Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Mixed-Integer Model Predictive Control of Hybrid Impulsive Linear Systems

N. Bajcinca * **M. V. A. Pedrosa** * **V. Yfantis** *

* *Department of Mechanical and Process Engineering*
Technische Universität Kaiserslautern
67663 Kaiserslautern, Germany
(e-mail: naim.bajcinca@mv.uni-kl.de).

Abstract: We devise a model predictive control algorithm for impulsive linear systems with autonomous flow dynamics and controlled jumps. Thereby the moments of jumps are not fixed, but rather considered as decision variables. To this end, the complete system dynamics is formulated as a mixed-logical dynamical system after an appropriate discretization step. The resulting optimization problem contains both discrete and continuous decision variables, giving rise to a mixed-integer programming problem. The objective of the optimization is to steer the states into a target set. The stability is addressed through an appropriate cost function together with invariance conditions, as well as by introducing terminal constraints which are only enforced within a certain distance to the target set, thus, providing a trade-off between guaranteed convergence to the target set and computational complexity.

Keywords: Impulsive systems, hybrid systems, model predictive control, mixed-logic dynamics, mixed-integer quadratic programming

1. INTRODUCTION

Impulsive systems are a class of hybrid systems, which present continuous-time states (*flow*) that are exposed to discrete changeovers in time (*jumps*). The literature on model predictive control (MPC) formulations for impulsive systems is still scarce, among which we can highlight the measure-driven framework in Pereira et al. (2015). In particular, publications of MPC for impulsive systems were formalized mainly for impulsive linear systems (ILS), in which Sopasakis et al. (2015) stands out as stated in the survey of Sanfelice (2019). In this context, most works related to ILS have the presence of periodic state jumps in common, meaning that the period between two successive jumps is constant and fixed a priori. However, ILS with non-fixed jump times are important due to their ability to model i. a. cyber-physical systems, as for instance multi-agent communication systems (Duz et al., 2018). In this work, we propose an MPC that also schedules the optimal jumps at arbitrary discrete times, generating non-periodic jumps. To this end, the considered impulsive system is reformulated as a mixed-logical dynamic (MLD) system, which was proposed by Bemporad and Morari (1999), to deal with the nonlinear constraints arising from the relation of the two types of decision variables, the input in the jump and the impulse time. Therefore, the MPC solves a mixed-integer quadratic programming (MIQP) problem in each iteration. Prior to the reformulation, due to a considered minimum dwell time constraint, the system is augmented by a new clock variable.

In Sopasakis et al. (2015), the stabilization of the states' trajectories into a reference set is formulated by an impulsive controlled invariant (ICI) set. However, Rivadeneira et al. (2015) point out that the calculation of this ICI set could be difficult to implement due to the high computational effort as the prediction horizon should be chosen large enough. The authors propose a reference set based on equilibrium points that leads the states to describe an orbit.

In this paper, as the moments of jumps are arbitrary, the construction of an ICI set is strenuous. Therefore, the invariance related to the reference set is considered through conditions over the reference set and the MPC objective function.

This paper is organized as follows. In Section 2, the ILS is formulated and the invariance approach used for the reference set is given. Section 3 proposes a reformulation of the ILS in terms of an MLD system, presenting the MPC as an MIQP problem to minimize the distance of the trajectories to the reference set. A novel activation condition for a terminal constraint is also presented. In Section 4, an application in pharmacokinetics is shown and compared to results from the literature.

2. PRELIMINARIES

In this paper, we consider hybrid dynamical systems in the form of the following linear time-invariant (LTI) impulsive first-order differential equation,

$$\begin{cases} \dot{x}(t) & = Ax(t), & \forall t \neq \tau_i, \\ x(\tau_i^+) & = Ex(\tau_i) + Fu(\tau_i), & \forall i \in \mathbb{N}, \\ \tau_{i+1} - \tau_i & \geq C, & \forall i \in \mathbb{N}, \end{cases} \tag{1}$$

where the first differential equation describes the autonomous flow of the system, while the second equation represents the jumps. The variable $t \in \mathbb{R}_+$ denotes time, $x \in \mathcal{X} \subseteq \mathbb{R}^{n_x}$ is the state vector, $u \in \mathcal{U} \subseteq \mathbb{R}^{n_u}$ is the impulsive input vector and the matrices $A, E \in \mathbb{R}^{n_x \times n_x}$ and $F \in \mathbb{R}^{n_x \times n_u}$ are time-invariant. At time $\tau_i \in \mathbb{R}_+$, the ith jump occurs. Furthermore, a time threshold $C \in \mathbb{R}_+$ has to elapse between two consecutive jumps. The sets $\mathcal{X}$ and $\mathcal{U}$ are defined as convex polyhedral sets. The initial state at time $t = t_0$, $x(t_0) = x_0$, is given. We define $x(\tau_i^+) := \lim_{t \searrow \tau_i} x(t)$.

Differently from the benchmark literature, where the jump times are periodic or at least given (see Sopasakis et al. (2015); Rivadeneira et al. (2015); Altın et al. (2018); Pereira et al. (2015)), our configuration is more flexible, as the set of time instants

$$\mathcal{T} := \{\tau_i \mid i \in \mathbb{N}\} \tag{2}$$

at which jumps occur constitutes a degree of freedom in the formulation of the MPC, presented in the next section.

As the system is autonomous in the flow and the moments of jumps are constrained by the time threshold, it might not be possible to stabilize the system around a fixed set-point, apart from the origin. Therefore, the states of the system are desired to remain inside a non-empty convex polyhedral reference set $\mathcal{R} \subset \mathcal{X}$ which does not necessarily contain the origin. Then, in order to keep the trajectories of (1) inside the reference set $\mathcal{R}$, we compute a controlled invariant subspace $\mathcal{X}^{\mathrm{f}} \subseteq \mathcal{R} \subset \mathcal{X}$ through a geometric approach of polyhedral sets inclusion (Dórea and Hennet, 1999), where

$$\mathcal{X}^{\mathrm{f}} = \{x \in \mathcal{X} \mid \underline{x}_j^{\mathrm{f}} \leq x_j \leq \overline{x}_j^{\mathrm{f}},\ j = 1, \ldots, n_x\}$$

is the target set and j-subscription denotes the jth element of the underlying vector.

For dynamical systems, a set is said to be positively invariant with respect to a dynamical system if the trajectory, once inside the invariant set, never leaves it. When a control input is present, a set is controlled invariant if a control action keeps the trajectory inside the set for any initial condition belonging to the set. When we have this condition for the ILS, the set is said to be impulsive controlled invariant.

For the system (1), we denominate a nonempty set $\mathcal{V} \subseteq \mathcal{X}$ as (A)-invariant if it is positively invariant with respect to the flow dynamics, i. e. if

$$x(\tau_i) \in \mathcal{V} \implies e^{A(t-\tau_i)} x(\tau_i) \in \mathcal{V},\ \forall t \in (\tau_i, \tau_{i+1}],\ \forall i \in \mathbb{N}$$

and as (E, F)-invariant if it is controlled invariant with respect to the jump dynamics, that is if for any

$$x(\tau_i) \in \mathcal{V},\ \exists u(\tau_i) \in \mathcal{U} \implies x(\tau_i^+) \in \mathcal{V},\ \forall i \in \mathbb{N}.$$

A sufficient geometric condition for a set to be the impulsive controlled invariant, i. e. invariant for the system (1), can be established by interleaving results for continuous-time and discrete-time LTI systems (Lawrence, 2014), in terms of the following lemma:

Lemma 1. A subspace $\mathcal{V} \subseteq \mathcal{X}$ is impulsive controlled invariant for the system (1) if it is both (A)-invariant and (E, F)-invariant.

In this paper, we do not specify any upper bound for the time between consecutive jumps is unbounded, rendering the construction of an invariant set that does not necessarily contain the origin, regarding the flow, challenging. However, once the moments of jump are controlled, the MPC can set the (A)-invariance property by choosing the impulse times such that the trajectory does not leave $\mathcal{X}^{\mathrm{f}}$. However, this set should be controlled invariant for the jumps. Then, it is defined as

$$\mathcal{X}^{\mathrm{f}} := \text{supremal (E,F)-invariant set cointained in } \mathcal{R} \tag{3}$$

and calculated through the classical algorithm given by Dórea and Hennet (1999), configured to include a rectangular cone into a polyhedral set. For this setting, we will consider that a suitable value of the clock constraint C is given together with a chosen cost function that coerces the trajectory to respect the (A)-invariance. These conditions will be outlined in Section 3.

3. PROPOSED MIXED-INTEGER MPC

In the following, the reformulation of the ILS (1) into a mixed-logical dynamic system is presented. This system representation forms the basis for the employed MPC algorithm, in which an MIQP problem is solved in each iteration. The system states are augmented by a clock variable in order to directly incorporate the time threshold between jumps in the MPC problem. Special consideration is given to the convergence of the system to the invariant target set, directly addressing the stability of the proposed approach.

3.1 Problem setting

In this paper, the moments of jumps are free decision variables in the proposed system class (1), albeit constrained by the time threshold between consecutive jumps. In order to adequately account for this additional degrees of freedom and constraints, they should be incorporated in the optimization problem of the MPC algorithm. To this end, system (1) is augmented by an additional clock variable $c(t)$ as follows:

$$\begin{cases} \dot{x}(t) &= Ax(t), & \forall t \neq \tau_i, \\ \dot{c}(t) &= 1, & \forall t \neq \tau_i, \\ x(\tau_i^+) &= Ex(\tau_i) + Fu(\tau_i), & \forall i \in \mathbb{N}, \\ c(\tau_i^+) &= 0, & \forall i \in \mathbb{N}, \\ \tau_{i+1} - \tau_i &\geq C, & \forall i \in \mathbb{N}. \end{cases} \tag{4}$$

The new state $c(t) \in \mathbb{R}_+$ represents a clock that increases with a constant rate during the flow and resets to 0 after each jump. By defining an augmented state vector $\hat{x}(t) := [x^T(t), c(t)]^T$, the system can be rewritten as

$$\begin{cases} \dot{\hat{x}}(t) &= \hat{A}\hat{x}(t) + \xi, & \forall t \neq \tau_i, \\ \hat{x}(\tau_i^+) &= \tilde{E}\hat{x}(\tau_i) + \tilde{F}u(\tau_i), & \forall i \in \mathbb{N}, \\ \tau_{i+1} - \tau_i &\geq C, & \forall i \in \mathbb{N}, \end{cases} \tag{5}$$

with

$$\hat{A} := \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{(n_x+1)\times(n_x+1)},\ \xi := \begin{bmatrix} 0 \\ 1 \end{bmatrix} \in \mathbb{R}^{n_x+1},$$

$$\tilde{E} := \begin{bmatrix} E & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{(n_x+1)\times(n_x+1)},\ \tilde{F} := \begin{bmatrix} F \\ 0 \end{bmatrix} \in \mathbb{R}^{(n_x+1)\times n_u}.$$

As the time between two consecutive jumps is not upper bounded, also no upper bound is defined for the last state in $\hat{x}$, i. e. the clock variable. MPC algorithms usually rely on a discrete-time representation of the system. Time is discretized with a fixed sample period $\Delta \in \mathbb{R}_+ \backslash \{0\}$ and $\{t_{k \geq 0}\}$ denotes the sequence of synchronous time points, such that $t_k = t_0 + k\Delta$, where t_0 stands for the initial time. In the following, $x(t_k)$ and $u(t_k)$ will be abbreviated by x_k and u_k respectively. Using the sample time, the continuous dynamics of the system given by $\dot{\hat{x}}(t) = \hat{A}\hat{x}(t) + \xi$ can be discretized by integrating the first order differential equation between two consecutive time points as

$$\hat{x}_{k+1} = \hat{\Phi}\hat{x}_k + \hat{\xi}, \tag{6}$$

with $\hat{\Phi} := \begin{bmatrix} \Phi & 0 \\ 0 & 1 \end{bmatrix}$, $\Phi := e^{A\Delta}$ and $\hat{\xi} := \begin{bmatrix} 0 \\ \Delta \end{bmatrix}$. The discrete dynamics of the hybrid system (5), i. e. the jumps, must also be formulated as a linear difference equation, similar to (6). As time is discretized for the application of MPC, control actions, and therefore jumps, as τ is considered to be a decision variable, can only occur at the discrete time points, i. e. $\mathcal{T} \subseteq \{t_{k \geq 0}\}$.

Assume that a jump takes place at time t_k, i.e.

$$\hat{x}(t_k^+) = \tilde{E}\hat{x}_k + \tilde{F}u_k. \tag{7}$$

Now, note that $\hat{x}(t_k^+) \neq \hat{x}(t_{k+1})$, cf. Fig. 1. Between the jump at time t_k and the next discrete time point $t_{k+1} = t_k + \Delta$, an additional flow takes place. The jump can therefore be written as a difference equation in the form:

$$\begin{aligned} \hat{x}_{k+1} &= \hat{\Phi}(\tilde{E}\hat{x}_k + \tilde{F}u_k) + \hat{\xi} \\ &= \hat{E}\hat{x}_k + \hat{F}u_k + \hat{\xi}, \end{aligned} \tag{8}$$

with $\hat{E} := \hat{\Phi}\tilde{E}$ and $\hat{F} := \hat{\Phi}\tilde{F}$.
The discretization according to equations (6) and (8) is illustrated in Fig. 1. The system now consists of two

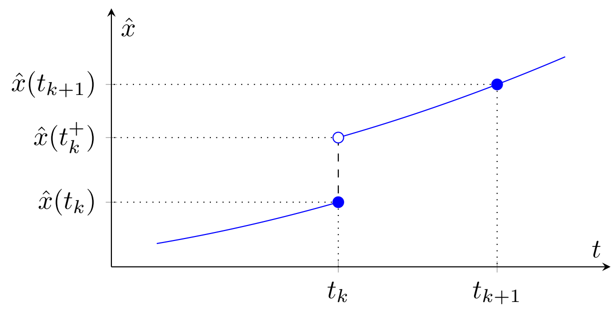

Fig. 1. Illustration of the discretization of the jump dynamics.

difference equations, one for the flow (6) and one for the jumps (8). In order to decide which part of the dynamics is valid at any point in time, the binary variable $\delta_k \in \{0, 1\}$ is introduced, which is equal to one if a jump occurs at time t_k. This binary decision variable is considered as a degree of freedom in the optimization in order to schedule the moments of jump. Using the binary variable, the two difference equations can be combined into a single one,

$$\hat{x}_{k+1} = (1 - \delta_k)(\hat{\Phi}\hat{x}_k + \hat{\xi}) + \delta_k(\hat{E}\hat{x}_k + \hat{F}u_k + \hat{\xi}). \tag{9}$$

Rearranging equation (9) yields

$$\hat{x}_{k+1} = \hat{\Phi}\hat{x}_k + (\hat{E} - \hat{\Phi})\delta_k\hat{x}_k + \hat{F}\delta_k u_k + \hat{\xi}. \tag{10}$$

However, the derived difference equation now contains two bilinear terms, which are computationally unfavorable in the optimization, as they would result in nonlinearities in the constraints. In order to avoid these terms in the optimization, they can be replaced by a set of linear constraints, as proposed by Bemporad and Morari (1999), resulting in an MLD system. In this approach, the bilinear terms are replaced by a set of linear constraints. Therefore, we define auxiliary decision variables $z_k := \delta_k\hat{x}_k$ and $w_k := \delta_k u_k$ and model them through linear (componentwise) inequalities,

$$z_{j,k} \leq \delta_k \overline{x}_j, \tag{11a}$$
$$z_{j,k} \geq \delta_k \underline{x}_j, \tag{11b}$$
$$z_{j,k} \leq x_{j,k} - (1 - \delta_k)\underline{x}_j, \tag{11c}$$
$$z_{j,k} \geq x_{j,k} - (1 - \delta_k)\overline{x}_j \tag{11d}$$

and

$$w_{j,k} \leq \delta_k \overline{u}_j, \tag{12a}$$
$$w_{j,k} \geq \delta_k \underline{u}_j, \tag{12b}$$
$$w_{j,k} \leq u_{j,k} - (1 - \delta_k)\underline{u}_j, \tag{12c}$$
$$w_{j,k} \geq u_{j,k} - (1 - \delta_k)\overline{u}_j, \tag{12d}$$

where $x_{j,k}$, $z_{j,k}$ and $w_{j,k}$ denotes the jth element of its corresponding vector at time t_k and $\overline{x}$, $\underline{x}$, $\overline{u}$ and $\underline{u}$ are the vectors of upper and lower bounds of the states and inputs respectively, which can be derived from the polyhedral sets $\mathcal{X}$ and $\mathcal{U}$ by solving a simple optimization problem a priori. The first two constraints in (11) and (12) force the auxiliary variables to zero if no jump occurs, i. e. $\delta_k = 0$. Otherwise, they are trivially satisfied. On the other hand, the last two constraints force the auxiliary variables to take the values of the states and inputs accordingly if a jump occurs. If no jump takes place, they are trivially satisfied. The final linear difference equation used in the MPC optimization, including the auxiliary variables, reads as

$$\hat{x}_{k+1} = \hat{\Phi}\hat{x}_k + (\hat{E} - \hat{\Phi})z_k + \hat{F}w_k + \hat{\xi}. \tag{13}$$

Equation (13) combined with constraints (11) and (12) model the continuous and discrete dynamics of system (1). The remaining constraint on the moments of jumps can be easily formulated by the introduced clock variable c_k (in discrete time) and the binary variable δ_k,

$$c_k \geq \delta_k C. \tag{14}$$

The binary variable δ_k can only be set equal to one, indicating the occurrence of a jump, if the value of the clock variable c_k exceeds the time threshold C between consecutive jumps.
Fig. 1, additionally, illustrates a drawback of the discretization of the jump dynamics according to eq. (8). If a jump occurs at time t_k, the difference equation is used to compute the value of the states at time t_{k+1}, i. e. $\hat{x}(t_{k+1})$. However, the value of the states after the jump, $x(t_k^+)$, is not computed in the prediction. Therefore, undesirable effects, e. g. constraint violations, can occur after a jump, as long as the states reach a feasible point until the next sampling point. In order to avoid this situation and further consider the state values after a jump in the objective of the optimization, a new variable x_k^+ is introduced, corresponding to the value of the states if a jump occurs at time t_k. This variable can be computed by

$$x_k^+ = \delta_k(Ex_k + Fu_k) = Ez_k + Fw_k. \tag{15}$$

Because of the definitions for the variables z_k and w_k, the variable x_k^+ takes the value zero if no jump occurs at time t_k. In order to avoid jumps into infeasible regions of the state space, the value of the states after a jump have to lie within the feasible subset of states,

$$x_k^+ \in \mathcal{X} \cup \{0\}. \tag{16}$$

3.2 Cost function

The goal of the optimization is to steer the states x of the system into a given target set $\mathcal{X}^{\mathrm{f}}$. A natural objective for this purpose is the minimization of the weighted squared distance of the predicted states to the target set,

$$\mathrm{dist}_Q(x_k, \mathcal{X}^{\mathrm{f}}) = \min_{\chi \in \mathcal{X}^{\mathrm{f}}} \|x_k - \chi\|_Q^2 , \tag{17}$$

where $\|\cdot\|_Q$ represents the weighted euclidean vector norm and $Q \in \mathbb{R}^{n_x \times n_x}$ denotes the positive definite weight matrix. In order to include this distance metric in the optimization of the MPC, further auxiliary variables $\hat{x}^{\mathrm{f}}$ are defined and constrained to lie within the target set, i. e.

$$x^{\mathrm{f}} \in \mathcal{X}^{\mathrm{f}}. \tag{18}$$

So, the weighted sum of the distance of the states to the target set over the prediction horizon N can then take the form

$$J_1(x_k, \mathcal{X}^{\mathrm{f}}) = \sum_{l=0}^{N-1} \left\|x_{k+l} - x_{k+l}^{\mathrm{f}}\right\|_Q^2 . \tag{19}$$

In order to penalize the states leaving the target set, a penalty term including the variables x_k^+ is considered,

$$J_2(x_k^+, \mathcal{X}^{\mathrm{f}}) = \sum_{l=0}^{N-1} \left\|x_{k+l}^+ - x_{k+l}^{\mathrm{f},+}\right\|_{Q^+}^2 , \tag{20}$$

where the auxiliary variables $x_k^{\mathrm{f},+}$ is constrained by

$$\delta_k \underline{x}^{\mathrm{f}} \le x_k^{\mathrm{f},+} \le \delta_k \overline{x}^{\mathrm{f}}. \tag{21}$$

All terms in (20) vanish for the time points at which no jump occurs. It, therefore, only penalizes jumps that move the states away from the target set. Additionally to the penalization of the states, the control inputs u_k are also penalized in the objective function according to

$$J_3(u_k, \mathcal{U}^{\mathrm{f}}) = \sum_{l=0}^{N-1} \left\|u_{k+l} - u_{k+l}^{\mathrm{f}}\right\|_R^2 , \tag{22}$$

where u^{f} denotes auxiliary variables such that

$$u^{\mathrm{f}} \in \mathcal{U}^{\mathrm{f}} := \{u \in \mathcal{U} \mid Ex + Fu \in \mathcal{X}^{\mathrm{f}},\ \forall x \in \mathcal{X}^{\mathrm{f}}\}. \tag{23}$$

The input target set can be approximated by computing lower and upper bounds on the inputs through a minimization and maximization of $u \in \mathcal{U}$ subject to $x \in \mathcal{X}^{\mathrm{f}}$ and $Ex + Fu \in \mathcal{X}^{\mathrm{f}}$, respectively. As the state target set is a polyhedron and the jump dynamics are linear, the computation requires the solution of linear programs.

As the flow is an autonomous system, it is important to state the following sufficient clock condition C considered in this paper:

Remark 2. We can assume C as suitable to allow a jump to happen before the flow leaves the target set, i.e.,

$$\exists u \in \mathcal{U}^{\mathrm{f}} : e^{At}(Ex+Fu) \in \mathcal{X}^{\mathrm{f}},\ \forall x \in \mathcal{X}^{\mathrm{f}},\ \forall t \le \left\lceil \tfrac{C}{\Delta} \right\rceil \Delta. \tag{24}$$

However, if the states might leave the target set, without violating the constraints, the algorithm would steer them back at the next possible moment of jump. Note as well, that the MPC sampling time Δ has to be small enough to capture all relevant dynamic effects during the flow.

3.3 Convergence to the target set

A common approach to guarantee the convergence of the states to the target set upon convergence of the MPC optimization problem is to apply terminal constraints, i. e. $x_{k+N-1} \in \mathcal{X}^{\mathrm{f}}$. This typically requires a long prediction horizon N in order to give the system enough time to reach the set, since no feasible solution might be found otherwise. An increased prediction horizon however increases the size of the underlying optimization problem. As the proposed MPC approach involves the solution of an MIQP problem in each iteration, whose complexity can grow exponentially with its size in the worst case, a long prediction horizon might lead to major computational drawbacks. As a trade-off between computational performance and reachability of the target set, a region close to the target set is defined and the following condition is implemented:

$$|x_{j,k} - x_{j,k}^{\mathrm{f}}| \le \varepsilon_j,\ \forall j \implies x_{k+N-1} \in \mathcal{X}^{\mathrm{f}}, \tag{25}$$

with $\varepsilon_j \ge 0$. Condition (25) states that only if all states x_j lie within a distance ε_j of the target set at time t_k (initial time point of the current iteration), then the terminal constraint, demanding convergence to the target set at the end of the prediction horizon, is applied. The condition (25) can be modeled through linear constraints as follows.

Modeling $|x_{j,k} - x_{j,k}^f|$: The norm expression can be modelled considering

$$|x_{j,k} - x_{j,k}^{\mathrm{f}}| = \alpha_j + \omega_j ,$$

with $\alpha_j, \omega_j \ge 0$, which is equivalent to the transformation

$$x_{j,k} - x_{j,k}^{\mathrm{f}} = \alpha_j - \omega_j , \tag{26a}$$

$$\alpha_j \le M\mu_j , \tag{26b}$$

$$\omega_j \le M(1 - \mu_j), \tag{26c}$$

for all j, where μ_j is a binary variable and $M \ge \|x - x^{\mathrm{f}}\|_\infty$ for $x \in \mathcal{X}, x^{\mathrm{f}} \in \mathcal{X}^{\mathrm{f}}$. The binary variable μ_j has to be equal to one if the term $x_{j,k} - x_{j,k}^{\mathrm{f}}$ is positive, setting the value of ω_j to zero through constraint (26c). Otherwise, μ_j, and consequently α_j, are set to zero.

Conditional statement: The inequality $\alpha_j + \omega_j \le \varepsilon_j$ can be modeled through the following linear constraints (Bemporad and Morari, 1999):

$$\alpha_j + \omega_j - \varepsilon_j \ge \eta - (\varepsilon_j + \eta)\gamma_j , \tag{27a}$$

$$\alpha_j + \omega_j - \varepsilon_j \le M(1 - \gamma_j), \tag{27b}$$

for all j, with $\eta > 0$ and γ_j being a binary variable, indicating if state x_j is close to the target set. Finally, the condition (25), i.e. activating the terminal constraints if the above inequalities are satisfied for all elements j, can be modeled jointly with (26) and (27) through

$$\zeta \le \gamma_j , \tag{28a}$$

$$\sum_{j=1}^{n_x} \gamma_j - \zeta \le n_x - 1, \tag{28b}$$

$$\underline{x}_j^{\mathrm{f}} - x_{j,k+N-1} \le M(1 - \zeta), \tag{28c}$$

$$\overline{x}_j^{\mathrm{f}} - x_{j,k+N-1} \ge -M(1 - \zeta). \tag{28d}$$

for all j, where a binary variable ζ is defined. If any state is not close to their respective target set, constraint (28a) sets ζ to zero and if all states are close to the target set, constraint (28b) sets it to one. If ζ is equal to one, constraints (28c) and (28d) enforce the terminal constraints, otherwise they are relaxed.

Note that we are not considering the augmented states in the constraints (26) and (28), since it is meaningless to have convergence in the clock variable $c(t)$.
Choosing small values of ε allows for shorter prediction horizons in order to assure feasibility and enhanced computational performance. However, more MPC iterations may have to elapse until the terminal constraints are enforced, guaranteeing convergence to the target set, if the values are chosen too small. The choice of ε therefore poses a trade-off between computational performance and convergence to the target set. Despite of not having a theoretical convergence guarantee to the ε-zone, the objective function steers the trajectory towards the target set and, consequently, leads it to reach this region. If the states do not reach a distance ε to $\mathcal{X}^{\mathrm{f}}$, it can be argued that the system can not reach the target set anyway.

Remark 3. If a stability assurance from the initial state is required, instead of setting $\varepsilon_j = M, \ \forall j$, one can substitute (28) to simply $x_{k+N-1} \in \mathcal{X}^{\mathrm{f}}$ with a larger prediction horizon.

3.4 MPC Problem

Consider a given x_0. Then, the MPC will solve, for each iteration time k, the optimization problem below:

$$\begin{aligned} &\underset{u_{k+l},\delta_{k+l}}{\text{minimize}} \ J(x_k) \\ &\text{subject to } x \in \mathcal{X}, \\ &\qquad\qquad u \in \mathcal{U}, \\ &\qquad\qquad (11)\text{–}(16), (18), (21), (23), (26)\text{–}(28), \end{aligned}$$

$\forall l = 0, 1, \dots, N-1$, where the objective function is

$$J(x_k) = J_1(x_k, \mathcal{X}^{\mathrm{f}}) + J_2(x_k^+, \mathcal{X}^{\mathrm{f}}) + J_3(u_k, \mathcal{U}^{\mathrm{f}}), \qquad (30)$$

from equations (19), (20) and (22). Note that, as we have the condition (3) and the assumption (24),

$$x_k \in \mathcal{X}^{\mathrm{f}} \implies J^*(x_k) = 0 \iff \begin{cases} x_{k+l} \in \mathcal{X}^{\mathrm{f}}, \\ x_{k+l}^+ \in \mathcal{X}^{\mathrm{f}} \cup \{0\}, \\ u_{k+l} \in \mathcal{U}^{\mathrm{f}}, \end{cases}$$

for all $l = 0, 1, \dots, N-1$. The optimization problem is an MIQP, for which a global minimum can be found through the branch-&-cut algorithm. Thus, this global minimum will be found by the optimizer, assuring the invariance of $\mathcal{X}^{\mathrm{f}}$.

4. EXAMPLE

The approach described above can be validated by the following benchmark example, used in Sopasakis et al. (2015) and Rivadeneira et al. (2015). A physiological pharmacokinetic model describing the distribution of lithium ions upon oral administration, was introduced in Ehrlich et al. (1980). In this model, the state vector $x(t) := [x_P(t)\ x_R(t)\ x_M(t)]^T$ represents the concentrations in plasma (P), in red blood cells (R) and in muscle cells (M). This example was used in both Sopasakis et al. (2015) and Rivadeneira et al. (2015) with fixed jump times of 3 h. The impulsive system is described by the following matrices

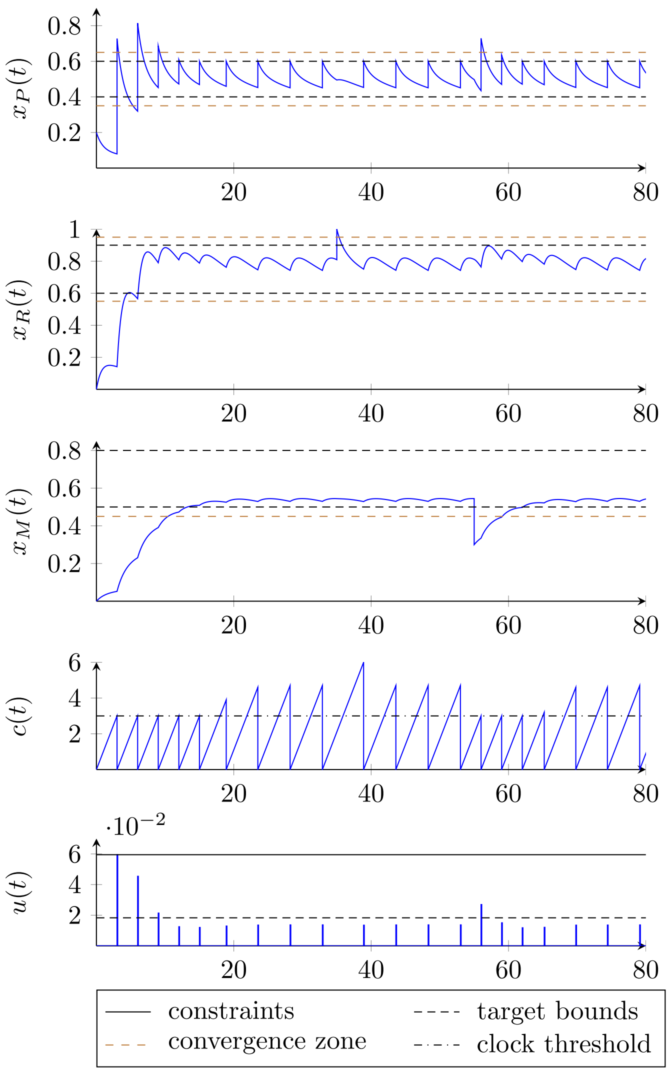

Fig. 2. Augmented states, input and jumps evolution.

$$A = \begin{bmatrix} -0.6137 & 0.1835 & 0.2406 \\ 1.2644 & -0.8 & 0 \\ 0.2054 & 0 & -0.19 \end{bmatrix}, \quad E = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$F = \begin{bmatrix} 10.9 \\ 0 \\ 0 \end{bmatrix}, \qquad C = 3.$$

In this particular case, the set $\mathcal{X}$ is a rectangular cuboid with lower and upper bounds

$$\underline{x} = [0\ 0\ 0]^T, \quad \overline{x} = [2\ 1.2\ 1.2]^T,$$
$$\underline{u} = 0, \quad \overline{u} = 0.0595.$$

The computed target set is equal to the full reference set, i. e. $\mathcal{X}^{\mathrm{f}} = \mathcal{R}$. Thereby, the target set $\mathcal{X}^{\mathrm{f}}$ and $\mathcal{U}^{\mathrm{f}}$ are defined by the following boundaries

$$\underline{x}^{\mathrm{f}} = [0.4\ 0.6\ 0.5]^T, \quad \overline{x}^{\mathrm{f}} = [0.6\ 0.9\ 0.8]^T,$$
$$\underline{u}^{\mathrm{f}} = 0, \quad \overline{u}^{\mathrm{f}} = 0.01835.$$

The MPC parameters were chosen as follows:

$$Q = Q^+ = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad R = 1, \quad \varepsilon = \begin{bmatrix} 0.05 \\ 0.05 \\ 0.05 \end{bmatrix},$$

$$N = 35, \qquad \eta = 10^{-4}.$$

The system was discretized with a sampling time of $\Delta = 0.1$, and its initial value was $x_0 = [0.2\ 0\ 0]^T$.

Figure 2 shows the evolution of the augmented states, input and jumps, i. e. $\hat{x}(t)$, $u(t)$ and $\delta(t)$, respectively. At the beginning until $t = 15$ h, the jumps occur more frequently prior to all the states reaching the target set. For this period of time, when the clock threshold is reached, a jump occurs and the clock is reset to 0. Once the states reach the target set, they remain inside of it respecting the target bounds, while the optimized times of jumps occur only every 4.7 h after a steady state is attained. The results show that, compared to the ones with periodic jumps of 3 h, the optimized jump times lead to less frequent impulses. Without concerning about practical/physical viability, but rather exploring the flexibility of the MPC, one perturbation was added in $x_R(35)$ and a second one in $x_M(55)$. These states are interesting as the input is present only for $x_P(t)$. In both cases, the system could steer the states back to the target set, while appropriately scheduling the moments of jump.

The code was implemented in Julia version 1.1.1 (Bezanson et al., 2017) and solved on a Windows 10 machine with 3.40 GHz Intel Core i5 CPU, 32 GB RAM. On 800 iterations, the average computation time was 0.1079 s (st.dev.: 0.3046 s, max.: 2.6576 s).

5. CONCLUSION AND OUTLOOK

An MPC problem in the form of an MIQP for a class of MLD formulated discretized impulsive linear systems with non-fixed moments of jumps has been discussed. A suitable cost function has been chosen together with a terminal constraint to deal with the stabilization through a reference set, while a minimum dwell time condition is additionally considered. The results show that the MPC has been capable to confine the set into the target set, respecting its boundaries, with less jumps as compared to a numerical example taken from the literature. Also, disturbance rejection of the compiled algorithm has been numerically demonstrated.

The synthesis of an impulsive controlled invariant target set for the adopted class of systems, possibly with the inclusion of inputs in the flow or a maximal dwell time condition, is a challenge for future works. A robustness formulation can also be addressed. Lastly, the approach used in this paper can be extended to general nonlinear and/or unstable systems as well, where e. g. the real-time solution of mixed-integer nonlinear programming (MINLP) problems and a potential finite escape times between consecutive jumps have to be specifically addressed.

REFERENCES

Altın, B., Ojaghi, P., and Sanfelice, R.G. (2018). A model predictive control framework for hybrid dynamical systems. *IFAC-PapersOnLine*, 51(20), 128 – 133. 6th IFAC Conference on Nonlinear Model Predictive Control NMPC 2018.

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

Bezanson, J., Edelman, A., Karpinski, S., and Shah, V.B. (2017). Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1), 65–98.

Dórea, C.E.T. and Hennet, J.C. (1999). (A, B)-invariant polyhedral sets of linear discrete-time systems. *Journal of Optimization Theory and Applications*, 103(3), 521–542.

Duz, A., Phillips, S., Fagiolini, A., Sanfelice, R.G., and Pasqualetti, F. (2018). Stealthy attacks in cloud-connected linear impulsive systems. In *2018 Annual American Control Conference (ACC)*, 146–152.

Ehrlich, B.E., Clausen, C., and Diamond, J.M. (1980). Lithium pharmacokinetics: Single-dose experiments and analysis using a physiological model. *Journal of Pharmacokinetics and Biopharmaceutics*, 8(5), 439–461.

Lawrence, D.A. (2014). Controlled invariant subspaces for linear impulsive systems. In *2014 American Control Conference*, 2336–2341.

Pereira, F., Fontes, F., Aguiar, A.P., and Borges de Sousa, J. (2015). *An Optimization-Based Framework for Impulsive Control Systems*, 277–300.

Rivadeneira, P.S., Ferramosca, A., and Gonzlez, A.H. (2015). MPC with state window target control in linear impulsive systems. *IFAC-PapersOnLine*, 48(23), 507 – 512. 5th IFAC Conference on Nonlinear Model Predictive Control NMPC 2015.

Sanfelice, R.G. (2019). *Hybrid Model Predictive Control*, 199–220. Birkhäuser, Cham.

Sopasakis, P., Patrinos, P., Sarimveis, H., and Bemporad, A. (2015). Model predictive control for linear impulsive systems. *IEEE Transactions on Automatic Control*, 60(8), 2277–2282.