**Proceedings of the
46th IEEE Conference on Decision and Control
New Orleans, LA, USA, Dec. 12-14, 2007**

Hybrid Modeling and Control of a Multibody Magnetic Actuator for Automotive Applications

Alberto Bemporad†, Stefano Di Cairano†, Ilya Kolmanovsky‡ and Davor Hrovat‡

Abstract— **This paper investigates model predictive control (MPC) techniques based on hybrid models for a multi-mass magnetic actuator. The actuator has four operating modes depending on the mutual interaction of two moving masses and is modeled as a hybrid dynamical system. The control law optimizes a performance index and enforces several types of constraints: soft-landing during collisions to reduce mechanical wear, current limits and consequent position-dependent nonlinear bounds on the available magnetic force, and restrictions on the positions of the moving masses. Two different approaches are considered: (i) a hybrid MPC design based on the full two-mass model, and (ii) a switched MPC control design, which switches between two simpler hybrid MPC controllers, one for the case in which the masses are moving in contact, and the other case where the masses are decoupled, commanded by a simple switching logic. Simulation results and performance comparisons of the two control schemes are discussed.**

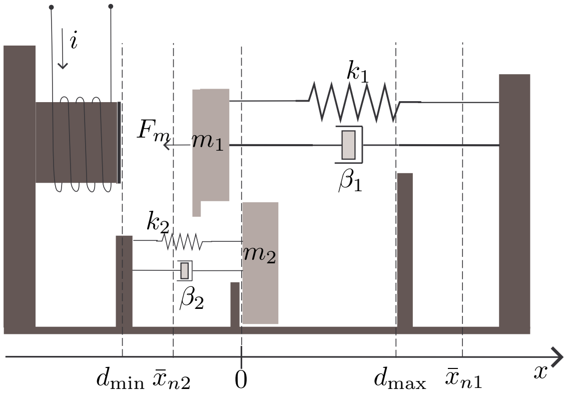

Fig. 1. Multibody magnetic actuator schematics.

I. Introduction

Magnetic actuators represent an effective technology for high performance actuation mechanisms with high precision and high reliability, in part due to the reduction in mechanical stresses and friction that they provide. A large interest in magnetic actuators exists in the automotive industry [1], [2] to control different types of devices that are operated several times per second. On the other hand, controlling magnetic actuators is a challenging task. A magnetic actuator is composed of an electrical and magnetic subsystem, and of a mechanical subsystem. The overall dynamics are nonlinear and stringent constraints must be enforced to ensure the correct operation of the controlled device. Model Predictive Control (MPC) [3] is a promising technique for control of such devices. The MPC strategy is based on the solution of a constrained optimization problem, which enforces constraints on state and input variables, and optimizes a performance index related to the desired tracking properties of the closed-loop dynamics. The use of MPC in applications has been traditionally limited by the computation time required to solve the optimization problem online. Explicit MPC [4], [5] overcomes this limitation by precomputing the solution of the optimization problem and providing the optimal control law in an easily implementable piecewise affine form.

Previous research [6], [7] showed that MPC can be satisfactorily applied to magnetic actuators in which the mechanical component is modeled as a single mass-spring-damper system. By decoupling the nonlinear (faster) electrical and magnetic dynamics from the mechanical ones, the control problem was solved by a hierarchical controller. More specifically, the control system can be composed of an inner-loop controller regulating the electrical subsystem to track the reference provided by a higher level MPC controller that plans the dynamics of the mechanical subsystem. In [6] it was shown that the limitations on the available magnetic force can be modeled as a state-dependent piecewise affine saturation function, thereby permitting the MPC controller to be designed using hybrid MPC methodologies [5], [8]. In [7] it was shown that a single hybrid MPC controller can be designed for the full system dynamics, obtaining superior performance, but at a price of an excessive controller complexity.

To move beyond the assumption of a single mass-spring-damper mechanical subsystem, in this paper we consider a more complex control problem, relevant to a wider class of actuators, in which the actuator is composed by two possibly impacting masses (see Figure 1). Each mass is subject to a spring force and damping, one of them (m_1) can also be attracted by the magnetic force generated by a coil. The actuated mass m_1 is used to push the other mass m_2, which represents the physical system to be displaced. Mass m_1 models a slider attracted by the coil and able to move mass m_2. Compared to the single-mass case [6], [7], the description of the overall dynamics is much more involved because there are different operating modes. The two masses

This work was supported by the HYCON Network of Excellence (contract number FP6-IST-511368), by Ford Motor Company, and by the Italian Ministry for University and Research (MIUR) under project "Advanced control methodologies for hybrid dynamical systems".

† A. Bemporad and S. Di Cairano are with Dipartimento di Ingegneria dell'Informazione, Università di Siena, Italy `bemporad,dicairano@dii.unisi.it`

‡ I. Kolmanovsky and D. Hrovat are with Ford Motor Company, Dearborn, Michigan, USA. `ikolmano,dhrovat@ford.com`

1-4244-1498-9/07/$25.00 ©2007 IEEE.

may be in contact between them and/or with a mechanical stop, or in free motion.

Hybrid dynamical systems [9], which allow the definition of different discrete operating modes, are a natural framework to model such a multi-mass actuation system. To mitigate the critical effect of violent impacts, soft-landing constraints are enforced in the design. These reduce the relative velocity (mass-to-mass or mass-to-stop) when the impact is approaching. In this way, not only mechanical wear and noise are prevented, but also most impacting phenomena become negligible (for a thorough treatment of continuous-time mechanical systems with impacts the reader is referred to [10], [11], as well as to the SICONOS platform [12], a software tool for modeling, simulation, control and analysis of non-smooth dynamical systems).

This paper is organized as follows. The magnetic actuation system is modeled in Section II as a discrete-time hybrid mechanical system with four different operating modes. Based on such a model, Section III is devoted to the design of a hybrid MPC controller which enforces all the aforementioned constraints, and provides closed-loop simulation results. A simpler approach is proposed in Section IV based on two switched single-mass MPC controllers, one designed for the two masses in contact, the other for the two masses in free motion, and both coordinated by a simple switching logic. The reported simulation results show that the performance gets slightly worsened, but the complexity of the control law is reduced.

II. Hybrid Model

The multibody magnetic actuator considered in this paper is decomposed into two subsystems: a mechanical one based on two (possibly impacting) masses, and an electromagnetic one generating the magnetic force that actuates the system.

A. Mechanical Subsystem

The schematics of the magnetic actuator are shown in Figure 1, where $d_{\max} = 4$ mm and $d_{\min} = -0.5$ mm. The system is composed of two masses $m_1 = 0.08$ Kg and $m_2 = 0.07$ Kg. Mass m_1, referred to as the "active mass", is subject to the magnetic force F_m generated by the coil, which can attract, but not repel, the mass. It is also affected by a spring with stiffness $k_1 = 1.5 \cdot 10^5$ N/m and with neutral position $\bar{x}_{n1} > d_{\max} > 0$, and by a damper with friction coefficient $\beta_1 = 15$ N s/m. Mass m_2, referred to as the "passive mass", is subject to the effects of a spring with stiffness $k_2 = 1.5 \cdot 10^5$ N/m and neutral position $\bar{x}_{n2} < 0$, and of a damper with friction coefficient $\beta_2 = 15$ N s/m. A mechanical stop prevents m_2 to move to a negative position, so that a preloaded spring force $F_{PL} = k_2 \bar{x}_{n2} = 100$ N is present at the position $x = 0$. The two masses can interact: The force that the active mass exercises on the passive mass is indicated by $F_{1,2}$, while the reaction force of the passive mass on the active one is indicated by $F_{2,1} = -F_{1,2}$. Because of the preload force of the spring connected to m_2 at $x_2 = 0$, a force $F_{1,2} > -F_{PL}$ is necessary to move m_2. In conclusion, the mechanical system is described by the equations

$$\begin{aligned} \ddot{x}_1 &= \frac{1}{m_1}\Big(- k_1(x_1 - \bar{x}_{n1}) - \beta_1 \dot{x}_1 - F_m + F_{2,1}\Big) \\ \ddot{x}_2 &= \frac{1}{m_2}\Big(- k_2(x_2 - \bar{x}_{n2}) - \beta_2 \dot{x}_2 + F_{1,2}\Big), \end{aligned} \tag{1}$$

where $F_{2,1} = -F_{1,2}$ and $F_m \geq 0$, as m_1 can be moved only towards the coil placed in the negative direction. When the two masses are not in contact, the interaction forces satisfy $F_{1,2} = F_{2,1} = 0$ and the dynamics in (1) become decoupled.

Instead of modelling the complex dynamics during the interaction of the two masses ($F_{1,2},\ F_{2,1} \neq 0$) the system behavior is represented as a hybrid dynamical system with different operating modes, under the following assumption.

Assumption 1: The impact between m_1 and m_2 is totally inelastic, and immediately after the impact $\dot{x}_2 = \dot{x}_1$.
Assumption 1 is equivalent to say that m_1, m_2 stick and move together immediately after the impact and is useful to avoid modeling the transition from the decoupled to the coupled mass dynamics. Note that this simplified model will be used for control design, its purpose is not to reproduce accurately in simulation the behavior of the real system. Rather than complicating the prediction model and obtaining, as a result, a more complex control law, modeling errors will be dealt with by MPC feedback. Assumption 1 can also be argued to hold thanks to our control approach which assures soft-landing of the parts with no bounce, as it will be shown in Section II-C.

We consider four operating modes:

M1) m_1, m_2 move separately in free motion,
M2) m_1 moves freely, m_2 is stuck at the stop ($x_2 = 0$),
M3) m_1, m_2 are both stuck at the stop ($x_1 = x_2 = 0$),
M4) m_1, m_2 move together.

Mode M1 occurs when both masses are moving independently (no contact between them), and are therefore treated as separate bodies ($F_{1,2} = F_{2,1} = 0$ in (1)).

Mode M2 occurs when m_2 is blocked at the stop, so $x_2 = 0$, while m_1 is freely moving (hence, $x_1 < 0$). The equations are

$$\ddot{x}_1 = \frac{1}{m_1}\Big(- k_1(x_1 - \bar{x}_{n1}) - \beta_1 \dot{x}_1 - F_m\Big) \tag{2a}$$

$$x_2 = 0, \quad \dot{x}_2 = 0. \tag{2b}$$

Mode M3 occurs when both masses are blocked at the stop, which is possible when $F_{1,2} \leq F_{PL}$. The equations describing M3 are

$$x_1 = 0, \quad \dot{x}_1 = 0 \tag{3a}$$

$$x_2 = 0, \quad \dot{x}_2 = 0. \tag{3b}$$

Mode M4 occurs when the two masses are stuck together and move as a single body. The system equations become

$$\begin{aligned} \ddot{x}_1 = \frac{1}{m_1 + m_2}\Big(&- k_1(x_1 - \bar{x}_{n1}) \\ &-k_2(x_1 - \bar{x}_{n2}) - (\beta_1 + \beta_2)\dot{x}_1 - F_m\Big) \end{aligned} \tag{4a}$$

$$x_2 = x_1, \quad \dot{x}_2 = \dot{x}_1. \tag{4b}$$

$\delta_P, \delta_D \rightarrow$	0 0	0 1	1 0	1 1
$\delta_B, \delta_C \downarrow$				
0 0	M1	M1	M1	M1
0 1	M4	M1	M4	M1
1 0	M2	M2	M2	M2
1 1	M3	M3	M4	M4

TABLE I
MODE-SELECTION TABLE

By introducing the following binary variables

$$\begin{aligned}
[\delta_B = 1] \quad &\leftrightarrow \quad x_2 \leq 0, && \text{(5a)}\\
[\delta_C = 1] \quad &\leftrightarrow \quad x_1 \geq x_2, && \text{(5b)}\\
[\delta_P = 1] \quad &\leftrightarrow \quad -F_m - k_1(x_1 - \bar{x}_{n1}) - \beta_1 \dot{x}_1 - \\
&\qquad k_2(x_2 - \bar{x}_{n2}) > 0, && \text{(5c)}\\
[\delta_D = 1] \quad &\leftrightarrow \quad \frac{1}{m_1}\Big(- k_1(x_1 - \bar{x}_{n1}) - \beta_1 \dot{x}_1 - F_m \Big) < \\
&\qquad \frac{1}{m_2}\Big(- k_2(x_2 - \bar{x}_{n2}) - \beta_2 \dot{x}_2 \Big), && \text{(5d)}
\end{aligned}$$

the conditions for which exactly one of modes M1, M2, M3, M4 is active are defined by Table I.

The meaning of the binary variables in (5) is the following:

- $\delta_B = 1$ indicates that m_2 has reached a stop;
- $\delta_C = 1$ is verified when m_1 has reached m_2;
- $\delta_P = 1$ means that m_1 is able to push m_2 to beat the preload force $-k_2\bar{x}_{n2}$;
- $\delta_D = 1$ happens when the acceleration of m_1 is smaller than the acceleration of m_2, so that the two masses tend to separate.

According to Table I the following mutually exclusive Boolean variables

$$\begin{aligned}
c_1 &= (\neg\delta_B \wedge \neg\delta_C) \vee (\neg\delta_B \wedge \delta_C \wedge \delta_D) && \text{(6a)}\\
c_2 &= \delta_B \wedge \neg\delta_C && \text{(6b)}\\
c_3 &= \delta_B \wedge \delta_C \wedge \neg\delta_P && \text{(6c)}\\
c_4 &= (\neg\delta_B \wedge \delta_C \wedge \neg\delta_D) \vee (\delta_B \wedge \delta_C \wedge \delta_P) && \text{(6d)}\\
\text{TRUE} &= c_1 \oplus c_2 \oplus c_3 \oplus c_4 && \text{(6e)}
\end{aligned}$$

identify the corresponding active mode.

In order to setup the MPC algorithm described in Section III, the dynamics in each of the operating modes M1, M2, M3, M4 are discretized with sampling period $T_s = 0.2$ ms, obtaining the switched affine dynamics

$$x(k+1) = A_i x(k) + B_i F_m(k) + f_i, \; i = 1, \ldots, 4 \qquad (7)$$

where $x = \begin{bmatrix} x_1 \\ \dot{x}_1 \\ x_2 \\ \dot{x}_2 \end{bmatrix}$. Note that while smaller sampling periods could have been used, this would increase the average computational load per time unit of the model-based control laws proposed in the following sections.

Together with the event conditions (5) and the mode selection rules of Table I, the switched affine dynamics (7) form a Discrete Hybrid Automaton (DHA) [13].

i	$\bar{x}_1^i$ [mm]	p_i [hN/mm]	q_i [hN]
0	−0.50	−12.97	40.02
1	2.14	−6.55	26.29
2	2.84	−2.97	16.12

TABLE II
PARAMETERS OF PIECEWISE AFFINE APPROXIMATION (11) OF (10)

B. Electrical Subsystem

The electrical subsystem is responsible for converting the primary system input, the voltage, into magnetic force. The magnetic force is a function of the current feed into the coil

$$F_m = \frac{k_a i^2}{(\ell + k_b)^2}, \qquad (8)$$

where ℓ is the distance between the active mass and the coil ($\ell = x_1 + 0.5$ mm), and k_a and k_b are physical constants ($k_a = 3.74 \cdot 10^{-5}$ Nm^2/A^2, $k_b = 4.16 \cdot 10^{-5}$ m). The relation between current i and voltage V for the circuit considered is given by Faraday's law

$$\dot{\lambda} = V - Ri, \qquad (9)$$

where λ is the magnetic flux linkage, which is in its turn a function of i. Thus, the equation that binds V and F_m is nonlinear and it would increase the complexity of the problem. However, in a situation where the electrical dynamics are faster than the mechanical ones [6], it is possible to decouple the control of the electrical subsystem from the control of the mechanical subsystem. Accordingly, a model predictive controller based on the mechanical model generates a force reference which is fed into the controller for the electrical subsystem to produce the current which results in the specified force, according to a hierarchical control strategy. The controller for the electrical subsystem can be a standard nonlinear controller that is able to make the force track the desired force F_m. Such a hierarchical control approach is also used here, under the condition that the commanded force F_m should be reproducible by the electrical subsystem: Given the maximum current $i_{max} = 22$ A due to power electronics limitations, the MPC controller must enforce the saturation constraint

$$0 \leq F_m \leq \frac{k_a i_{max}^2}{(\ell + k_b)^2}, \qquad (10)$$

which is a nonconvex constraint. Note that (10) is a state-dependent saturation. Following the approach of [6] we approximate (10) through the piecewise affine constraint

$$0 \leq F_m \leq p_i x_1 + q_i, \text{ if } x_1 \in (\bar{x}_1^i, \bar{x}_1^{i+1}], \; i = 0, \ldots, 2 \quad (11)$$

where $\bar{x}_1^0 = -0.5$ mm, $\bar{x}_1^3 = 4$ mm and $\bar{x}_1^1 = 2.14$ mm, $\bar{x}_1^2 = 2.84$ mm are the breakpoints at which the linear approximation changes. The full set of parameters of the piecewise affine approximation is reported in Table II.

The PWA saturation function in (11) is modeled by introducing two auxiliary binary variables

$$\begin{aligned}
[\delta_{m1} = 1] \quad &\leftrightarrow \quad [x_1 \leq \bar{x}_1^1] && \text{(12a)}\\
[\delta_{m2} = 1] \quad &\leftrightarrow \quad [x_1 \leq \bar{x}_1^2] && \text{(12b)}
\end{aligned}$$

with $[\delta_{m1}=1] \rightarrow [\delta_{m2}=1]$, and two auxiliary continuous variables

$$z_{m1} = \begin{cases} (m_0 - m_1)x_1 + (q_0 - q_1) & \text{if} \quad \delta_{m1} = 1 \\ 0 & \text{if} \quad \delta_{m1} = 0 \end{cases} \tag{13a}$$

$$z_{m2} = \begin{cases} m_1 x_1 + q_1 & \text{if} \quad \delta_{m2} = 1 \\ m_2 x_1 + q_2 & \text{if} \quad \delta_{m2} = 0 \end{cases} \tag{13b}$$

so that the saturation constraint can be expressed as

$$0 \le F_m \le z_{m1} + z_{m2}. \tag{14}$$

C. Soft Landing

To prevent mechanical wear and noise due to violent impacts, and at the same time to validate our assumption of neglecting impact dynamics and its effects on velocities, we want to impose a soft-landing constraint in the model, requiring that the velocity gets smaller and smaller when approaching a contact between m_1 and m_2 and between m_2 and the stop at $x_2 = 0$. Accordingly, the constraints are defined by

$$\begin{aligned} &[c_1 = 1] \vee [c_2 = 1] \rightarrow [\dot{x}_1 - \dot{x}_2 \le k_{sl1}(x_2 - x_1) + h_{sl1}] \\ &[c_1 = 1] \vee [c_4 = 1] \rightarrow [-\dot{x}_2 \le k_{sl2}x_2 + h_{sl2}] \end{aligned} \tag{15}$$

where $k_{sl1}, k_{sl2}, h_{sl1}, h_{sl2} \ge 0$. The negative sign in front of the second equation in (15) is due to the fact that when m_2 is approaching the stop, the velocity $\dot{x}_2$ is negative. Note that we do not consider here the effects of the impacts on the velocity after the collision. By the momentum conservation law the velocity v of $m_1 + m_2$ after the impact satisfies $m_1 v_1 + m_2 v_2 = (m_1 + m_2)v$, where v_1, v_2 are the velocities of m_1, m_2 before the impact. In our model we assume instead $v = v_1$. Despite the fact that the conservation law could be enforced by introducing a *reset mode* [13], such an approach would complicate the model. Thus, we prefer to keep a modelling error, which is anyway small because the soft-landing constraints force the relative velocity $v_1 - v_2$ before the impact to be small, so that $v \approx v_1$.

Note that constraints (15) are related to the desired closed-loop behavior and do not lead to system failures if mildly violated. On the other hand, enforcing them ensures a certain robustness with respect to disturbance and modelling errors. Thus, they are treated as *soft constraints*, where "soft" here means that they may be violated, although such violations will be heavily penalized in the control setup of Section III. Soft constraining is achieved by introducing two variables

$$\chi_1 = \begin{cases} \dot{x}_1 - \dot{x}_2 - k_{sl1}(x_2 - x_1) - h_{sl1} & \text{if } c_1 = c_2 = 1 \\ 0 & \text{otherwise} \end{cases} \tag{16a}$$

$$\chi_2 = \begin{cases} -\dot{x}_2 - k_{sl2}x_2 - h_{sl2} & \text{if } c_1 = c_4 = 1 \\ 0 & \text{otherwise} \end{cases} \tag{16b}$$

and by imposing that

$$\chi_1 \quad \le \quad \rho \tag{17a}$$

$$\chi_2 \quad \le \quad \rho. \tag{17b}$$

where $\rho \ge 0$ is a *slack variable* that should ideally be zero to enforce the soft-landing constraint.

D. Discrete-time Hybrid Model

The DHA described by (5), (7) and Table I, together with the piecewise affine saturation constraint (11) and the soft-landing constraints (16), (17) are modeled through the modeling language HYSDEL [13]. The DHA is converted automatically by the Hybrid Toolbox [14] into a mixed logical dynamical (MLD) system form [8]

$$x(k+1) = B_3 z(k), \tag{18a}$$

$$E_2\delta(k) + E_3 z(k) \le E_1 u(k) + E_4 x(k) + E_5, \tag{18b}$$

where $u(k) = [F_m(k) \ \ \rho(k)]' \in \mathbb{R}^2$, $\delta(k) = [\delta_B,\ \delta_C,\ \delta_P,\ \delta_D,\ c_1,\ c_2,\ c_3,\ c_4,\ \delta_{m1},\ \delta_{m2}]' \in \{0,1\}^{10}$, and $z(k) \in \mathbb{R}^{15}$ are auxiliary real-valued variables required to express (7), (11) (as described in (13)), and (15). The inequalities (18b) include a big-M representation [8] of (5), (12), (13), and (16), a conversion of (6) into conjunctive normal form and then into linear inequalities [15], and inequalities (14), (17).

E. Position Constraints

Because of the system structure, the model is valid when

$$d_{\min} \le \quad x_1 \quad \le d_{\max} \tag{19a}$$

$$0 \le \quad x_2 \quad \le d_{\max} \tag{19b}$$

$$x_1 \quad \le x_2 \tag{19c}$$

where $d_{\min} = -0.5$mm, and $d_{\max} = 4$mm. Constraints (19) will be enforced by the control strategy proposed in the next section.

III. Hybrid Model Predictive Control

Consider the finite-time optimal control problem

$$\begin{aligned} \min_{\{u(k)\}_{k=0}^{N-1}} & (x(N) - r_x)^T Q_N (x(N) - r_x) + \\ & \sum_{k=0}^{N-1} (x(k) - r_x)^T Q_x (x(k) - r_x) + \\ & (u(k) - r_u)^T Q_u (u(k) - r_u) \end{aligned} \tag{20a}$$

$$\text{subject to : MLD dynamics (18)} \tag{20b}$$

$$\text{position constraints (19)} \tag{20c}$$

where $N = 3$ is the prediction horizon, $r_x = [r_{x1}\ 0\ r_{x2}\ 0]$ is a state reference vector, and $r_u = [r_{F_m}\ 0]'$ is a consistent input reference so that when $x = r_x$ the input to keep that state value has null cost. The position is expressed in mm, the velocity in mm/s, and the force in 10^2N. The optimizer contains both the input F_m and the slack variable ρ that softens the soft-landing constraints. The weights are $Q_x = Q_N = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$, $Q_u = \begin{bmatrix} 10^{-2} & 0 \\ 0 & 10^2 \end{bmatrix}$. The rationale behind the choice of the weights is that the primary objective is to have m_2 tracking its reference position r_{x2}, but we also want m_1 to track the same reference $r_{x1} = r_{x2}$. The reason for the latter is that when r_{x2} cannot be reached by m_2, that

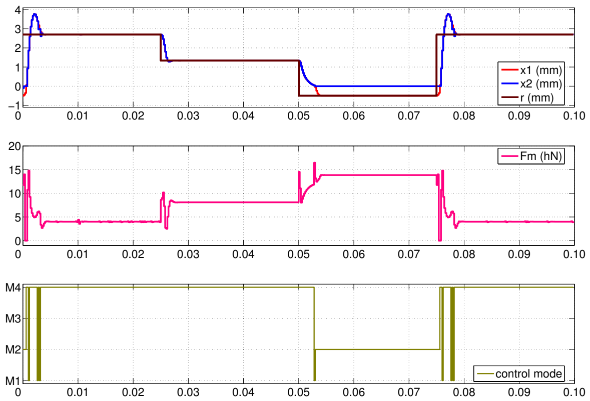

Fig. 2. Closed-loop hybrid MPC simulation. Time axis is expressed in seconds

is when $r_{x2} < 0$, it may be still useful to position m_1 on certain negative positions while m_2 is stopped at $x = 0$. The large weight 100 on ρ allows ρ to be sensibly greater than zero only when a violation of the soft-landing constraints is unavoidable. Note also that position and force constraints are enforced instead as hard constraints.

The MLD hybrid dynamics (18) has the advantage of making the optimal control problem (20) solvable by mixed-integer quadratic programming (MIQP). At each time step t, given the current reference values $r_x(t)$, $r_u(t)$ and the current state $x(t)$, Problem (20) is solved to get the first optimal input sample $F_m^*(0)$, which is commanded as the reference magnetic force to the inner-loop controller at the lower hierarchical level.

A. Simulation Results

The behavior of the system is simulated for a given reference profile r_{x2} during 0.1 s, corresponding to 500 sampling steps. The input reference is $r_{F_m} = k_1(x_{1e}-r_{x1})+k_2(x_{2e}-r_{x2})$, when the position reference is feasible for both m_1 and m_2, and to $r_{F_m} = k_1(x_{1e}-r_{x1})$, when the reference is feasible for m_1 only. The results are reported in Figure 2, where we assume that the inner-loop controller is fast enough to make the actual magnetic force perfectly track the value commanded by the MPC algorithm.

From the simulation we see that for a given reference value which is feasible for m_1 only, $r_{x1} = r_{x1} = -0.5$ mm, the controller leads the two masses at the closest feasible points ($x_1 = -0.5$ mm, $x_2 = 0$ mm, respectively). The availability of the control force F_m is ensured since hard constraint (10) is enforced. The overshoots in the position tracking are due to the small magnetic force available when the masses are far from the coil. They can be reduced by increasing the prediction horizon, through which the controller can anticipatively detect that the breaking force will become small, hence deciding to start breaking in advance.

The soft-landing constraints (15) are enforced. Figure 3 reports the phase plane $(x_2, \dot{x}_2)$, the state trajectory of mass m_2 (solid line) during the interval $[0.01, 0.053]$ s, and the lower bound (dashed line) of the region in the plane $(x_2, \dot{x}_2)$ where the soft-landing constraint of m_2 with respect to the stop is satisfied. The absolute value of the velocity is reduced as the mass approaches 0 mm, to obtain the soft landing.

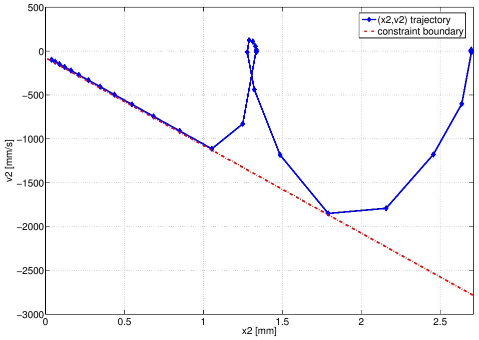

Fig. 3. Phase plane trajectory $(x_2, \dot{x}_2)$ and soft-landing constraint

IV. Switching MPC Implementation

The results of Section III-A (Figure 2) highlight that the system is mainly working in modes M1 or M2, where m_1 moves freely from m_2, and mode M4, where the masses move together. For comparison purposes we design two simpler MPC controllers, one based on M1/M2 and one based on M4, and let an external logic switch them based on current mass positions. Although we expect the performance of the closed-loop system to be degraded because the mode switching is not considered in the prediction model, the associated MIQP problems will be much simpler to solve.

The first controller (MPC1) is valid when m_1 and m_2 are not in contact and it neglects the dynamics of m_2. Thus, it is based on the dynamics of m_1

$$\ddot{x}_1 = \frac{1}{m_1}\Big(- k_1(x_1 - \bar{x}_{n1}) - \beta_1 \dot{x}_1 - F_m \Big). \tag{21}$$

The system dynamics (21), the force constraint (11), and the soft-landing constraint $\dot{x}_1 - \dot{x}_2 \leq k_{sl1}(x_2 - x_1) + h_{sl1} + \rho$ are formulated as an MLD system. Note that when m_1 and m_2 move independently, only the soft-landing of m_1 towards m_2 can be enforced, since only m_1 can be controlled through the magnetic coil.

The cost function minimized by MPC1 for the freely moving mass m_1 is

$$10(x_1(3) - r_{x2})^2 + \sum_{k=0}^{2} 10(x_1(k) - r_{x2})^2 + \\ 0.01(F_m(k) - r_{F_m})^2 + 100\rho^2(k) \tag{22}$$

subject to the reduced MLD model, position constraints (19a), (19c). It is pointless to weight the tracking error of m_2, since in the prediction model either $x_2 = x_1$, when the two masses are in contact, or the input does not affect x_2, when the two masses are not in contact. Note that x_2, $\dot{x}_2$ are treated by MPC1 as measured disturbances. More in detail, in the prediction model of MPC1 we do not account for the stop at 0 mm and we use a constant velocity

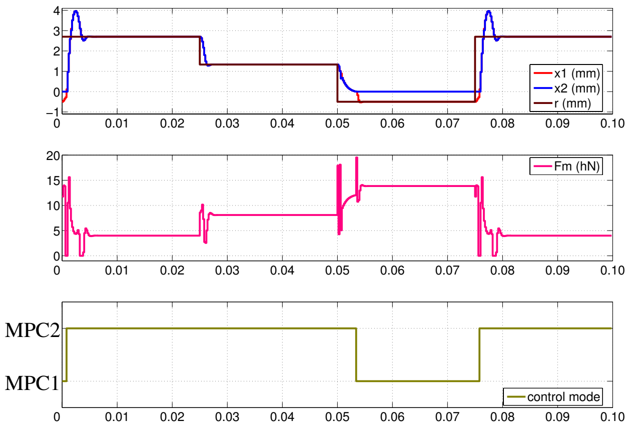

Fig. 4. Switching MPC control law. Time axis is expressed in seconds

approximation of the dynamics of m_2

$$x_2(t+k) = x_2(t) + \dot{x}_2(t)kT_s. \tag{23}$$

The second controller (MPC2) accounts for the situation in which the masses are moving together and the dynamics are described by (4). In this case only the soft-landing of m_2 approaching the stop, defined by constraints (16b) and (17b), is enforced, together with the force saturation constraint (11) and the position constraint (19b). The same values for the weights and horizon defined for controller MPC1 are used.

The switching logic is defined as follows: if $x_1 < x_2$ then use MPC1, otherwise use MPC2.

We simulated the switching MPC control law under the same initial conditions and setpoints of Section III. The input command F_m is applied to the full MLD model (18) which is only used for simulation purposes. The results of the simulation are reported in Figure 4. The results are very close to the ones obtained for the full controller. The cumulative squared tracking error $\mathcal{E} = \sum_{k=0}^{500} (x_2(k) - r_x(k))^2$ is $\mathcal{E}_{\text{full}} = 127.9$ for the full controller of Section III and is $\mathcal{E}_{\text{switch}} = 147.1$ for the switching controller. The degradation of the performance amount to $\simeq 15\%$, and one may note that a part of the cumulative squared tracking error is due to the one-step delay of the controller in reacting to the reference variations. By removing this additional bias of the tracking error, which amounts to 25.7 for the simulation considered here and which does not depend on the choice of the controller, the performance degradation caused by the use of the switched controller is $\simeq 19\%$, still close. In terms of input action the performance degradation of the switched controller is more evident. The squared cumulative input action $\mathcal{E}^u = \sum_{k=0}^{500} (F_m(k) - r_{F_m}(k))^2$ is $\mathcal{E}^u_{\text{full}} = 1.06 \cdot 10^3$ for the full controller of Section III and is $\mathcal{E}^u_{\text{switch}} = 1.58 \cdot 10^3$ for the switching controller. On the other hand, the switching MPC controller problems are much simpler to solve. The average solution time for Problem (20) on an Intel Pentium-M 2 GHz, equipped with 1 GB RAM and running Matlab 7 and Cplex 9.1, is 0.064s, while the average solution time for MPC1 and MPC2 is 0.007s, i.e., 89% less. Moreover, while the full MPC, due to the large number of discrete variables, may be too complex for implementation as an explicit controller [5], the switching MPC controller can be implemented as an explicit MPC controller. The two simplified prediction models are nothing else than mass-spring-damper systems, with different parameters of mass, spring stiffness and damping coefficient. The results of [7] have shown that the explicit implementation of the MPC controller that enforces state-dependent saturation, soft-landing and position constraints on a mass-spring-damper system is feasible within the specified sampling time.

V. Conclusions

In this paper we have considered a multibody automotive mechatronic actuator in which the mechanical subsystem is composed by two colliding masses. By applying hybrid model predictive control concepts, the mass which is not directly controlled by the magnetic force is able to track a desired reference position signal. Two different MPC design approaches have been investigated based on a single (more complex, but higher fidelity) hybrid model with two masses and four operating modes, and based on a switched MPC design in which two simpler MPC controllers are triggered by an external logic unit.

References

[1] D. Hrovat, J. Asgari, and M. Fodor, "Automotive mechatronic systems," in *Mechatronic Systems, Techniques and Applications: Volume 2–Transportation and Vehicle Systems*, C. Leondes, Ed. Gordon and Breach Science Publishers, 2000, pp. 1–98.

[2] L. Guzzella and A. Sciarretta, *Vehicle Propulsion Systems: Introduction to Modeling and Optimization*. Springer Verlag, 2005.

[3] J. Maciejowski, *Predictive control with constraints*. Englewood Cliffs, NJ: Prentice Hall., 2002.

[4] A. Bemporad, M. Morari, V. Dua, and E. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, 2002.

[5] F. Borrelli, M. Baotić, A. Bemporad, and M. Morari, "Dynamic programming for constrained optimal control of discrete-time linear hybrid systems," *Automatica*, vol. 41, no. 10, pp. 1709–1721, Oct. 2005.

[6] S. Di Cairano, A. Bemporad, I. Kolmanovsky, and H. Hrovat, "Model predictive control of nonlinear mechatronic systems: An application to a magnetically actuated mass spring damper," in *2nd IFAC–Analysis and Design of Hybrid Systems*, 2006.

[7] ——, "Model predictive control of magnetic automotive actuators," in *Proc. American Contr. Conf.*, New York, NY, 2006, pp. 5082 – 5087.

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

[9] P. Antsaklis, "A brief introduction to the theory and applications of hybrid systems," *Proc. IEEE, Special Issue on Hybrid Systems: Theory and Applications*, vol. 88, no. 7, pp. 879–886, July 2000.

[10] B. Brogliato, *Non-smooth Impact Mechanics*, ser. Lecture Notes in Control and Information Sciences. Springer-Verlag, 1996.

[11] C. Glocker, *Set-Valued Force Laws, Dynamics of Non-Smooth Systems*, ser. Lecture Notes in Applied Mechanics. Springer-Verlag, 2001.

[12] V. Acary and F. Pérignon, *SICONOS Software*, Sept. 2006, http://siconos.gforge.inria.fr/.

[13] F. Torrisi and A. Bemporad, "HYSDEL — A tool for generating computational hybrid models," *IEEE Trans. Contr. Systems Technology*, vol. 12, no. 2, pp. 235–249, Mar. 2004.

[14] A. Bemporad, *Hybrid Toolbox – User's Guide*, Dec. 2003, http://www.dii.unisi.it/hybrid/toolbox.

[15] R. Raman and I. Grossmann, "Relation between MILP modeling and logical inference for chemical process synthesis," *Computers & Chemical Engineering*, vol. 15, no. 2, pp. 73–84, 1991.