A Mixed Integer Approach for the Solution of Hybrid Model Predictive Control Problems

Iman Nodozi and Ricardo G. Sanfelice

***Abstract*—This paper presents an algorithm for solving the optimization problem associated with hybrid model predictive control for a class of discretized hybrid control systems. The proposed approach consists of reformulating the optimal control problem as a mixed integer quadratic problem (MIQP), which can be solved using well-established algorithms in the literature. Specifically, the given discretized hybrid control system is transformed into a mixed logical dynamical (MLD) system that, for the class of discretized hybrid control systems considered, gives rise to an MIQP. The MLD model is obtained through an intermediate step that transforms the discretized hybrid control system into a discrete-time control system. The approach is illustrated in several examples.**

I. Introduction

Model predictive control (MPC) is a powerful feedback control technique as it assures asymptotic stability with optimality and constraint satisfaction [1]. Being an optimization-based technique, MPC tends to be computationally expensive and heavily depends on the performance of the optimization scheme employed. It is well documented that MPC may require substantial amount of time to compute due to the time required for the optimization scheme to terminate [2], [3].

As the optimal control problem (OCP) that is to be solved at each control recomputation event is, in general, a nonlinear programming problem, there are many approaches and algorithms available in the literature. For instance, the OCP associated with MPC can be formulated to predict the state and perform optimization sequentially or simultaneously. The OCP, itself, can be solved using a myriad of methods, such as sequential quadratic programming techniques, penalty methods, Lagrangian-based approaches, interior point methods, among others. The survey [4] outlines these approaches and solution methods. These techniques are applicable to important classes of MPC problems for continuous-time and discrete-time systems. On the other hand, techniques for the solution of MPC problems for hybrid systems are much less developed. When the hybrid system is modeled as a piecewise affine (PWA) system or a mixed logical dynamical (MLD) system, and the cost functions are quadratic, the OCP can be formulated as a mixed-integer quadratic problem (MIQP) [5], [6]. This approach is quite powerful as it enables the use of MIQP solvers available in the literature.

Motivated by the success of formulating the OCP as an MIQP for hybrid systems modeled as PWA or MLD systems, we propose an MIQP approach for the solution to the hybrid MPC problem formulated in [7], [8], [9]. In these articles, hybrid systems are modeled by *hybrid equations*, which are given in terms of constrained differential and difference equations. A general theory of robust asymptotic stability and hybrid control design for such class of systems is available in [10] and [11], respectively, where the versatility and generality of the framework is displayed in several applications. Although key theoretical aspects of hybrid MPC are addressed in [7], [8], [9], the numerical solution of the associated (hybrid) OCP is not investigated therein.

As a first step towards efficient methods for the solution to such problems, we consider the discretized version of hybrid equations considered in [12] (see also [13]) and the hybrid MPC problem therein and propose an efficient method to compute solutions to the associated OCP. To accommodate the binary variables involved, we employ the so-called McCormick Relaxation to reformulate the hybrid MPC problem as an MIQP. For the class of discretized hybrid equations considered—specifically, those with linear flow and jump maps, and flow and jump sets given in terms of inequalities involving a function of the state and input— and following the ideas in [14], we derive a (discrete-time) MLD system model of the hybrid equation and formulate the OCP associated with hybrid MPC as an MIQP. Our approach consists of transforming the discretized hybrid equation into a discrete-time system with binary variables, followed by its reformulation as an MLD system. To mathematically formalize these transformations, we establish equivalences between the trajectories (or solutions) to each system. With such relationships in place, we relate the OCP solution associated with the equivalent MLD system—which can be obtained using MIQP solvers— to the solution to the OCP associated with the discretized hybrid equation. Consequently, our results provide an MIQP solution to hybrid MPC for the class of systems considered. In addition, we present an algorithm that implements our approach and we illustrate it in several examples, namely, the canonical bouncing ball system and a congestion control mechanism used in models of transmission control protocols.

Organization: The outline of this paper is as follows. Sec. II presents the definitions of discretized hybrid control systems, mixed logical dynamical systems, and their solutions. The MPC problem for discretized hybrid dynamical system is formulated in Sec. III. In Sec. IV, we detail the mixed integer formulation of discretized hybrid model predictive control, followed by numerical simulations in Sec. V. Concluding remarks are provided in Sec. VI. Due to space constraints, all proofs are given in [15].

I. Nodozi and R. G. Sanfelice are with the Department of Electrical and Computer Engineering, University of California, Santa Cruz. Email: inodozi, ricardo@ucsc.edu. Research partially supported by NSF Grants no. ECS-1710621, CNS-1544396, and CNS-2039054, by AFOSR Grants no. FA9550-19-1-0053, FA9550-19-1-0169, and FA9550-20-1-0238, by ARO Grant no. W911NF-20-1-0253.

II. Preliminaries

A. Notation

We denote by $\mathbb{R}$ the set of real numbers, $\mathbb{R}_{\geq 0}$ its nonnegative subset, and by $\mathbb{N}$ the set of nonnegative integers. Boolean "or," "and," and "not" are denoted by $\vee$, $\wedge$, and $\sim$, respectively. The standard projection onto $\mathbb{R}^n$ is defined by function $\Pi : \mathbb{R}^n \times \mathbb{R}^p \to \mathbb{R}^n$, such that $\Pi(x, y) = x$. The n-dimensional identity matrix is denoted by I_n.

B. Hybrid Control Systems

In this paper, we consider an affine discretized hybrid control system given by

$$\mathcal{H}_d : \begin{cases} x^+ = f(x,u) := A_f x + B_f u + c_f & (x,u) \in C \\ x^+ = g(x,u) := A_g x + B_g u + c_g & (x,u) \in D, \end{cases} \tag{1}$$

where $(x, u) \in C \cup D =: \mathcal{X} \subset \mathbb{R}^n \times \mathbb{R}^m$ are the state and the input of the system, respectively. The set C is called the flow set, and D is called the jump set. The affine functions $f : C \to \mathbb{R}^n$ and $g : D \to \mathbb{R}^n$ are the flow and jump maps, respectively, and $\mathcal{X}$ is compact.

Definition 1: A set $E \subset \mathbb{N} \times \mathbb{N}$ is called a discrete hybrid time domain if, for each $(K, J) \in E$, there exists a nondecreasing sequence $\{k_j\}_{j=0}^{J+1}$ such that $k_0 = 0$, $k_{j+1} \in \mathbb{N}$ for each $j \in \{1, 2, ..., J\}$, and

$$E \cap (\{0, 1, ..., K\} \times \{0, 1, ..., J\}) = \bigcup_{j=0}^{J} \bigcup_{k=K_j}^{K_{j+1}} (k, j).$$

•

The state and input are presented by discrete hybrid time $(k, j) \in \mathbb{N} \times \mathbb{N}$, where k and j index the evolution during flow and jumps, respectively.

Definition 2 (Solution pair [12]): A pair ($x : \operatorname{dom} x \to \mathbb{R}^n$, $u : \operatorname{dom} u \to \mathbb{R}^m$) is a solution pair to $\mathcal{H}_d$ if the following conditions hold:

(S1) $\operatorname{dom} x = \operatorname{dom} u$ is a discrete hybrid time domain.

(S2) $(x(0, 0), u(0, 0)) \in \mathcal{X}$.

(S3) For each $(k, j) \in \operatorname{dom} x$ such that $(k + 1, j) \in \operatorname{dom} x$,

$$x(k+1, j) = f(x(k, j), u(k, j)) \quad (x(k, j), u(k, j)) \in C.$$

(S4) For each $(k, j) \in \operatorname{dom} x$ such that $(k, j + 1) \in \operatorname{dom} x$,

$$x(k, j+1) = g(x(k, j), u(k, j)) \quad (x(k, j), u(k, j)) \in D.$$

•

Throughout, $\widehat{\mathcal{S}}_{\mathcal{H}_d}(x_0)$ is the set to solution pairs (x, u) of $\mathcal{H}_d$ such that $x(0, 0) = x_0$. The pair $(L, J) \in \operatorname{dom}(x, u)$ is called the terminal time to the solution pair (x, u) if $k \leq K$ and $j \leq J$ for all $(k, j) \in \operatorname{dom}(x, u)$.

C. Mixed Logical Dynamical Systems

A general MLD model is given by [14]

$$\hat{x}^+ = A\hat{x} + B_1\hat{u} + B_2\hat{\delta} + B_3 z + B_4 \tag{2}$$

$$\text{subject to } E_2\hat{\delta} + E_3 z \leq E_1\hat{u} + E_4\hat{x} + E_5, \tag{3}$$

where $\hat{x} \in \mathbb{R}^n$ is the state and $\hat{u} \in \mathbb{R}^m$ is the input of the system. The auxiliary continuous and binary variables are represented by $z \in \mathbb{R}^{n_d}$ and $\hat{\delta} \in \{0, 1\}^{m_d}$, respectively. All of these variables have binary and continuous values. The matrices A, $\{B_i\}_{i=1}^4$, and $\{E_i\}_{i=1}^5$ have appropriate dimensions.

The MLD model in (2)-(3) can be expressed as

$$\mathcal{H}_{MLD} : \begin{cases} \hat{x}^+ = \Phi(z, \hat{\delta}, \hat{x}, \hat{u}) \\ \Psi(z, \hat{\delta}, \hat{x}, \hat{u}) \leq 0, \end{cases} \tag{4}$$

where

$$\begin{aligned} \Phi(z, \hat{\delta}, \hat{x}, \hat{u}) &:= A\hat{x} + B_1\hat{u} + B_2\hat{\delta} + B_3 z + B_4, \\ \Psi(z, \hat{\delta}, \hat{x}, \hat{u}) &:= E_2\hat{\delta} + E_3 z - E_1\hat{u} - E_4\hat{x} - E_5. \end{aligned} \tag{5}$$

A solution to $\mathcal{H}_{MLD}$ is defined as follows.

Definition 3: A function $\mathcal{M} \ni \ell \mapsto \big(z(\ell), \hat{\delta}(\ell), \hat{x}(\ell), \hat{u}(\ell)\big)$ is a solution to $\mathcal{H}_{MLD}$ if it satisfies

$$\begin{aligned} &\hat{x}(\ell + 1) = \Phi(z(\ell), \hat{\delta}(\ell), \hat{x}(\ell), \hat{u}(\ell)), \\ &\Psi(z(\ell), \hat{\delta}(\ell), \hat{x}(\ell), \hat{u}(\ell)) \leq 0 \end{aligned} \tag{6}$$

for all $\ell \in \mathcal{M}$, where $\mathcal{M}$ is of the form $\{0, 1, \ldots, K\}$, with K finite, or equal to $\mathbb{N}$. •

III. Hybrid Model Predictive Control for Discretized Hybrid Control Systems

This section formulates a Model Predictive Control (MPC) problem for discretized hybrid dynamical system given by $\mathcal{H}_d$ in (1). Based on the framework given in [16], we first introduce some details related to the MPC of discretized hybrid systems.

A. Prediction Horizon

A fixed end-time optimal control problem can be appropriately used in continuous/discrete-time MPC that optimal controls are updated periodically, and each computed control input has the same terminal time. But, due to the nature of (discrete) hybrid time domains, using a fixed end-time optimal control problem is restrictive. For (discretized) hybrid dynamical systems, we must keep in mind that the solutions might only flow or only jump, or combination of two, so the prediction horizon that is defined must accommodate solutions having different discrete hybrid time domains. To address these issues, as in [16], we define the prediction horizon $\mathcal{T} \subset \mathbb{N} \times \mathbb{N}$ as

$$\mathcal{T} := \{(k, j) \in \mathbb{N} \times \mathbb{N} : \max\{k, j\} = \tau_p\} \tag{7}$$

where τ_p is a given integer. Thus, for some $\tau_p \in \{1, 2, \ldots\}$, the terminal time (T, J) of every feasible solution pair satisfies $\max\{T, J\} = \tau_p$.

B. Cost Functional

Given a solution pair (x,u) to $\mathcal{H}_d$ with compact domain and terminal time (L,J), let $\{K_j\}_{j=0}^{J+1}$ be a nondecreasing sequence such that $\mathrm{dom}(x,u) = \bigcup_{j=0}^{J} \bigcup_{k=K_j}^{K_{j+1}} (k,j)$, and $K_{J+1} = L$, and $X \subset \Pi(\mathcal{X})$ be the terminal constraint set. If $x(K,J) \in X$, then the cost of the pair (x,u) is given by

$$\begin{aligned}\mathcal{J}(x,u) := &\Big(\sum_{j=0}^{J}\sum_{k=K_j}^{K_{j+1}-1} L_C(x(k,j),u(k,j))\Big)\\ &+\Big(\sum_{j=0}^{J-1} L_D(x(k,j),u(k,j))\Big) + V(x(L,J)).\end{aligned} \tag{8}$$

In (8), L_C is called the flow cost and is defined on the flow set C, L_D is called the jump cost and is defined on the jump set D, and V is called the terminal cost defined on the terminal constraint set X.

C. Hybrid Optimal Control Problem

Given the terminal constraint set X and the prediction horizon $\mathcal{T}$, the minimization of the cost functional $\mathcal{J}$ is performed over solution pairs of $\mathcal{H}_d$ with initial condition x_0.

Problem 1: Given an initial condition $x_0 \in \mathbb{R}^n$,

$$\begin{aligned}\text{minimize}\quad & \mathcal{J}(x,u)\\ \text{subject to}\quad & (x,u) \in \widehat{\mathcal{S}}_{\mathcal{H}_d}(x_0)\\ & x(L,J) \in X\\ & (L,J) \in \mathcal{T},\end{aligned} \tag{9}$$

where the constraints $x(L,J) \in X$ and $(L,J) \in \mathcal{T}$ dictate that solutions pairs have terminal conditions in X and terminal times in $\mathcal{T}$, respectively. •

If a solution pair (x,u) satisfies the constraints in (9) with $x(0,0) = x_0$, then we call it a feasible solution. A feasible solution is called the optimal solution if it minimizes $\mathcal{J}$.

In the next section, we show that the model $\mathcal{H}_d$ and Problem 1 can be reformulated as a mixed integer quadratic problem (MIQP) and solve it with an MIQP solver.

IV. A Mixed Integer Formulation of Discretized Hybrid Model Predictive Control

We formulate a version of Problem 1 that can be solved using mixed integer tools. To this end, we proceed as follows:

Step 1) The discretized hybrid control system $\mathcal{H}_d$ is converted into a discrete-time control system, denoted $\tilde{\mathcal{H}}_d$;
Step 2) The new discrete-time control system $\tilde{\mathcal{H}}_d$ is converted into an MLD system, denoted $\mathcal{H}_{MLD}$;
Step 3) Problem 1 is formulated for $\mathcal{H}_{MLD}$ and solved using mixed integer tools.

The conversion in Step 1 is an intermediate step leading to a model that can be recast as an MLD system. This conversion is technical and is described in Appendix A. For this reformulation to be possible, we impose the following structure on the flow set and the jump set of $\mathcal{H}_d$.

Assumption 1: The flow set C is given as

$$C = C_1 \cup C_2 \tag{10}$$

and the jump set D is given as

$$D = D_1 \cap D_2 \tag{11}$$

where, for each $i \in \{1,2\}$,

$$C_i = \{(x,u) \in \mathcal{X} : h_i(x,u) - \sigma_i \le 0\}, \tag{12}$$

$$D_i = \{(x,u) \in \mathcal{X} : h_i(x,u) + \sigma_i \ge 0\}, \tag{13}$$

$h_i : \mathcal{X} \to \mathbb{R}$ is defined as $h_i(x,u) = h_{i1}^\top x + h_{i2}^\top u$, with h_{i1} and h_{i2} vectors of appropriate dimension and $\sigma_i \ge 0$ is a constant. •

For each $i \in \{1,2\}$, we define set-valued maps $\mathcal{U}_i : C_i \cup D_i \rightrightarrows \{0,1\}$ as follows:

$$\mathcal{U}_i(x,u) := \begin{cases} 1 & \text{if} \quad (x,u) \in C_i \setminus D_i\\ 0 & \text{if} \quad (x,u) \in D_i \setminus C_i\\ \{0,1\} & \text{if} \quad (x,u) \in C_i \cap D_i,\end{cases} \tag{14}$$

We exploit the MLD system structure enabled by Assumption 1 to formulate an MIQP version of Problem 1. For this purpose, we impose the following assumption on the flow cost, jump cost, and terminal cost in the cost functional $\mathcal{J}$ in (8).

Assumption 2: The flow cost L_C, the jump cost L_D, and the terminal cost V are given by

$$\begin{aligned} &L_C(x,u) = x^\top Q_c x + u^\top R_c u,\\ &L_D(x,u) = x^\top Q_d x + u^\top R_d u, \quad V(x) = x^\top P x\end{aligned} \tag{15}$$

for each $(x,u) \in \mathcal{X}$, where $P \succeq 0$, $Q_c \succeq 0$, $R_c \succ 0$, $Q_d \succeq 0$, and $R_d \succ 0$. •

A. Recasting $\mathcal{H}_d$ as an MLD system

Using McCormick Relaxation (also known as binary decomposition) from [17] and [18], we formulate the following lemma that allows for the discrete-time system $\mathcal{H}_d$ in (1) to be transformed into an MLD system $\mathcal{H}_{MLD}$ as in (4). The proof of the next Lemma is given in [15].

Lemma 1: Consider a compact set $\Lambda \subset \mathbb{R}^n$ and a continuous function $p : \Lambda \to \mathbb{R}$. Define

$$M := \max_{x\in\Lambda} p(x), \quad m := \min_{x\in\Lambda} p(x). \tag{16}$$

Given functions $\delta : \Lambda \to \{0,1\}$ and $z : \Lambda \to \mathbb{R}$,

$$z(x) = \delta(x)p(x) \qquad \forall x \in \Lambda \tag{17}$$

holds, if and only if, for each $x \in \Lambda$, the following hold:

$$z(x) \le M\delta(x), \tag{18a}$$

$$z(x) \ge m\delta(x), \tag{18b}$$

$$z(x) \le p(x) - m(1-\delta(x)), \tag{18c}$$

$$z(x) \ge p(x) - M(1-\delta(x)). \tag{18d}$$

□

Now, we are ready to formulate an MLD system associated with the discretized hybrid system $\mathcal{H}_d$. See [15] for a proof.

Theorem 1: Suppose the discretized hybrid dynamical system $\mathcal{H}_d$ in (1) with data $(C, A_f, B_f, c_f, D, A_g, B_g, c_g)$ satisfies Assumption 1 and $\mathcal{X}$ is a compact set. Let A, B_i, E_j for all $i \in \{1, \dots 4\}$ and $j \in \{1, \dots, 5\}$ in (5) take the following values:

$$
\begin{aligned}
&A := A_g,\ B_1 := B_g,\ B_2 := \begin{bmatrix} c_f - c_g & c_f - c_g \end{bmatrix},\\
&B_3 := \begin{bmatrix} A_f - A_g & B_f - B_g & c_f - c_g \end{bmatrix},\ B_4 := c_g,\\
&E_1 := \begin{bmatrix} 0\\0\\0\\0\\0\\0\\0\\0\\h_{12}\\-h_{12}\\h_{22}\\-h_{22}\\0\\I_m\\0\\-I_m \end{bmatrix},\ E_2 := \begin{bmatrix} 0 & 0\\ -1 & 0\\ 0 & -1\\ 1 & 1\\ -M_1 & -M_1\\ -M_2 & -M_2\\ m_1 & m_1\\ m_2 & m_2\\ m_{31}+\sigma_1 & 0\\ M_{31}-\sigma_1 & 0\\ 0 & m_{32}+\sigma_2\\ 0 & M_{32}-\sigma_2\\ -m_1 & -m_1\\ -m_2 & -m_2\\ M_1 & M_1\\ M_2 & M_2 \end{bmatrix},\\
&E_3 := \begin{bmatrix} 0 & 0 & -I_n\\ 0 & 0 & I_n\\ 0 & 0 & I_n\\ 0 & 0 & -I_n\\ I_n & 0 & -M_1\\ 0 & I_n & -M_2\\ -I_n & 0 & m_1\\ 0 & -I_n & m_2\\ 0 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & 0\\ I_n & 0 & -m_1\\ 0 & I_n & -m_2\\ -I_n & & M_1\\ 0 & I_n & M_2 \end{bmatrix}, E_4 := \begin{bmatrix} 0\\0\\0\\0\\0\\0\\0\\0\\h_{11}\\-h_{11}\\h_{21}\\-h_{21}\\I_n\\0\\-I_n\\0 \end{bmatrix}, E_5 := \begin{bmatrix} 0\\0\\0\\1\\0\\0\\0\\0\\\sigma_1\\M_{31}\\\sigma_2\\M_{32}\\-m_1\\-m_2\\M_1\\M_2 \end{bmatrix},
\end{aligned}
\tag{19}
$$

where

$$
\begin{aligned}
&M_1 := \max\{x : (x,u) \in \mathcal{X}\}, \quad m_1 := \min\{x : (x,u) \in \mathcal{X}\},\\
&M_2 := \max\{\tilde{u} : (x,u) \in \mathcal{X}\}, \quad m_2 := \min\{u : (x,u) \in \mathcal{X}\},\\
&M_{31} := \max\{h_1(x,u) : (x,u) \in \mathcal{X}\},\\
&m_{31} := \min\{h_1(x,u) : (x,u) \in \mathcal{X}\},\\
&M_{32} := \max\{h_2(x,u) : (x,u) \in \mathcal{X}\},\\
&m_{32} := \min\{h_2(x,u) : (x,u) \in \mathcal{X}\},
\end{aligned}
\tag{20}
$$

where $\sigma_1, \sigma_2, h_{11}, h_{12}, h_{21}, h_{22}, h_1, h_2$ are given parameters and functions come from (10) and (11). Let $\mathcal{H}_d$ be defined as in (1). Then, for each solution $(k,j) \mapsto (x(k,j), u(k,j))$ to $\mathcal{H}_d$, the function $\ell \mapsto \big(z(\ell), \hat{\delta}_1(\ell), \hat{\delta}_2(\ell), \hat{x}(\ell), \hat{u}(\ell)\big)$ is defined as

$$
\begin{cases}
\hat{\delta}_1(\ell) \in \mathcal{U}_{f1}(x(k,j), u(k,j)), & \text{(21a)}\\
\hat{\delta}_2(\ell) \in \mathcal{U}_{f2}(x(k,j), u(k,j)), & \text{(21b)}\\
\hat{u}(\ell) := u(k,j), & \text{(21c)}\\
\hat{x}(\ell) := x(k,j), & \text{(21d)}\\
z(\ell) = \begin{bmatrix} z_1(\ell)\\ z_2(\ell)\\ z_3(\ell)\end{bmatrix} := \\
\qquad \begin{bmatrix} (\hat{\delta}_1(\ell) + \hat{\delta}_2(\ell) - \hat{\delta}_1(\ell)\hat{\delta}_2(\ell))x(k,j)\\ (\hat{\delta}_1(\ell) + \hat{\delta}_2(\ell) - \hat{\delta}_1(\ell)\hat{\delta}_2(\ell))u(k,j)\\ \hat{\delta}_1(\ell)\hat{\delta}_2(\ell) \end{bmatrix}, & \text{(21e)}
\end{cases}
$$

for each $\ell = k + j$ with $(k,j) \in \operatorname{dom}(x,u)$ is a solution to $\mathcal{H}_{MLD}$ in (4) with

$$
\Psi(z, \hat{\delta}_1, \hat{\delta}_2, \hat{x}, \hat{u}) = \begin{pmatrix}
-z_3 & r_1\\
z_3 - \hat{\delta}_1 & r_2\\
z_3 - \hat{\delta}_2 & r_3\\
-z_3 + \hat{\delta}_1 + \hat{\delta}_2 - 1 & r_4\\
z_1 - M_1(\hat{\delta}_1 + \hat{\delta}_2 + z_3) & r_5\\
z_2 - M_2(\hat{\delta}_1 + \hat{\delta}_2 + z_3) & r_6\\
m_1(\hat{\delta}_1 + \hat{\delta}_2 + z_3) - z_1 & r_7\\
m_2(\hat{\delta}_1 + \hat{\delta}_2 + z_3) - z_2 & r_8\\
(m_{31} + \sigma_1)\hat{\delta}_1 - h_1(\hat{x}, \hat{u}) - \sigma_1 & r_9\\
(M_{31} - \sigma_1)\hat{\delta}_1 + h_1(\hat{x}, \hat{u}) - M_{31} & r_{10}\\
(m_{32} + \sigma_2)\hat{\delta}_2 - h_2(\hat{x}, \hat{u}) - \sigma_2 & r_{11}\\
(M_{32} - \sigma_2)\hat{\delta}_2 + h_2(\hat{x}, \hat{u}) - M_{32} & r_{12}\\
z_1 - \hat{x} + m_1(1 - (\hat{\delta}_1 + \hat{\delta}_2 + z_3)) & r_{13}\\
z_2 - \hat{u} + m_2(1 - (\hat{\delta}_1 + \hat{\delta}_2 + z_3)) & r_{14}\\
\hat{x} - z_1 - M_1(1 - (\hat{\delta}_1 + \hat{\delta}_2 + z_3)) & r_{15}\\
\hat{u} - z_2 - M_2(1 - (\hat{\delta}_1 + \hat{\delta}_2 + z_3)) & r_{16}
\end{pmatrix}.
\tag{22}
$$

and

$$
\begin{aligned}
\Phi(z, \hat{\delta}_1, \hat{\delta}_2, \hat{x}, \hat{u}) = {}& (A_f - A_g)z_1 + (B_f - B_g)z_2\\
&+ (c_f - c_g)(\hat{\delta}_1 + \hat{\delta}_2 + z_3) + A_g\hat{x} + B_g\hat{u} + c_g
\end{aligned}
\tag{23}
$$

defined for each $z = (z_1, z_2, z_3) \in \mathbb{R}^{n+m} \times \{0,1\}$ and $\hat{\delta} = (\hat{\delta}_1, \hat{\delta}_2) \in \{0,1\}^2$. Furthermore, for each solution $\ell \mapsto \big(z(\ell), \hat{\delta}_1(\ell), \hat{\delta}_2(\ell), \hat{x}(\ell), \hat{u}(\ell)\big)$ to $\mathcal{H}_{MLD}$, the function $(k,j) \mapsto (x(k,j), u(k,j))$ defined as

$$
\begin{cases}
x(k,j) := \hat{x}(\ell), & \text{(24a)}\\
u(k,j) := \hat{u}(\ell), & \text{(24b)}
\end{cases}
$$

for each $k = \sum_{i=1}^{\ell} (\hat{\delta}_1(i) + \hat{\delta}_2(i) - \hat{\delta}_1(i)\hat{\delta}_2(i))$ and $j = \ell - k$ with $\ell \in \operatorname{dom}(z, \hat{\delta}_1, \hat{\delta}_2, \hat{x}, \hat{u})$, is a solution to $\mathcal{H}_d$. □

The system in the following example is based on [14, Example 4.1], but modified such that the intersection between C and D is nonempty and the overlap is adjustable by the parameter σ. We relate a solution of $\mathcal{H}_d$ in (1) to a solution of $\mathcal{H}_{MLD}$ in (4) via Theorem 1. Then, we find the MLD solution using an MIQP solver [19].

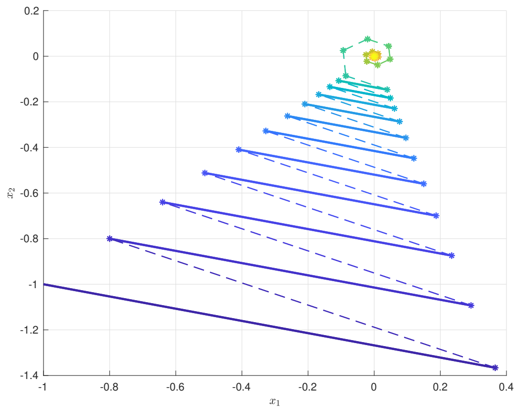

Fig. 1: A trajectory resulting from linear discretized hybrid system of the Example 1 that transformed into MLD system.

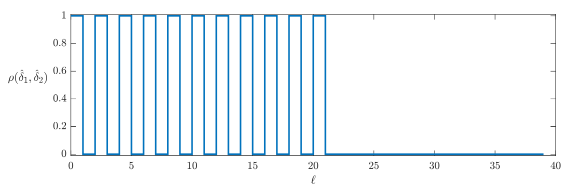

Fig. 2: Binary variable $\rho(\hat{\delta}_1, \hat{\delta}_2)$ obtained from simulation in of the Example 1.

Example 1: Consider the discretized hybrid dynamical system $\mathcal{H}_d$ in (1) with $n = 2$, $m = 1$, and data (A_f, B_f, A_g, C, D) given by

$$A_f = \begin{bmatrix} 0.5 & -0.86 \\ 0.86 & 0.5 \end{bmatrix}, \quad B_f = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad A_g = \begin{bmatrix} 0.4 & 0.69 \\ -0.69 & 0.4 \end{bmatrix},$$
$$B_g = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad H_1 = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad H_2 = 0, \quad \sigma = 0.1,$$
$$c_f = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, c_g = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \tag{25}$$

and

$$\mathcal{X} = \{(x, u) \mid x \in [-10,\ 10]^2,\ u \in [-1,\ 1]\}.$$

We substitute the parameters from (25) into (19) to obtain the parameters of the corresponding MLD system. Then, we find a solution to this MLD system using an MIQP solver. As shown in Fig. 1 and 2, the obtained solution starting from $x(0, 0) = (-1, -1)$ is a solution to the discretized hybrid system given in (25). Figures confirm that, when $(x, u) \in C \setminus D$, then $\rho(\hat{\delta}_1, \hat{\delta}_2) = 1$ and the solution flows according to $x^+ = f(x, u) = A_f x + B_f u$. Furthermore, when $(x, u) \in D \setminus C$, then $\rho(\hat{\delta}_1, \hat{\delta}_2) = 0$ and the solution jumps according to $x^+ = g(x, u) = A_g x + B_g u$. Finally, if $(x, u) \in C \cap D$, then $\rho(\hat{\delta}_1, \hat{\delta}_2) \in \{0, 1\}$ and the solution will either jump or flow. [1]

□

[1]Files for this simulation can be found at the following address: https://github.com/HybridSystemsLab/HybridMPCMLD2Dsystem.git

B. *MIQP version of the Hybrid Optimal Control Problem*

Now, we use Theorem 1, to convert the hybrid optimal control problem in Problem 1 to an MIQP problem. To enforce the prediction horizon constraint, we add two auxiliary variables $\hat{r}_c$ and $\hat{r}_d$ to the proposed $\mathcal{H}_{MLD}$ system in (4). By including $\hat{r}_c$ and $\hat{r}_d$ we can keep track of flows and the number of jumps elapsed. To this end, we rewrite the MLD system with new variables as follows:

$$\mathcal{H}_{\mathcal{MLD}} : \begin{cases} \hat{\zeta}^+ = \begin{bmatrix} \hat{x}^+ \\ \hat{r}_c^+ \\ \hat{r}_d^+ \end{bmatrix} = \begin{bmatrix} \Phi(z, \hat{\delta}_1, \hat{\delta}_2, \hat{x}, \hat{u}) \\ \rho(\hat{\delta}_1, \hat{\delta}_2) + r_c \\ 1 - \rho(\hat{\delta}_1, \hat{\delta}_2) + r_d \end{bmatrix} \\ \Psi(z, \hat{\delta}_1, \hat{\delta}_2, \hat{x}, \hat{u}) \preceq 0, \end{cases} \tag{26}$$

where $\hat{\zeta} := (\hat{x}, \hat{r}_c, \hat{r}_d)$, $\rho(\hat{\delta}_1, \hat{\delta}_2) = \hat{\delta}_1 + \hat{\delta}_2 - \hat{\delta}_1\hat{\delta}_2$ and $\hat{\delta}_1, \hat{\delta}_2, z$, and $\hat{x}$ are given in (21), $\Psi(z, \hat{\delta}_1, \hat{\delta}_2, \hat{x}, \hat{u})$ and $\Phi(z, \hat{\delta}_1, \hat{\delta}_2, \hat{x}, \hat{u})$ are given in (22) and (23), respectively.

Now, considering (21e), $z_1(\ell) = \rho(\hat{\delta}_1(\ell), \hat{\delta}_2(\ell))\hat{x}(\ell)$ and $z_2(\ell) = \rho(\hat{\delta}_1(\ell), \hat{\delta}_2(\ell))\hat{u}(\ell)$, $\mathcal{J}$ in (8) is written as

$$\begin{aligned} \widehat{\mathcal{J}}(z, \hat{\zeta}, \hat{u}) = \sum_{\ell=0}^{N-1} \bigg(& \Big(z_1(\ell)^\top Q_c \hat{x}(\ell) + z_2(\ell)^\top R_c \hat{u}(\ell) \Big) \\ & + \Big(\hat{x}(\ell)^\top Q_d \hat{x}(\ell) + \hat{u}(\ell)^\top R_d \hat{u}(\ell) \Big) \\ & - \Big(z_1(\ell)^\top Q_d \hat{x}(\ell) + z_2(\ell)^\top R_d \hat{u}(\ell) \Big) \bigg) \\ & + \hat{x}(N)^\top P \hat{x}(N). \end{aligned} \tag{27}$$

with $\mathcal{H}_{\mathcal{MLD}}$ defined in (26). The corresponding MIQP problem to Problem 1 to be solved is as follows.

Problem 2: Given an initial condition $z_0 = (z_0, \hat{\zeta}_0) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \times \{0\} \times \{0\}$

$$\begin{aligned} \text{minimize} \quad & \widehat{\mathcal{J}}(z, \hat{\zeta}, \hat{u}) \\ \text{subject to} \quad & (z, \hat{\delta}_1, \hat{\delta}_2, \hat{\zeta}, \hat{u}) \in \widehat{\mathcal{S}}_{\mathcal{H}_{\mathcal{MLD}}}(z_0) \\ & \hat{x}(N) \in X \\ & (r_c(N), r_d(N)) \in \mathcal{T}, \end{aligned} \tag{28}$$

•

where $N \in [\tau_p, 2\tau_p]$ is the terminal time of $(z, \hat{\delta}_1, \hat{\delta}_2, \hat{\zeta}, \hat{u})$, and $\widehat{\mathcal{S}}_{\mathcal{H}_{\mathcal{MLD}}}(z_0)$ is the set of solution pairs of $\mathcal{H}_{\mathcal{MLD}}$ from z_0.

Using (23) and (2), solution to $\mathcal{H}_{\mathcal{MLD}}$ is derived as follows

$$\begin{aligned} \hat{x}(\ell) = \sum_{i=0}^{\ell-1} A^i \big[& B_1 \hat{u}(\ell - 1 - i) \\ & + B_2 \rho(\hat{\delta}_1(\ell - 1 - i), \hat{\delta}_2(\ell - 1 - i)) \\ & . + B_3 z(\ell - 1 - i) + B_4 \big] + A^\ell \hat{x}_0, \end{aligned} \tag{29}$$

for each $\ell = k + j$ with $(k, j) \in \operatorname{dom}(x, u)$ where A and $\{B_i\}_{i=1}^4$ are given in (19). Substituting (29) into (27) and (22) and defining the following vectors

$$\mathcal{Z}(\ell) = \big[z(\ell), \hat{r}_c(\ell), \hat{r}_d(\ell), \hat{u}(\ell), \rho(\hat{\delta}_1(\ell), \hat{\delta}_2(\ell)) \big]^\top \tag{30}$$

$$\mathcal{V} = \begin{bmatrix} \mathcal{Z}(0) \\ \vdots \\ \mathcal{Z}(N-1) \end{bmatrix},$$

Problem 2 is formulated as follows.

Problem 3: Given an initial condition $\mathcal{Z}_0 = \mathcal{Z}(0) \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \times \{0\} \times \{0\} \times \mathbb{R}^m \times \{0,1\}$

$$\begin{aligned} &\text{minimize} \quad \mathcal{V}^\top S_1 \mathcal{V} + 2(S_2 + x_0^\top S_3)\mathcal{V} \\ &\text{subject to} \quad F_1 \mathcal{V} \preceq F_2 + F_3 x_0, \end{aligned} \tag{31}$$

where $\{S_i, F_i\}_{i=1}^3$ are appropriately defined. •

Theorem 2: Suppose a solution to Problem 3 is given by $\mathcal{V}$ and $\mathcal{Z}(\ell)$ for all $\ell \in [0, N-1]$ is given in (30). Then, $((x(0,,0), u(0,0)), \dots, (x(L,J,), u(L,J)))$ is a solution to Problem 1, where $L + J = N - 1$ and

$$\begin{cases} x(k,j) := \sum_{i=0}^{\ell-1} A^i [B_1 \hat{u}(\ell-1-i) \\ \qquad + B_2 \rho(\hat{\delta}_1(\ell-1-i), \hat{\delta}_2(\ell-1-i)) \\ \qquad + B_3 z(\ell-1-i) + B_4] + A^\ell \hat{x}_0, \\ u(k,j) := \hat{u}(\ell), \end{cases} \tag{32}$$

for each $k = \sum_{i=1}^{\ell} (\hat{\delta}_1(i) + \hat{\delta}_2(i) - \hat{\delta}_1(i)\hat{\delta}_2(i))$ and $j = \ell - k$ with $\ell \in \operatorname{dom}\left(z, \hat{r}_c, \hat{r}_d, \hat{u}, \rho(\hat{\delta}_1, \hat{\delta}_2)\right)$. □

See [15] for a proof of Theorem 2.

C. Implementation of Hybrid MPC using MIQP solver

Using Problem 3, an algorithm for solving a hybrid MPC problem with an MIQP solver is given as follows.

Algorithm 1: Implementation of Hybrid MPC by using MIQP Problem 3

Set $i = 0$, $\ell_0 = 0$, $\hat{\zeta}_0 = (\hat{x}_0, 0, 0)$;
while *true* **do**
  Solve Problem 3 to obtain the optimal solution $\mathcal{V}^*$;
  while $\max\{\hat{r}_c(\ell - \ell_i), \hat{r}_d(\ell - \ell_i) \le \tau_c\}$ **do**
    generate trajectory $\hat{x}$ using (32);
  end
  set $i = i + 1$, $\ell_i = \ell$, $\hat{\zeta}_0 = (\hat{x}(\ell_i), 0, 0)$
end

where $\tau_c \le \tau_p$ is a positive integer number and used to parametrize the control horizon. The control horizon regulates the optimization times and it has the same structure as the prediction horizon $\mathcal{T}$ defined in (7).

In the following, we apply Algorithm 1 to solve the hybrid MPC problem for different examples.

V. Examples

Example 2: (Discretized Bouncing Ball) Consider a ball bouncing vertically on a horizontal surface. In [20, p. 27] the bouncing ball is modeled as a point mass with height x_1 and vertical velocity x_2. The motion of the ball evolves according to the following equations:

$$x_1^+ = x_1 + T_s x_2 - T_s^2 \delta, \;\; x_2^+ = x_2 - T_s \delta \quad \text{when} \quad x_1 \ge 0 \tag{33}$$

$$x_1^+ = x_1 - T_s x_2, \;\; x_2^+ = -\lambda x_2 + u \quad \text{when} \quad x_1 = 0 \quad x_2 \le 0, \tag{34}$$

where $\delta = 9.8$, $\lambda \in [0,1]$, and T_s are the gravitational constant, the coefficient of restitution, and the sample time, respectively. When $x_1 \ge 0$, the state $x = (x_1, x_2)$ evolves according to the difference equations $x_1^+ = x_1 + T_s x_2 - T_s^2 \delta$, $x_2^+ = x_2 - T_s \delta$ and impacts occur when the ball reaches the surface with nonpositive velocity; i.e., when $x_1 = 0$ and $x_2 \le 0$. At this point the state is reset according to the difference equations $x_1^+ = x_1 - T_s x_2$, $x_2^+ = -\lambda x_2 + u$. The data of the discretized hybrid dynamical system in (1) for this example is presented as follows

$$A_f = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}, A_g = \begin{bmatrix} 1 & -T_s \\ 0 & -\lambda \end{bmatrix},$$

$$c_f = \begin{bmatrix} -T_s^2 \delta \\ -T_s \delta \end{bmatrix}, c_g = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \tag{35}$$

$$B_f = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, B_g = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

$$\mathcal{X} = \{(x,u) | x \in [0,10] \times [-10,10], u \in [-0.01, 0.01]\} \cap L_V(c),$$

$$C = \{(x,u) \in \mathcal{X} : x_1 \ge 0\}, \tag{36}$$

$$D = \{(x,u) \in \mathcal{X} : x_1 = 0, x_2 \le 0\}, \tag{37}$$

where $L_V(c)$ is the sublevel set of function $V(x_1, x_2) = \frac{1}{2}x_2^2 + \delta x_1$ for a constant c. We need to restrict C and D to a compact set that is forward invariant. To this end, the set $\mathcal{X}$ is defined to be the sublevel set of the function $V(x_1, x_2)$ which is a Lyapunov function for the system [21]. To represent the flow and jump sets, in (36) and (37), in the form given in Assumption 1, we choose the functions $h_1(x_1, x_2) = -x_1$, $h_2(x_1, x_2) = -x_2$ and $\sigma_i = 0$ for each $i \in \{1, 2\}$. The control objective is to minimize the cost functional (8) with $Q_c = 0.2I_2$, $R_c = 0.01$, $Q_d = 0.2I_2$, $R_d = 0.01$, and $P = 0.1I_2$. Also, the prediction and control horizon are given with $\tau_p = 2$ and $\tau_c = 1$, respectively. As shown in Fig. 3, when $(x,u) \in C \setminus D$, then $\rho(\hat{\delta}_1, \hat{\delta}_2) = 1$ and the solution flows according to $f(x,u) = A_f x + B_f u$. When $(x,u) \in D \setminus C$, then $\rho(\hat{\delta}_1, \hat{\delta}_2) = 0$ and the solution jumps according to $g(x,u) = A_g x + B_g u$. Finally, if $(x,u) \in C \cap D$, then $\rho(\hat{\delta}_1, \hat{\delta}_2) \in \{0, 1\}$ and the solution will either jump or flow, and also the control input has adhered to the intended restriction as given in $\mathcal{X}$. [2] □

[2]Files for this simulation can be found at the following address: https://github.com/HybridSystemsLab/HybridMPCMLDBouncingBall.git

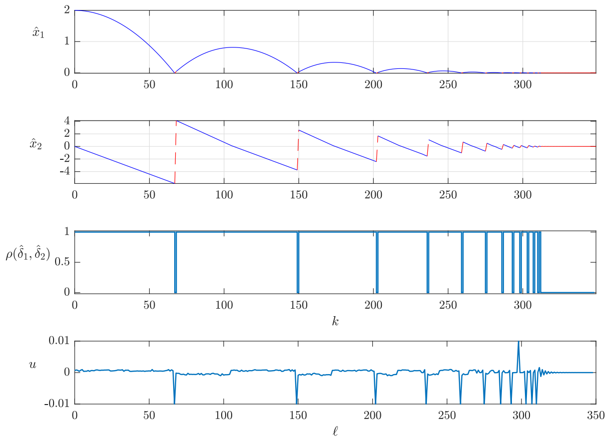

Fig. 3: The solution of the bouncing ball with input resulting from transforming the linear discretized hybrid system into an MLD system.

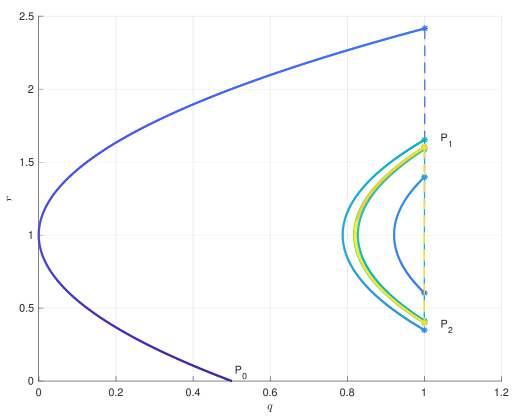

Fig. 4: State Trajectory of TCP in Example 3.

In the next example, we present a congestion control mechanism using in models of transmission control protocols (TCP) with a limit cycle. [3]

Example 3: Consider the congestion control mechanism given by the hybrid system [22]:

$$\begin{cases} \begin{bmatrix} \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} r - B \\ a \end{bmatrix} \quad \text{when} \quad q \in [0, q_{\max}] \\ \begin{bmatrix} q^+ \\ r^+ \end{bmatrix} = \begin{bmatrix} q_{\max} \\ mr \end{bmatrix} \quad \text{when} \quad q = q_{\max}, r \geq B. \end{cases} \tag{38}$$

The flow map is discretized by the sample time $T_s = 0.001$ to present (38) as a discretized hybrid dynamical system as (1). with data as follows:

$$A_f = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + T_s I_2, \quad A_g = \begin{bmatrix} 0 & 0 \\ 0 & m \end{bmatrix},$$
$$c_f = T_s \begin{bmatrix} -B \\ a \end{bmatrix}, \qquad c_g = \begin{bmatrix} q_{\max} \\ 0 \end{bmatrix},$$
$$B_f = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \qquad B_g = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

In Fig 4, the solution obtained by an MIQP solver approaches a limit cycle with one jump per period with parameters $B = 1$, $a = 1$, $m = 0.25$, and $q_{\max} = 0.25$. The initial condition for the simulation is $P_0 = (\frac{B^2}{2a}, 0) = (0.5, 0)$. When the solution reaches to point P_1, it jumps to P_2 and then flows to P_1, and this repetition makes a limit cycle with initial point of $P_2 = (1, 0.4)$. □

In the next example, we add MPC to the system given in Example 1 and solve the hybrid MPC problem via MIQP solvers.

Example 4: Consider the hybrid system $\mathcal{H}_d$ given in Example 1, and cost functional (8) with $Q_c = 0.2I_2$, $R_c = 0.1$, $Q_d = 0.1I_2$, $R_d = 0.1$, and $P = 0.01I_2$. The prediction and control horizon are given as $\tau_p = 2$ and $\tau_c = 1$, respectively.[4]

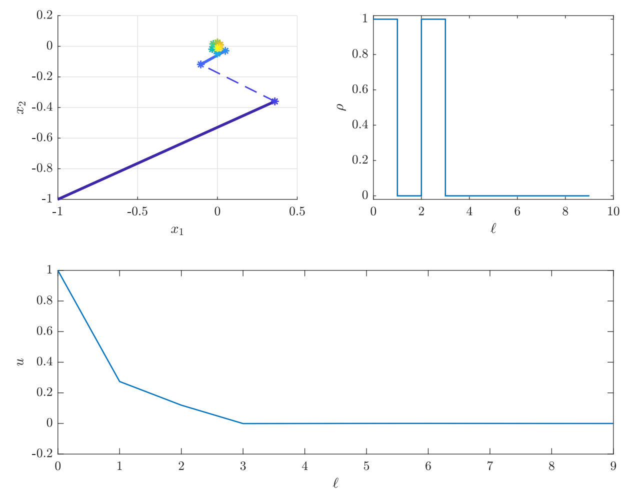

Fig. 5: Model predictive control of system (1) in Example 4 resulting from problem 3.

As shown in Fig. 5, when $(x, u) \in C \setminus D$, then $\rho(\hat{\delta}_1, \hat{\delta}_2) = 1$ and the solution flows according to $x^+ = f(x, u) = A_f x + B_f u$. Furthermore, when $(x, u) \in D \setminus C$, then $\rho(\hat{\delta}_1, \hat{\delta}_2) = 0$ and the solution jumps according to $x^+ = g(x, u) = A_g x + B_g u$. Finally, if $(x, u) \in C \cap D$, then $\rho(\hat{\delta}_1, \hat{\delta}_2) \in \{0, 1\}$ and the solution either jumps or flow. Note that the control input has adhered to the intended restriction as given in $\mathcal{X}$. □

VI. Conclusion

In this paper, a new mixed-integer model predictive control approach for discretized hybrid systems is presented. To

[3]Files for this simulation can be found at the following address: https://github.com/HybridSystemsLab/HybridMPCMLDCongestionControl.git

[4]Files for this simulation can be found at the following address: https://github.com/HybridSystemsLab/HybridMPCMLD2Dsystem.git

solve the formulated MPC problem for the discretized hybrid dynamical system, boolean algebra is employed to formulate a mixed integer quadratic program for the transformed MLD system. The proposed approach consists of converting the discretized hybrid system into a nonlinear discrete-time system and transforming the converted nonlinear discrete-time system into an MLD system using McCormick Relaxation. Our results establish that solving MPC for the discretized hybrid dynamical system, namely $\mathcal{H}_d$ in (1) is equivalent to solving Problem 3 for the MLD system, namely $\mathcal{H}_{\mathcal{MLD}}$ in (26). The proposed MPC algorithm for the discretized hybrid system is applied to several examples.

REFERENCES

[1] J. B. Rawlings and D. Q. Mayne. *Model predictive control: Theory and design*. Nob Hill Pub., 2009.

[2] Carlos E Garcia, David M Prett, and Manfred Morari. Model predictive control: Theory and practice—a survey. *Automatica*, 25(3):335–348, 1989.

[3] K. Zhang, J. Sprinkle, and R. G. Sanfelice. Computationally-aware switching criteria for hybrid model predictive control of cyber-physical systems. *IEEE Transactions on Automation Science and Engineering*, 13:479–490, 2016.

[4] Moritz Diehl, Hans Joachim Ferreau, and Niels Haverbeke. Efficient numerical methods for nonlinear mpc and moving horizon estimation. In *Nonlinear model predictive control*, pages 391–417. Springer, 2009.

[5] Alberto Bemporad and Domenico Mignone. miqp.m: a matlab function for solving mixed integer quadratic programs version 1.02-user guide. 2000.

[6] F. Borrelli, A. Bemporad, and M. Morari. *Predictive control for linear and hybrid systems*. Cambridge University Press, 2017.

[7] B. Altin, P. Ojaghi, and R. G. Sanfelice. A model predictive control framework for hybrid systems. In *NMPC*, volume 51, pages 128–133, August 2018.

[8] B. Altin and R. G. Sanfelice. Asymptotically stabilizing model predictive control for hybrid dynamical systems. In *Proceedings of the American Control Conference*, July 2019.

[9] Berk Altın and Ricardo G Sanfelice. Model predictive control for hybrid dynamical systems: Sufficient conditions for asymptotic stability with persistent flows or jumps. In *2020 American Control Conference (ACC)*, pages 1791–1796. IEEE, 2020.

[10] R. Goebel, R. G. Sanfelice, and A. R. Teel. *Hybrid Dynamical Systems: Modeling, Stability, and Robustness*. Princeton University Press, New Jersey, 2012.

[11] R. G. Sanfelice. *Hybrid Feedback Control*. Princeton University Press, January 2021.

[12] Pegah Ojaghi, Berk Altın, and Ricardo G Sanfelice. A model predictive control framework for asymptotic stabilization of discretized hybrid dynamical systems. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 2356–2361. IEEE, 2019.

[13] R. G. Sanfelice and A. R. Teel. Dynamical properties of hybrid systems simulators. *Automatica*, 46(2):239–248, 2010.

[14] Alberto Bemporad and Manfred Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, 1999.

[15] Iman Nodozi and Ricardo G. Sanfelice. A Mixed Integer Approach for the Solution of Hybrid Model Predictive Control Problems. Technical report, University of California, Santa Cruz, Department of Electrical and Computer Engineering, 2022. https://hybrid.soe.ucsc.edu/sites/default/files/preprints/TR-HSL-03-2022.pdf, Password: hMPC-MLD.

[16] Berk Altın, Pegah Ojaghi, and Ricardo G Sanfelice. A model predictive control framework for hybrid dynamical systems. *IFAC-PapersOnLine*, 51(20):128–133, 2018.

[17] Alexandre S Freire, Eduardo Moreno, and Juan Pablo Vielma. An integer linear programming approach for bilinear integer programming. *Operations Research Letters*, 40(2):74–77, 2012.

[18] Akshay Gupte, Shabbir Ahmed, Myun Seok Cheon, and Santanu Dey. Solving mixed integer bilinear problems using milp formulations. *SIAM Journal on Optimization*, 23(2):721–744, 2013.

[19] A. Bemporad. Hybrid Toolbox - User's Guide, 2004. http://cse.lab.imtlucca.it/~bemporad/hybrid/toolbox.

[20] Alberto Bemporad and MPC Explicit. Model predictive control of hybrid systems. *University of Siena*, 2007.

[21] Jun Chai and Ricardo G Sanfelice. On notions and sufficient conditions for forward invariance of sets for hybrid dynamical systems. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 2869–2874. IEEE, 2015.

[22] Joao P Hespanha, Stephan Bohacek, Katia Obraczka, and Junsoo Lee. Hybrid modeling of TCP congestion control. In *International Workshop on Hybrid Systems: Computation and Control*, pages 291–304. Springer, 2001.

APPENDIX

A. Intermediate Step: Converting the discretized hybrid control system $\mathcal{H}_d$ into a nonlinear discrete-time system (Step 1)

We introduce a new state $\tilde{x}$ and input $\tilde{u}$, which play the role of x and u in $\mathcal{H}_d$, respectively. The new discrete-time control system is defined as

$$\tilde{\mathcal{H}}_d : \begin{cases} \tilde{x}^+ \in \bigcup_{\substack{u_{f1} \in \mathcal{U}_1(\tilde{x},\tilde{u}) \\ u_{f2} \in \mathcal{U}_2(\tilde{x},\tilde{u})}} \{\rho(u_{f1}, u_{f2}) f(\tilde{x}, \tilde{u}) \\ \qquad\qquad + (1 - \rho(u_{f1}, u_{f2})) g(\tilde{x}, \tilde{u})\} \\ \qquad\qquad (\tilde{x}, \tilde{u}) \in C \cup D, \end{cases} \tag{39}$$

where $\rho(u_{f1}, u_{f2}) = u_{f1} + u_{f2} - u_{f1}u_{f2}$ determines whether the state $\tilde{x}$ flows or jumps, and C and D are given in (10) and (11), respectively. A notion of a solution to $\tilde{\mathcal{H}}_d$ is defined as follows.

Definition 4: A function $\mathcal{M} \ni \ell \mapsto (\tilde{x}(\ell), \tilde{u}(\ell), u_{f1}(\ell), u_{f2}(\ell))$ is a solution to $\tilde{\mathcal{H}}_d$ in (39) if it satisfies

$$\begin{aligned} \tilde{x}(\ell+1) \in \bigcup_{\substack{u_{f1}(\ell) \in \mathcal{U}_{f1}(\tilde{x},\tilde{u}) \\ u_{f2}(\ell) \in \mathcal{U}_{f2}(\tilde{x},\tilde{u})}} \{ & (u_{f1}(\ell) + u_{f2}(\ell) \\ & - u_{f1}(\ell)u_{f2}(\ell)) f(\tilde{x}(\ell), \tilde{u}(\ell)) \\ + (1 - (u_{f1}(\ell) + u_{f2}(\ell) & - u_{f1}(\ell)u_{f2}(\ell))) g(\tilde{x}(\ell), \tilde{u}(\ell))\} \end{aligned} \tag{40}$$

for all $\ell \in \mathcal{M}$ such that $\ell + 1 \in \mathcal{M}$, where $\mathcal{M}$ is of the form $\{0, 1, \ldots, K\}$, with K finite, or equal to $\mathbb{N}$. When $\mathcal{M} = \mathbb{N}$, the solution is said to be complete. •

The following result establishes a relationship between the solutions to $\mathcal{H}_d$ in (1) and to $\tilde{\mathcal{H}}_d$ in (39). See [15] for a proof.

Lemma 2: For each solution $(k, j) \mapsto (x(k, j), u(k, j))$ to $\mathcal{H}_d$ in (1), the function $\ell \mapsto (\tilde{x}(\ell), \tilde{u}(\ell), u_{f1}(\ell), u_{f2}(\ell))$ defined as

$$\begin{aligned} &\tilde{x}(\ell) := x(k, j), \quad \tilde{u}(\ell) := u(k, j), \\ &u_{f1}(\ell) \in \mathcal{U}_{f1}(x(k, j), u(k, j)), \\ &u_{f2}(\ell) \in \mathcal{U}_{f2}(x(k, j), u(k, j)), \end{aligned} \tag{41}$$

for each $\ell = k + j$ with $(k, j) \in \mathrm{dom}(x, u)$, is a solution to $\tilde{\mathcal{H}}_d$ in (39). Also, for each solution $\ell \mapsto (\tilde{x}(\ell), \tilde{u}(\ell), u_{f1}(\ell), u_{f2}(\ell))$ to $\tilde{\mathcal{H}}_d$, the function $(k, j) \mapsto (x(k, j), u(k, j))$ defined as

$$x(k, j) := \tilde{x}(\ell), \qquad u(k, j) := \tilde{u}(\ell), \tag{42}$$

for each $k = \sum_{i=1}^{\ell} (u_{f1}(i) + u_{f2}(i) - u_{f1}(i)u_{f2}(i))$ and $j = \ell - k$ with $\ell \in \mathrm{dom}(\tilde{x}, \tilde{u}, u_{f1}, u_{f2})$, is a solution to $\mathcal{H}_d$. □