller applied, and thus should not be considered in comparing performances. Following this reasoning, the increase of net performance of the hybrid MPC algorithm is about 20% with respect to the linear-saturated one.

5. EXPLICIT IMPLEMENTATION OF THE CONTROLLER

The implementation of the MPC controllers described in the previous sections in a typical automotive micro-controller with the sampling time $T_s = 0.5$ ms can be very difficult because of the time required for the online solution of the underlying optimization problem. With the motivation to complete off-line a large part of the computations we developed explicit versions of the MPC controllers.

In (Bemporad *et al.*, 2002) it is shown that the solution to Problem (6) can be obtained as a function of the parameters x_0 and r_y (i.e., the actual state and output reference) by using multi-parametric quadratic programming (mp-QP). Using the mp-QP solver in the Hybrid Toolbox, we obtain an explicit feedback law $u(x, r_y)$ in continuous piecewise affine form consisting of 80 regions, which can be evaluated on-line very quickly. The mp-QP algorithm also returns the value function $V(x, r_y) = J^*(x, r_y)$, which is a piecewise quadratic function.

It must be stressed that the implicit MPC controller and the explicit one produce the same results, but there is a difference in the amount of computation required at each sampling step. More specifically, this difference is between the solution of an online optimization problem versus the evaluation of a set of inequalities and the computation of an affine state feedback term.

Figure 5 shows a section of the three-dimensional polyhedral partition of the explicit linear MPC controller, for $r_y = 0$. There is an affine state feedback controller associated to each region in the partition. Figure 5 also shows the state trajectory superimposed over the polyhedral partition.

In the case of hybrid MPC, we use the algorithm of (Bemporad, 2003) to obtain a representation of the MPC controller as a set of (possibly overlapping) piecewise affine controllers. During the on-line operation, at each step for each controller the value function is evaluated, and the input corresponding to the minimum cost is applied. Thus the explicit hybrid MPC solution involves the additional operation of comparing online the value functions. In addition, the number of regions increases to 671 regions, thus the controller requires a larger storage memory in the micro-controller and a larger number of comparison operations to find the active region.

6. CONCLUSIONS

We have compared two MPC solutions to the problem of controlling an electromagnetic actuator: one is based on linear MPC and handles one of the constraints via a-posteriori saturation; the other one is based on a hybrid model of the system and accounts for all the constraints in the design phase.

The hybrid MPC, which takes into account a piecewise linear approximation of the position dependent force constraint, achieves better performance than the linear MPC solution with superimposed saturation. Its main drawback is in higher complexity of the controller. Thus, the ultimate choice between these two MPC solutions can only be made once considering available computing resources and the aggressiveness of performance specifications.

An electromagnetically actuated mass-spring damper laboratory experiment is currently under construction at the University of Siena.

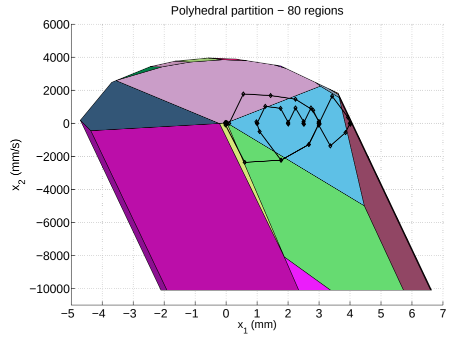


Fig. 5. Section of the linear explicit controller partitions obtained for $r_y = 0$.

REFERENCES

Barron, M.B. and W.F. Powers (1996). The role of electronic controls for future automotive mechatronic systems. *IEEE/ASME Transactions on Mechatronics* **1**(1), 80–88.

Bemporad, A. (2003). *Hybrid Toolbox – User's Guide*. http://www.dii.unisi.it/hybrid/toolbox.

Bemporad, A. and M. Morari (1999). Control of systems integrating logic, dynamics, and constraints. *Automatica* **35**(3), 407–427.

Bemporad, A., M. Morari, V. Dua and E.N. Pistikopoulos (2002). The explicit linear quadratic regulator for constrained systems. *Automatica* **38**(1), 3–20.

Di Cairano, S., A. Bemporad, I.V. Kolmanovsky and D. Hrovat (2006). Model predictive control of a magnetically actuated mass-spring-damper for automotive applications. In: *Proc. 45th IEEE Conf. on Decision and Control*. San Diego, CA. Submitted for publication.

Giorgetti, N., A. Bemporad, E. H. Tseng and D. Hrovat (2005). Hybrid model predictive control application towards optimal semi-active suspension. In: *Proc. IEEE Int. Symp. on Industrial Electronics*. Dubrovnik, Croatia.

Guzzella, L. and A. Sciarretta (2005). *Vehicle Propulsion Systems Introduction to Modeling and Optimization*. Springer Verlag.

Hrovat, D., J. Asgari and M. Fodor (2000). Automotive mechatronic systems. In: *Mechatronic Systems, Techniques and Applications, Volume 2*. pp. 1–98. Gordon and Breach Science.

Qin, S.J. and T.A. Badgwell (2003). A survey of industrial model predictive control technology. *Control Engineering Practice* **93, no. 316**, 733–764.

Torrisi, F.D. and A. Bemporad (2004). HYSDEL — A tool for generating computational hybrid models. *IEEE Trans. Contr. Systems Technology* **12**(2), 235–249.