Solving Hybrid Model Predictive Control Problems via a Mixed-Integer Approach

Iman Nodozi[0009-0006-9407-8934] and
Ricardo G. Sanfelice[0000-0002-6671-5362]

Abstract This book chapter presents a method to efficiently solve hybrid model predictive control (MPC) problems for a class of discretized hybrid control systems. The proposed method recasts the optimal control problem associated with hybrid MPC into a mixed-integer quadratic problem (MIQP), which can be efficiently solved using well-established algorithms. The approach consists of transforming the discretized hybrid control system into a mixed logical dynamical (MLD) system. This transformation enables the use of MIQP tools for the solution of the hybrid MPC problem. To arrive to such a system, an intermediate step converting the discretized hybrid control system into a discrete-time control system with set-valued dynamics is formulated. The proposed method is illustrated in several examples that demonstrate the ability of the approach to handle state jumps and logic variables present in the hybrid control system, highlighting its suitability for real-world applications featuring hybrid dynamics.

1 Introduction

Model predictive control (MPC) is an effective feedback control method due to ensuring asymptotic stability, optimality, and constraint adherence [1]. As an approach grounded in optimization, MPC often incurs high computational costs and is significantly influenced by the efficacy of the optimization method utilized. The computation-intensive nature of MPC, mainly due to time involved in the solution

Iman Nodozi
Department of Electrical and Computer Engineering, University of California, Santa Cruz, CA, USA , e-mail: inodozi@ucsc.edu

Ricardo G. Sanfelice
Department of Electrical and Computer Engineering, University of California, Santa Cruz, CA, USA, e-mail: ricardo@ucsc.edu

of the optimization problem, is well-documented in the literature, including studies such as [2] and [3], highlighting the potentially large amount of time required for the optimization schemes to converge. In many MPC applications, the optimal control problem (OCP) that is to be solved at each computation event is nonlinear. Methods are available in the literature to cope with such nonlinearity, including sequential quadratic programming, penalty methods, Lagrangian-based approaches, interior point methods, among others. The survey [4] provides a detailed presentation of techniques designed to handle nonlinear optimization problems emerging in MPC for continuous-time and discrete-time control systems. However, methods for the solution of MPC problems for hybrid control systems – namely, systems with continuous and discrete dynamics (see [5] and [6, Section 2.2 of Chapter 2]) – are much less developed. For hybrid control systems given in terms of a discrete-time piecewise affine (PWA) system or a mixed logical dynamical (MLD) system, the works of [7, 8] show that the OCP associated to the MPC problem with quadratic cost can be recast as a mixed-integer quadratic problem (MIQP). This approach is powerful as it allows for the use of MIQP solvers already available in the literature.

The general framework for robust asymptotic stability and control design for hybrid dynamical systems in [9] and [10] was recently augmented by the development of a hybrid MPC approach; see [11, 12, 13]. The hybrid system models therein are referred to as *hybrid inclusions/equations*, and are useful for a wide range of applications due to allowing for continuous-valued and discrete-valued states, set-valued and nonlinear dynamics, nonunique solutions, Zeno behavior, and beyond. Hybrid equations describe the continuous evolution (referred to as *flow*) of the state in terms of differential equations, and of the discrete events (referred to as *jumps*) via difference equations. Whether the state flows or jumps depends on conditions formulated in terms of sets, referred to as *flow set* and *jump set*; for more details, see [9]. While the work in [11, 12, 13] address most of the key theoretical aspects of hybrid MPC for such broad class of hybrid systems, the work therein does not delve into the numerical solution of the optimal control problem formulated for the control of the hybrid dynamical system. This paper aims at filling this gap. Specifically, inspired by the effectiveness of formulating a mixed-integer quadratic problem (MIQP) to solve MPC problems for discrete-time, continuous-time, and certain classes of hybrid systems, this paper presents a method enabling the use of MIQP solvers for hybrid equations.

In our pursuit of finding effective solutions to hybrid MPC problems with hybrid equations, we considered a discretized version of hybrid equations – as in [14] and [15] – and devise an efficient method for solving the related optimal control problem. To handle binary variables, we employ the McCormick Relaxation technique to reframe the hybrid MPC problem as a mixed-integer quadratic problem (MIQP). Specifically, we address *discretized hybrid equations* characterized by linear flow and jump maps, and flow and jump sets defined by inequalities related to the state and input. Building on the concepts in [16], we develop a (discrete-time) mixed logical dynamical (MLD) system model for the hybrid equation, subsequently reformulating the OCP of hybrid MPC as an MIQP. Our approach involves two key transformations.

The first transformation converts the discretized hybrid equation into a discrete-time system with binary variables and set-valued dynamics. The second transformation consists of reformulating the discrete-time system into an MLD system. To establish a solid mathematical foundation for these transformations, we show equivalences between the solutions to each system. These results allow us to link the solution of the OCP for the equivalent MLD system, which is obtainable via MIQP solvers, with the solution of the OCP related to the discretized hybrid equation. As a result, we derive an MIQP-based solution for hybrid MPC tailored to the class of systems considered. Additionally, we present an algorithm to numerically implement our method and demonstrate its application in various examples. The examples include a controlled bouncing ball system and a congestion control mechanism used in transmission control protocol models. These examples showcase the practicality and effectiveness of the proposed approach.

In [17], we presented a preliminary version of this work. This manuscript expands on it, providing detailed steps and methodologies not covered in the conference version, and further illustrating the effectiveness of the proposed approach with more examples.

Organization

This paper is structured as follows. In Section 2, we introduce the concepts of discretized hybrid control systems, mixed logical dynamical systems, and the solutions associated with them. Section 3 is dedicated to formulating the MPC problem for discretized hybrid dynamical systems. In Section 4, we formulate the mixed-integer approach to model predictive control for these systems. This formulation is followed by Section 5, where we present numerical simulations of several examples.

2 Preliminaries

2.1 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 the 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.

2.2 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 $(x,u) \in C \cup D \cup \{(g(x,u),u) \mid (x,u) \in D\} =: \mathcal{X} \subset \mathbb{R}^n \times \mathbb{R}^m$ are the state and the input of the system, respectively. The set C is called the flow set, and D is 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.

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

Definition 2 (Concept of solution pair [14]) A pair $x : \text{dom}\, x \to \mathbb{R}^n$, $u : \text{dom}\, u \to \mathbb{R}^m$ is a solution pair to $\mathcal{H}_d$ if the following conditions hold:

(S1) dom x = dom u is a discrete hybrid time domain,

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

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

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

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

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

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

2.3 Mixed Logical Dynamical Systems

A general MLD model is given by [16]

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

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

where $\hat{x} \in \mathbb{R}^n$ and $\hat{u} \in \mathbb{R}^m$ are the state and the input of the system, respectively. The auxiliary continuous and binary variables are represented by $z \in \mathbb{R}^{n_d}$ and $\hat{\delta} \in \{0, 1\}^{m_d}$, respectively. The matrices A, $\{B_i\}_{i=1}^3$, B_4, and $\{E_i\}_{i=1}^5$ have appropriate dimensions. The MLD model in (2)-(3) can be expressed in the compact form

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

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

$$\begin{aligned} &\hat{x}(\ell+1) = A\hat{x}(\ell) + B_1\hat{u}(\ell) + B_2\hat{\delta}(\ell) + B_3 z(\ell) + B_4, \qquad \forall \ell : \ell, \ell+1 \in \mathcal{M} \\ &E_2\hat{\delta}(\ell) + E_3 z(\ell) \leq E_1\hat{u}(\ell) + E_4\hat{x}(\ell) + E_5 \qquad \ell \in \mathcal{M}, \end{aligned} \tag{6}$$

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

MLD models have shown remarkable utility in complex control systems, demonstrating their importance in theoretical and practical applications. For instance, in the control of gas supply systems, MLD models adeptly handle the intricate balance between the physical flow of gas and the operational control mechanisms [16]. In the area of digital communications, our work explores the novel application of MLD models to Transmission Control Protocols (TCP) [23]. Additionally, their applicability is evident in the bouncing ball system, a key example in control theory that demonstrates the integration of continuous and discrete dynamics. Detailed explorations of the latter two examples, specifically the Transmission Control Protocols (TCP) and the bouncing ball system, will be provided in the later sections of this document, showcasing the versatility and effectiveness of MLD models in diverse scenarios.

3 Hybrid Model Predictive Control for Discretized Hybrid Control Systems

This section formulates an MPC problem for a discretized hybrid control system given by $\mathcal{H}_d$ in (1). Based on the framework given in [13], we first introduce some details related to MPC for discretized hybrid systems.

3.1 Prediction Horizon

A fixed end-time optimal control problem fits well with continuous or discrete-time Model Predictive Control (MPC), where we update the controls periodically, and each computed control input has the same final moment. But, due to the nature of (discrete) hybrid time domains, using a fixed end-time optimal control problem is restrictive [13]. For (discretized) hybrid control systems, the solutions might flow or jump, so the prediction horizon must accommodate solutions having different discrete hybrid time domains. To address these issues, as in [13], 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. This means that for some $\tau_p \in \{1, 2, \ldots\}$, the terminal time (T, J) of any feasible solution pair satisfies $\max\{T, J\} = \tau_p$.

3.2 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 $\operatorname{dom}(x, u) = \bigcup_{j=0}^{J} \bigcup_{k=K_j}^{K_{j+1}} (k, j)$, and $K_{J+1} = L$, and $X \subset \Pi(\mathcal{X})$ be the terminal constraint set. If $x(L, J) \in X$, then the cost of the pair (x, u) is given by

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

In the definition of the cost functional $\mathcal{J}$, L_C is called the *flow cost* defined on the flow set C, L_D is called the *jump cost* defined on the jump set D, and V is called the *terminal cost* defined on the terminal constraint set X.

3.3 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 cost functional $\mathcal{J}$.

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 a MIQP solver.

4 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, which we denote $\tilde{\mathcal{H}}_d$;

Step 2) The new discrete-time control system $\tilde{\mathcal{H}}_d$ is converted into a MLD system, which we denote $\mathcal{H}_{MLD}$;

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 given in terms of a MLD system. In this step, we convert the discretized hybrid control system $\mathcal{H}_d$ into a nonlinear discrete-time system. This conversion is technical and described in full detail 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 and the jump set D are given as*

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

and

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

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

$$C_i = \{(x, u) \in \mathcal{X} : h_i(x, u) - \sigma_i \leq 0\}, \tag{12}$$
$$D_i = \{(x, u) \in \mathcal{X} : h_i(x, u) + \sigma_i \geq 0\}, \tag{13}$$

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

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

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

Note that the subscript f in $\mathcal{U}_{fi}$ and u_{fi} is merely a naming convention and does not indicate any relationship with the function f. We exploit the MLD system structure enabled by Assumption 1 to formulate a MIQP version of Problem 1. For this purpose, we impose the following assumption on the flow cost, jump cost, and terminal cost involved 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 $P \geq 0$, $Q_c \geq 0$, $R_c > 0$, $Q_d \geq 0$, and $R_d > 0$.

4.1 Recasting $\mathcal{H}_d$ as a MLD system

Employing the McCormick Relaxation, as detailed in [18] and [19], we formulate a key lemma to facilitate the transformation of the discrete-time system $\mathcal{H}_d$, as defined in (1), into an MLD system, denoted $\mathcal{H}_{MLD}$, following the structure in (4). The McCormick Relaxation is used to linearize the product of a binary variable $\delta \in \{0, 1\}$ and a continuous variable x, commonly encountered in mixed-integer programming. This linearization is key in transforming nonlinear constraints into a linear form suitable for MLD modeling, effectively enabling the conversion of the system $\mathcal{H}_d$ into $\mathcal{H}_{MLD}$.

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:

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

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

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

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

Proof. Since Λ is compact and p is continuous, M and m in (16) are well-defined. To prove the equivalence between (17) and (18), we consider the two possible values of $\delta(x) \in \{0, 1\}$:

- **Case 1:** If $\delta(x) = 1$, then (17) implies $z(x) = p(x)$. The inequalities (18a) and (18b) reduce to $m \leq p(x) \leq M$, which are true by the definition of m and M. The inequalities (18c) and (18d) simplify to $z(x) = p(x)$, which also holds.
- **Case 2:** If $\delta(x) = 0$, then (17) implies $z(x) = 0$. The inequalities (18a) and (18b) reduce to $0 \leq 0$, which is trivially true. The inequalities (18c) and (18d) simplify to $m \leq p(x) \leq M$, which again hold by the definition of m and M.

Therefore, (17) holds if and only if all the inequalities in (18) hold. □

After presenting Lemma 1, we are now prepared to formulate an MLD system associated with the discretized hybrid system $\mathcal{H}_d$. In the proof of the next theorem, we utilize Lemma 1 to efficiently translate the nonlinear interactions between variables into a series of manageable linear inequalities.

Theorem 1. *Suppose the discretized hybrid control 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 $i \in \{1, 2, \ldots 4\}$ and $j \in \{1, 2, \ldots, 5\}$ in* (5) *take the following values:*

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

where

$$
\begin{aligned}
&M_1 := \max\{x(k,j) : (k,j) \in (K,J)\},\ \ m_1 := \min\{x(k,j) : (k,j) \in (K,J)\},\\
&M_2 := \max\{u(k,j) : (k,j) \in (K,J)\},\ \ m_2 := \min\{u(k,j) : (k,j) \in (K,J)\},\\
&M_{31} := \max\{h_1(x(k,j),u(k,j)) : (k,j) \in (K,J)\},\\
&m_{31} := \min\{h_1(x(k,j),u(k,j)) : (k,j) \in (K,J)\},\\
&M_{32} := \max\{h_2(x(k,j),u(k,j)) : (k,j) \in (K,J)\},\\
&m_{32} := \min\{h_2(x(k,j),u(k,j)) : (k,j) \in (K,J)\},
\end{aligned}
\tag{20}
$$

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

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

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

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

and

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

defined for each $z=(z_1,z_2,z_3)\in\mathbb{R}^{n+m}\times\{0,1\}$ *and each* $\hat{\delta}=(\hat{\delta}_1,\hat{\delta}_2)\in\{0,1\}^2$. *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*

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

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* $\mathcal{H}_d$.

Proof. Since $\mathcal{X}$ is compact, and h_i are continuous, then $\{M_q,\ m_q\}_{q=1}^2$ and $\{M_{3q},\ m_{3q}\}_{q=1}^3$ are well defined.

Pick a solution $(k,j)\mapsto(x(k,j),u(k,j))$ to $\mathcal{H}_d$. Considering (10) and (11), we conclude that

$$\begin{aligned} C_i\setminus D_i &= \{(\tilde{x},\tilde{u})\in\mathcal{X} \mid h_i(\tilde{x},\tilde{u})+\sigma_i<0\}, \\ D_i\setminus C_i &= \{(\tilde{x},\tilde{u})\in\mathcal{X} \mid h_i(\tilde{x},\tilde{u})-\sigma_i>0\}. \end{aligned} \tag{25}$$

Let us define $\hat{\delta}_1(\ell)\in\mathcal{U}_{f1}((x(k,j),u(k,j)))$, and $\hat{\delta}_2(\ell)\in\mathcal{U}_{f2}((x(k,j),u(k,j)))$, for each $\ell=k+j$ with $(k,j)\in\operatorname{dom}(x,u)$. If the solution $(k,j)\mapsto(x(k,j),u(k,j))$ is such that $((x(k,j),u(k,j))\in C_1\setminus C_2$, then

$$[h_1((x(k,j),u(k,j))+\sigma_1<0] \implies [\hat{\delta}_1(\ell)=1]. \tag{26}$$

Similarly, if $((x(k,j),u(k,j)) \in D_1 \setminus D_2$, then

$$[h_1((x(k,j),u(k,j)) - \sigma_1 > 0] \implies [\hat{\delta}_1 = 0]. \tag{27}$$

Note that

$$X_1 \implies X_2 \tag{28a}$$
$$\Updownarrow$$
$$(\sim X_1) \vee X_2. \tag{28b}$$

Furthermore, if $x \in \Lambda$ and $q : \Lambda \to \mathbb{R}$ is a continuous function,

$$[q(x) \leq 0] \;\vee\; [\delta(x) = 1] \tag{29a}$$
$$\Updownarrow$$
$$q(x) \leq \tilde{M}\delta(x) \tag{29b}$$

where $\tilde{M} := \max_{x \in \Lambda} q(x)$. Now, using (28) with X_1 given by $[h_1((x(k,j),u(k,j)) + \sigma_1 < 0]$, and X_2 by $[\hat{\delta}_1(\ell) = 1]$, given $\ell = k+j$ with $(k,j) \in \text{dom}\,(x,u)$, implication (26) is equivalent to

$$[h_1((x(k,j),u(k,j)) + \sigma_1 \geq 0] \vee [\hat{\delta}_1(\ell) = 1]. \tag{30}$$

Using (29) with $q = -(h_1 + \sigma_1)$, and $\delta = \hat{\delta}_1$, we conclude that (26) holds if and only if

$$-h_1((x(k,j),u(k,j)) - \sigma_1 \leq (-m_{31} - \sigma_1)\hat{\delta}_1(\ell) \tag{31}$$

holds.

Similarly, using (28) with X_1 given by $[h_1((x(k,j),u(k,j)) - \sigma_1 > 0]$, and X_2 by $[\hat{\delta}_1(\ell) = 0]$, given $\ell = k+j$ with $(k,j) \in \text{dom}\,(x,u)$, implication (27) is equivalent to

$$[h_1((x(k,j),u(k,j)) - \sigma_1 \leq 0] \vee [\hat{\delta}_1(\ell) = 0]. \tag{32}$$

Now, using (29) with $q = h_1 - \sigma_1$ and $\delta = 1 - \hat{\delta}_1$, we conclude that (32) holds if and only if

$$h_1((x(k,j),u(k,j)) - \sigma_1 \leq (M_{31} - \sigma_1)(1 - \hat{\delta}_1(\ell)) \tag{33}$$

holds. Let us define $\hat{x}(\ell) = x(k,j)$ and $\hat{u}(\ell) = u(k,j)$, for each $\ell = k+j$ with $(k,j) \in \text{dom}\,(x,u)$.

Then, $(\hat{\delta}_1(\ell), \hat{x}(\ell), \hat{u}(\ell))$ for all $\ell \in \text{dom}\,(z,\hat{\delta},\hat{x},\hat{u})$ satisfies (31) and (33) or equivalently makes the lines r_9 and r_{10} of Ψ in (22) nonpositive.

Similarly, for the case that $(k,j) \mapsto (x(k,j),u(k,j))$ is such that $(x(k,j),u(k,j)) \in C_2 \setminus C_1$ or $(x(k,j),u(k,j)) \in D_2 \setminus D_1$, then, $(\hat{\delta}_2(\ell), \hat{x}(\ell), \hat{u}(\ell))$ for all $\ell \in \text{dom}\,(z,\hat{\delta},\hat{x},\hat{u})$ make the lines r_{11} and r_{12} of Ψ in (22) nonpositive.

Now, consider (21e). Using Lemma 1 for each $\ell \in \text{dom}\,(z,\hat{\delta},\hat{x},\hat{u})$ with $p = \hat{\delta}_1(\ell)$, $\delta = \hat{\delta}_2(\ell)$, and $z = z_3$, we obtain that,

$$z_3(\ell) = \hat{\delta}_1(\ell)\hat{\delta}_2(\ell) \tag{34}$$

if and only if the following inequalities hold

$$\begin{aligned} z_3(\ell) &\leq \hat{\delta}_2(\ell), \\ z_3(\ell) &\geq 0, \\ z_3(\ell) &\leq \hat{\delta}_2(\ell), \\ z_3(\ell) &\geq \hat{\delta}_1(\ell) - (1 - \hat{\delta}_2(\ell)), \end{aligned} \tag{35}$$

or, equivalently, the lines r_1 to r_4 of Ψ in (22) are nonpositive. Using Lemma 1 for each $\ell \in \mathrm{dom}\,(z, \hat{\delta}, \hat{x}, \hat{u})$, with $p = \hat{x}(\ell)$, $\delta = z_3(\ell)$, and $z = z_1$, we obtain that,

$$z_2(\ell) = (u_{f1}(\ell) + u_{f2}(\ell) - z_3(\ell))\hat{x}(\ell) \tag{36}$$

if and only if the following inequalities hold:

$$\begin{aligned} z_2(\ell) &\leq M_1(\hat{\delta}_1(\ell) + \hat{\delta}_2(\ell) - z_3(\ell)), \\ z_2(\ell) &\geq m_1(\hat{\delta}_1(\ell) + \hat{\delta}_2(\ell) - z_3(\ell)), \\ z_2(\ell) &\leq \hat{x}(\ell) - m_1(1 - (\hat{\delta}_1(\ell) + \hat{\delta}_2(\ell) - z_3(\ell))), \\ z_2(\ell) &\geq \hat{x}(\ell) - M_1(1 - (\hat{\delta}_1(\ell) + \hat{\delta}_2(\ell) - z_3(\ell))). \end{aligned} \tag{37}$$

Considering (21a) and (21b), $(z(k), \hat{\delta}_1(\ell), \hat{\delta}_2(\ell), \hat{x}(\ell), \hat{u}(k))$ for all $k \in \mathrm{dom}\,(z, \hat{\delta}, \hat{x}, \hat{u})$ satisfy (37) or equivalently make the lines r_5, r_7, r_{13}, and r_{15} of Ψ in (22) nonpositive.

Similarly, using Lemma 1 for each $\ell \in \mathrm{dom}\,(z, \hat{\delta}, \hat{x}, \hat{u})$, with $p = \tilde{u}(\ell)$, $\delta = z_3(\ell)$, and $z = z_2(\ell)$, we obtain that,

$$z_2(\ell) = (\hat{\delta}_1(\ell) + \hat{\delta}_2(\ell) - z_3(\ell))\hat{u}(\ell) \tag{38}$$

if and only if the following inequalities hold

$$\begin{aligned} z_2(\ell) &\leq M_2(\hat{\delta}_1(\ell) + \hat{\delta}_2(\ell) - z_3(\ell)), \\ z_2(\ell) &\geq m_2(\hat{\delta}_1(\ell) + \hat{\delta}_2(\ell) - z_3(\ell)), \\ z_2(\ell) &\leq \hat{u}(\ell) - m_2(1 - (\hat{\delta}_1(\ell) + \hat{\delta}_2(\ell) - z_3(\ell))), \\ z_2(\ell) &\geq \hat{u}(\ell) - M_2(1 - (\hat{\delta}_1(\ell) + \hat{\delta}_2(\ell) - z_3(\ell))). \end{aligned} \tag{39}$$

Considering (21a) and (21b), $(z(\ell), \hat{\delta}_1(\ell), \hat{\delta}_2(\ell), \hat{x}(\ell), \hat{u}(\ell))$ for all $\ell \in \mathrm{dom}\,(z, \hat{\delta}, \hat{x}, \hat{u})$ satisfies (39) or, equivalently, makes the lines r_6, r_8, r_{14}, and r_{16} of Ψ in (22) nonpositive.

Combining the properties above, from (31), (33), (35), (37), and (39) the function $\ell \mapsto \big(z(\ell), \hat{\delta}(\ell), \hat{x}(\ell), \hat{u}(\ell)\big)$ defined in (21) satisfies

$$\Psi(z(\ell), \hat{\delta}(\ell), \hat{x}(\ell), \hat{u}(\ell)) \leq 0$$

for all $\ell \in \operatorname{dom}(z, \hat{\delta}, \hat{x}, \hat{u})$, where Ψ is given in (22).

Now, considering (21e) and (21d) result in [1]

$$
\begin{aligned}
\hat{x}(\ell+1) &= (\hat{\delta}_1(\ell) + \hat{\delta}_2(\ell) - \hat{\delta}_1(\ell)\hat{\delta}_2(\ell))\big(f(\hat{x}(\ell), \hat{u}(\ell))\big) \\
&+ (1 - (\hat{\delta}_1(\ell) + \hat{\delta}_2(\ell) - \hat{\delta}_1(\ell)\hat{\delta}_2(\ell)))\big(g(\hat{x}(\ell), \hat{u}(\ell))\big) \\
&= (\hat{\delta}_1 + \hat{\delta}_2 - \hat{\delta}_1\hat{\delta}_2)\big(A_f\hat{x} + B_f\hat{u} + c_f\big) \\
&+ (1 - (\hat{\delta}_1 + \hat{\delta}_2 - \hat{\delta}_1\hat{\delta}_2))\big(A_g\hat{x} + B_g\hat{u} + c_g\big) \\
&= (A_f - A_g)(\hat{\delta}_1 + \hat{\delta}_2 - \hat{\delta}_1\hat{\delta}_2)\hat{x} \\
&+ (B_f - B_g)(\hat{\delta}_1 + \hat{\delta}_2 - \hat{\delta}_1\hat{\delta}_2)\hat{u} \\
&+ (c_f - c_g)\hat{\delta} + A_g\hat{x} + B_g\hat{u} + c_g \\
&\overset{(21e)}{=} (A_f - A_g)z_1 + (B_f - B_g)z_2 \\
&+ (c_f - c_g)(\hat{\delta}_1 + \hat{\delta}_2 + z_3) + A_g\hat{x} + B_g\hat{u} + c_g \\
&= \Phi(z, \hat{\delta}, \hat{x}, \hat{u}).
\end{aligned}
\tag{40}
$$

Therefore, with $(z(\ell), \hat{\delta}(\ell), \hat{u}(\ell))$ given in (21e)-(21d), the function $\hat{x}(\ell) = x(k, j)$ given in (21d) is updated via (23) while making Ψ, given in (22), nonpositive, which corresponds exactly to the definition of a solution to $\mathcal{H}_{MLD}$ given in (6). Hence, the function $\ell \mapsto \big(z(\ell), \hat{\delta}(\ell), \hat{x}(\ell), \hat{u}(\ell)\big)$ defined in (21) is a solution to $\mathcal{H}_{MLD}$. Considering the proof of the first part of the theorem, the proof of the second part follows similarly. □

The system featured in the following example is a slightly modified version of [16, Example 4.1] in a way that the intersection between C and D is nonempty and is adjustable by the parameter σ. We relate a solution of $\mathcal{H}_d$ in (1) into a solution of $\mathcal{H}_{MLD}$ in (4), using Theorem 1. Then, we find the MLD solution using a mixed-integer quadratic programming (MIQP) solver [20].

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

$$
\begin{aligned}
&A_f = \begin{bmatrix} 0.5 & -0.86 \\ 0.86 & 0.5 \end{bmatrix}, \quad B_f = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad A_g = \begin{bmatrix} 0.4 & 0.69 \\ -0.69 & 0.4 \end{bmatrix}, \\
&B_g = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad H_1 = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad H_2 = 0, \quad \sigma = 0.1, \\
&c_f = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, c_g = \begin{bmatrix} 0 \\ 0 \end{bmatrix},
\end{aligned}
\tag{41}
$$

and

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

---

[1] In (40), to ease notation, we ignore writing (ℓ) after the second line.

We substitute the parameters given in (41) in (19) and obtain the parameters of the corresponding MLD system. Then we find a solution to this MLD system using a MIQP solver.[2] As shown in Figs. 1 and 2, the obtained solution starting from $x(0,0) = (-1,-1)$ is a solution to the discretized hybrid system given in (41). 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.

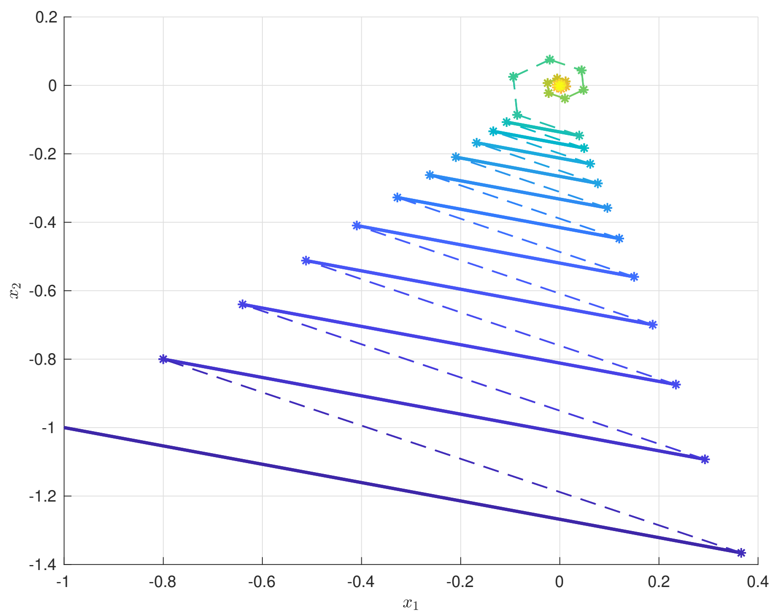

Fig. 1 A trajectory resulting from linear discretized hybrid system that transformed into MLD system

4.2 A MIQP version of the Hybrid Optimal Control Problem

Now we use Theorem 1, and convert the hybrid optimal control problem in Problem 1 to a MIQP problem. To enforce the prediction horizon constraint, we add two

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

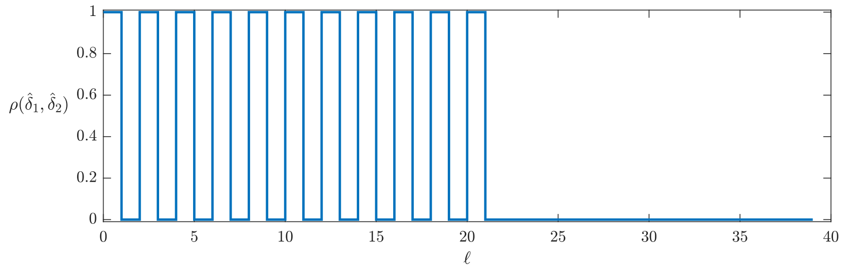

Fig. 2 Binary variable $\rho(\hat{\delta}_1, \hat{\delta}_2)$ obtained from simulation

auxiliary variables $\hat{r}_c$ and $\hat{r}_d$ to the proposed $\mathcal{H}_{MLD}$ system in (4). By adding these two auxiliary variables, 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:

$$\mathcal{H}_{\mathcal{MLD}}: \begin{cases} \hat{\zeta}^+ = \begin{bmatrix} \hat{x}^+ \\ \hat{r}_c^+ \\ \hat{r}_d^+ \end{bmatrix} = \begin{bmatrix} \Phi(z, \hat{\delta}_1, \hat{\delta}_2, \hat{x}, \hat{u}) \\ \rho(\hat{\delta}_1, \hat{\delta}_2) + r_c \\ 1 - \rho(\hat{\delta}_1, \hat{\delta}_2) + r_d \end{bmatrix} \\ \Psi(z, \hat{\delta}_1, \hat{\delta}_2, \hat{x}, \hat{u}) \leq 0, \end{cases} \tag{42}$$

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. Note that ρ is equal to one when $\hat{\delta}_1 = 1$ or $\hat{\delta}_2 = 1$, which corresponding to flow and $\rho = 0$ when $\hat{\delta}_1 = 0$ and $\hat{\delta}_2 = 0$, which corresponding to jump.

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

$$\begin{aligned} \widehat{\mathcal{J}}(z, \hat{\zeta}, \hat{u}) = \sum_{\ell=0}^{N-1} \Bigg(& \Big(z_1(\ell)^\top Q_c \hat{x}(\ell) + z_2(\ell)^\top R_c \hat{u}(\ell) \Big) \\ & + \Big(\hat{x}(\ell)^\top Q_d \hat{x}(\ell) + \hat{u}(\ell)^\top R_d \hat{u}(\ell) \Big) \\ & - \Big(z_1(\ell)^\top Q_d \hat{x}(\ell) + z_2(\ell)^\top R_d \hat{u}(\ell) \Big) \Bigg) \\ & + \hat{x}(N)^\top P \hat{x}(N). \end{aligned} \tag{43}$$

with the data of $\mathcal{H}_{\mathcal{MLD}}$ already defined in (42). So the corresponding MIQP problem to Problem 1 to be solved is as follows.

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

$$\begin{aligned} \text{minimize} \quad & \widehat{\mathcal{J}}(z, \hat{\zeta}, \hat{u}) \\ \text{subject to} \quad & (z, \hat{\delta}_1, \hat{\delta}_2, \hat{\zeta}, \hat{u}) \in \widehat{\mathcal{S}}_{\mathcal{MLH_D}}(z_0) \\ & \hat{x}(N) \in X \\ & (r_c(N), r_d(N)) \in \mathcal{T}, \end{aligned} \tag{44}$$

where N is the terminal time of $(z, \hat{\delta}_1, \hat{\delta}_2, \hat{\zeta}, \hat{u})$ and satisfies $N \in [\tau_p, 2\tau_p]$, 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), a solution to $\mathcal{H}_{\mathcal{MLD}}$ is given by

$$\begin{aligned} \hat{x}(\ell) = \sum_{i=0}^{\ell-1} A^i \big(& B_1 \hat{u}(\ell - 1 - i) \\ & + B_2 \rho(\hat{\delta}_1(\ell - 1 - i), \hat{\delta}_2(\ell - 1 - i)) \\ & + B_3 z(\ell - 1 - i) + B_4 \big) + A^\ell \hat{x}_0, \end{aligned} \tag{45}$$

for each $\ell = k + j$ with $(k, j) \in \text{dom}\,(x, u)$, where A and $\{B_i\}_{i=1}^4$ are given in (19). Substituting (45) in (43) and (22), and defining the vectors

$$\mathcal{Z}(\ell) = \left[z(\ell), \hat{r}_c(\ell), \hat{r}_d(\ell), \hat{u}(\ell), \rho(\hat{\delta}_1(\ell), \hat{\delta}_2(\ell)) \right]^\top \tag{46}$$

$$\mathcal{V} = \begin{bmatrix} \mathcal{Z}(0) \\ \vdots \\ \mathcal{Z}(N-1) \end{bmatrix}, \tag{47}$$

Problem 2 is rewritten as follows.

Problem 3 Given $\hat{x}_0$ and an initial condition $\mathcal{Z}(0) = \mathcal{Z}_0 \in \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^n \times \{0\} \times \{0\} \times \mathbb{R}^m \times \{0, 1\}$ and $\mathcal{V}$ defined in (47)

$$\begin{aligned} \text{minimize} \quad & \mathcal{V}^\top S_1 \mathcal{V} + 2(S_2 + x_0^\top S_3)\mathcal{V} \\ \text{subject to} \quad & F_1 \mathcal{V} \le F_2 + F_3 x_0, \end{aligned} \tag{48}$$

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 $\ell \in \{0, 1, \ldots, N-1\}$ is given in* (46). *Then,*

$$((x(0,0), u(0,0)), \ldots, (x(L,J), u(L,J)))$$

is a solution to Problem 1, where $L + J = N - 1$ and

$$\begin{cases} x(k,j) := \sum_{i=0}^{\ell-1} A^i \big(B_1 \hat{u}(\ell-1-i) \\ \qquad\qquad + B_2 \rho(\hat{\delta}_1(\ell-1-i), \hat{\delta}_2(\ell-1-i)) \\ \qquad\qquad + B_3 z(\ell-1-i) + B_4\big) + A^\ell \hat{x}_0, \\ u(k,j) := \hat{u}(\ell), \end{cases} \tag{49}$$

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* $\ell \in \operatorname{dom}\ (z, \hat{\delta}_1, \hat{\delta}_2, \hat{\zeta}, \hat{u})$.

Note that $\ell = k + j$ with $(k, j) \in \operatorname{dom}\ (x, u)$. When we generate the trajectory via (49), it means the solution either flows or jumps until it reaches the control horizon. This process continues until the solution reaches either τ_c steps of flow or τ_c jumps, i.e., $\max\{k, j\} = \tau_c$, at which point $\ell = k + j$. At every step, we update k and j, and accordingly ℓ is updated. For a visual demonstration of the algorithm, we refer the reader to Figures 2 and 3 of reference [13]. Here, $\tau_c \le \tau_p$ is a positive integer used to parametrize the control horizon.

Proof. The proof follows from the results in the previous sections. Theorem 1 converts Problem 1 to a MIQP problem presented in Problem 3. This shows that to find a solution to Problem 1 for $\mathcal{H}_d$, it is sufficient to solve Problem 3 for the MLD system, $\mathcal{H}_{\mathcal{MLD}}$, in (42) using a MIQP solver. □

4.3 Implementation of Hybrid MPC using MIQP solvers

Using Problem 3, the proposed algorithm for solving hybrid MPC by MIQP solver is given as follows.

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

Set $i = 0$, $\ell_0 = 0$, $\hat{x}(0) = \hat{x}_0$, and $\mathcal{Z}(0) = \mathcal{Z}_0$;
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 (49);
  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 is used to parametrize the control horizon. Control horizon is defined to regulate the optimization times and it has the same structure of prediction horizon $\mathcal{T}$ defined in (7).

In the subsequent sections, we showcase the practical implementation of our proposed method by addressing the hybrid MPC problem in a variety of scenarios. The examples chosen for this demonstration include a discretized version of a controlled bouncing ball system, a congestion control mechanism used in transmission control protocol (TCP) models that exhibit a limit cycle, and the application of MPC to the system described in Example 1. Through these diverse and illustrative cases, we aim to highlight the adaptability and efficacy of our algorithm in managing different types of systems and scenarios.

5 Examples

This section illustrates the proposed method to solve the hybrid MPC problem in examples that feature hybrid control systems with state jumps and logic variables.

Example 2 (Discretized Controlled Bouncing Ball) Consider a ball that is moving vertically and bouncing on a horizontal surface. According to [21, page 27], it is modeled as a point mass with x_1 and x_2 as its height and vertical velocity, respectively. The motion of the ball evolves according to the following discretized hybrid control system:

$$x_1^+ = x_1 + T_s x_2 - T_s^2\delta, \quad x_2^+ = x_2 - T_s\delta \quad \text{when} \quad x_1 \geq 0 \tag{50a}$$
$$x_1^+ = x_1 - T_s x_2, \quad x_2^+ = -\lambda x_2 + u \quad \text{when} \quad x_1 = 0, \quad x_2 \leq 0 \tag{50b}$$

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 $x_1 \geq 0$, 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 $x_2 \leq 0$. 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 control system in (1) for the state and input in a compact set is as follows:

$$A_f = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}, A_g = \begin{bmatrix} 1 & -T_s \\ 0 & -\lambda \end{bmatrix},$$
$$c_f = \begin{bmatrix} -T_s^2\delta \\ -T_s\delta \end{bmatrix}, c_g = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \tag{51}$$
$$B_f = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, B_g = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

$$\mathcal{X} = \{(x, u) | x \in [x_{1_{\min}}, x_{1_{\max}}] \times [x_{2_{\min}}, x_{2_{\max}}], u \in [u_{\min}, u_{\max}]\} \cap L_V(c),$$

$$C = \{(x, u) \in \mathcal{X} : x_1 \geq 0\}, \tag{52}$$

$$D = \{(x, u) \in \mathcal{X} : x_1 = 0, x_2 \leq 0\}, \tag{53}$$

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 given number c. We need to restrict C and D to a compact set that is forward invariant. To represent the flow set and jump set given in (53), in the form that is 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 $i \in \{1, 2\}$ and $x_{1_{\min}} = 0$, $x_{1_{\max}} = 10$, $x_{2_{\min}} = -10$, $x_{2_{\max}} = 10$, $u_{\min} = -0.01$, and $u_{\max} = 0.01$. 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 $f(x, u) = A_f x + B_f u$. Furthermore, 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 $\mathcal{X}$. [3]

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

Example 3 Consider the congestion control mechanism given by the hybrid system [23], which can be modeled by the hybrid equation

$$\begin{cases} \begin{bmatrix} \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} r - B \\ a \end{bmatrix} \quad \text{when} \quad q \in [0, q_{max}] \\ \begin{bmatrix} q^+ \\ r^+ \end{bmatrix} = \begin{bmatrix} q_{max} \\ mr \end{bmatrix} \quad \text{when} \quad q = q_{max}, r \geq B \end{cases} \tag{54}$$

The flow map is discretized by the sample time $T_s = 0.001$ to present (54) as a discretized hybrid control system as (1). So, the data of the discretized hybrid control system in (1) for this example is presented 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 MIQP solver, approach a limit cycle with one jump per period with parameters $B = 1$, $a = 1$, $m = 0.25$, and $q_{max} = 0.25$. The initial

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

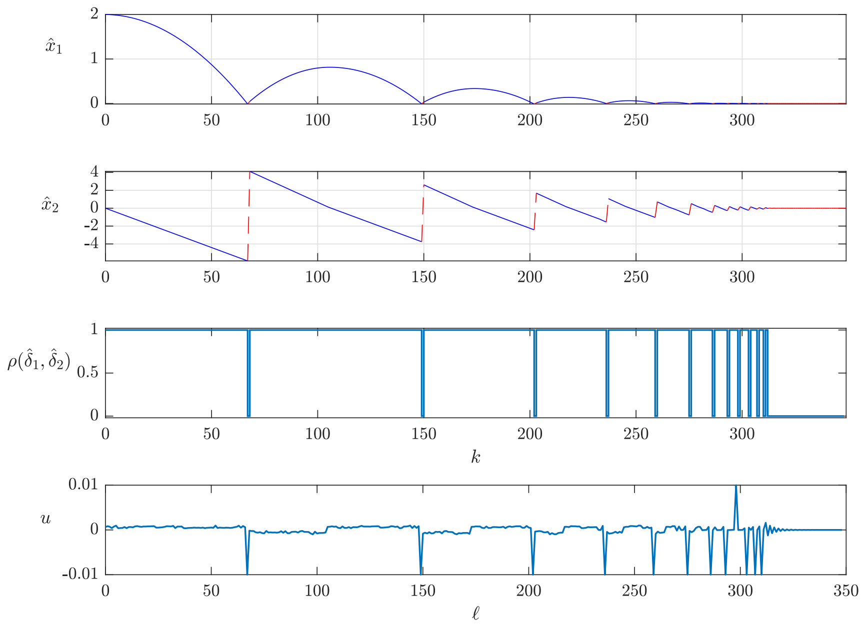

Fig. 3 The solution of the Bouncing ball with input resulting from the linear discretized hybrid system that is transformed into the MLD system.

point 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 $P_2 = (1, 0.4)$. [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 $P = 0.01I_2$. Also, the prediction and control horizon are given with $\tau_p = 2$ and $\tau_c = 1$, respectively.

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 $\mathcal{X}$. Note that "prediction Horizon × number of binary variables" equations should

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

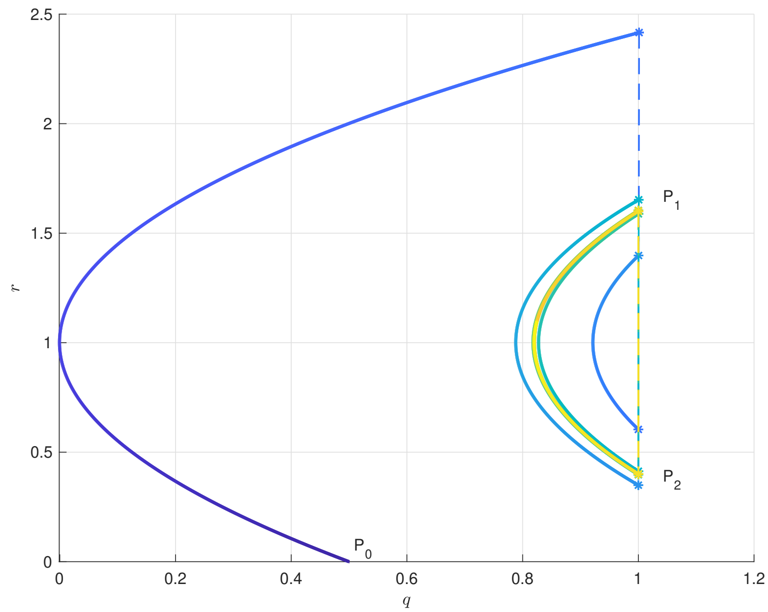

Fig. 4 State Trajectory of TCP described in (54). Illustration of a Limit Cycle in an MIQP Solution. Starting at $P_0 = (0.5, 0)$, the cycle shows a repetitive jump from P_1 to P_2 and back.

be solved at every step. So, adding more binary variables to the system or increasing the prediction horizon adds to the algorithm's complexity. [5]

6 Conclusion

In this work, a new mixed-integer-based approach for the solution of hybrid MPC problems for discretized hybrid control systems is presented. To solve the formulated MPC problem for the discretized hybrid control system, Boolean algebra is employed to formulate a mixed-integer quadratic programming 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. The presented results establish that solving MPC for the discretized hybrid control system, namely $\mathcal{H}_d$ in (1) is equivalent to solving Problem 3 for the MLD system, namely $\mathcal{H}_{\mathcal{MLD}}$ in (42). Future work includes the extension of the proposed method

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

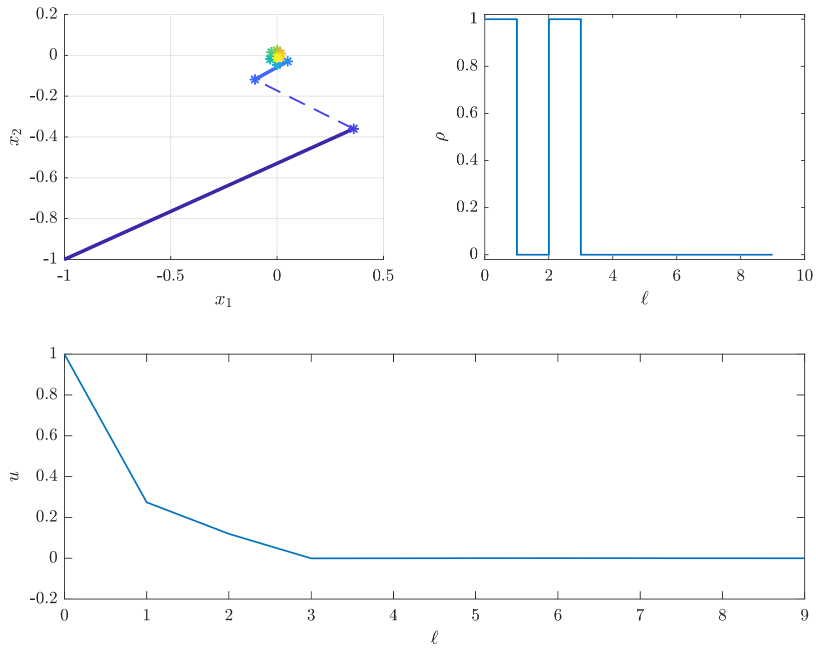

Fig. 5 Model predictive control of system (1) resulting from problem 3

for the case of hybrid equations with 1) uncertainty, 2) nonlinear flow and jump maps, and 3) nonlinear cost functionals.

References

1. James Blake Rawlings, David Q. Mayne, and Moritz Diehl. “Model predictive control: theory, computation, and design.” Vol. 2. Madison, WI: Nob Hill Publishing, 2017.
2. Carlos E. Garcia, David M. Prett, and Manfred Morari. “Model predictive control: Theory and practice—A survey.” Automatica 25.3, 1989, 335-348.
3. Kun Zhang, Jonathan Sprinkle, and Ricardo G. Sanfelice. “Computationally Aware Switching Criteria for Hybrid Model Predictive Control of Cyber.” IEEE Transactions on Automation Science and Engineering, 13.2, 2016.
4. Moritz Diehl, Hans Joachim Ferreau, and Niels Haverbeke. “Efficient numerical methods for nonlinear MPC and moving horizon estimation.” Nonlinear model predictive control: towards new challenging applications, 2009, 391-417.
5. Ricardo G. Sanfelice. “Handbook of Model Predictive Control,” Birkhäuser, 2018, 199-220.
6. Ricardo G. Sanfelice. “Hybrid Feedback Control,” Princeton University Press, 2021.
7. Alberto Bemporad and Domenico Mignone, “miqp.m: a Matlab function for solving mixed integer quadratic programs version 1.02-user guide.” 2000.

8. Francesco Borrelli, Alberto Bemporad, and Manfred Morari. “Predictive control for linear and hybrid systems.” Cambridge University Press, 2017.
9. Rafal Goedel, Ricardo G. Sanfelice, and Andrew R. Teel. “Hybrid dynamical systems: modeling stability, and robustness.” Princeton University Press, 2012.
10. Ricardo G. Sanfelice. “Hybrid feedback control.” Princeton University Press, 2021.
11. Berk Altin and Ricardo G. Sanfelice. “Asymptotically stabilizing model predictive control for hybrid dynamical systems.” In 2019 American Control Conference (ACC), pp. 3630-3635. IEEE, 2019.
12. 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), pp. 1791-1796. IEEE, 2020.
13. Berk Altın, Pegah Ojaghi, and Ricardo G. Sanfelice. “A model predictive control framework for hybrid dynamical systems.” IFAC-PapersOnLine 51.20, 2018, 128-133.
14. 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), pp. 2356-2361. IEEE, 2019.
15. Ricardo G. Sanfelice and Andrew R. Teel. “Dynamical properties of hybrid systems simulators.” Automatica 46.2, 2010, 239-248.
16. Alberto Bemporad and Manfred Morari. “Control of systems integrating logic, dynamics, and constraints.” Automatica 35.3, 1999, 407-427.
17. Iman Nodozi and Ricardo G. Sanfelice. “A Mixed Integer Approach to Solve Hybrid Model Predictive Control Problems.” In 2022 IEEE 61st Conference on Decision and Control (CDC), pp. 1956-1961. IEEE, 2022.
18. Alexandre S. Freire, Eduardo Moreno, and Juan Pablo Vielma. “An integer linear programming approach for bilinear integer programming.” Operations Research Letters 40.2, 2012, 74-77.
19. Akshay Gupte, Shabbir Ahmed, Myun Seok Cheon, and Santanu Dey. “Solving mixed integer bilinear problems using MILP formulations.” SIAM Journal on Optimization 23.2, 2013, 721-744.
20. Alberto Bemporad. “Hybrid Toolbox - User’s Guide,” 2004. http://cse.lab.imtlucca.it/ bemporad/hybrid/toolbox.
21. Alberto Bemporad. “Model predictive control of hybrid systems.” University of Siena, 2007.
22. Jun Chai and Ricardo G. Sanfelice. “On notions and sufficient conditions for forward invariance of sets for hybrid dynamical systems.” 2015 54th IEEE Conference on Decision and Control (CDC). IEEE, 2015.
23. 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 (pp. 291-304). Berlin, Heidelberg: Springer Berlin Heidelberg, 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. With these definitions, we define the following discrete-time control system:

$$\tilde{\mathcal{H}}_d : \begin{cases} \tilde{x}^+ \in \bigcup\limits_{\substack{u_{f1} \in \mathcal{U}_{f1}(\tilde{x},\tilde{u}) \\ u_{f2} \in \mathcal{U}_{f2}(\tilde{x},\tilde{u})}} \left\{\rho(u_{f1},u_{f2})f(\tilde{x},\tilde{u}) + (1-\rho(u_{f1},u_{f2}))g(\tilde{x},\tilde{u})\right\} \\ (\tilde{x},\tilde{u}) \in C \cup D, \end{cases} \tag{55}$$

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 (55) if it satisfies

$$\begin{aligned} \tilde{x}(\ell+1) \in & \bigcup_{\substack{u_{f1}(\ell) \in \mathcal{U}_{f1}(\tilde{x},\tilde{u}) \\ u_{f2}(\ell) \in \mathcal{U}_{f2}(\tilde{x},\tilde{u})}} \big\{\rho(u_{f1}(\ell),u_{f2}(\ell))f(\tilde{x}(\ell),\tilde{u}(\ell)) \\ & + \big(1-\rho(u_{f1}(\ell),u_{f2}(\ell))\big)\, g(\tilde{x}(\ell),\tilde{u}(\ell))\big\} \end{aligned} \tag{56}$$

for all $\ell \in \mathcal{M}$ such that $\ell + 1 \in \mathcal{M}$, where $\mathcal{M}$ is of the form $\{0, 1, \dots, K\}$, 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 (55).

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*

$$\begin{aligned} &\tilde{x}(\ell) := x(k,j), \quad \tilde{u}(\ell) := u(k,j), \\ &u_{f1}(\ell) \in \mathcal{U}_{f1}(x(k,j),u(k,j)), \\ &u_{f2}(\ell) \in \mathcal{U}_{f2}(x(k,j),u(k,j)), \end{aligned} \tag{57}$$

for each $\ell = k + j$ *with* $(k, j) \in \operatorname{dom}(x, u)$*, is a solution to* $\tilde{\mathcal{H}}_d$ *in* (55)*. 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*

$$x(k,j) := \tilde{x}(\ell), \qquad u(k,j) := \tilde{u}(\ell), \tag{58}$$

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

Proof. Pick a solution $(k, j) \mapsto (x(k, j), u(k, j))$ to $\mathcal{H}_d$. Considering (1), the state x is updated via f when $(x, u) \in C \setminus D$, and is updated via g when $(x, u) \in D \setminus C$. In the case where $(x, u) \in C \cap D$, the state x can be updated via either f or g. Now, with $\tilde{u}(\ell) = u(k, j)$, $u_{f1}(\ell) \in \mathcal{U}_{f1}((x(k, j), u(k, j)))$, and $u_{f2}(\ell) \in \mathcal{U}_{f2}((x(k, j), u(k, j)))$, the function $\tilde{x}(\ell) = x(k, j)$ given in (57) is updated as follows:

- via f when $(u_{f1}(\ell) + u_{f2}(\ell) - u_{f1}(\ell)u_{f2}(\ell)) = 1$, i.e., $(\tilde{x}, \tilde{u}) \in C \setminus D$,
- via g when $(u_{f1}(\ell) + u_{f2}(\ell) - u_{f1}(\ell)u_{f2}(\ell)) = 0$, i.e., $(\tilde{x}, \tilde{u}) \in D \setminus C$,
- via either f or g when $(u_{f1}(\ell) + u_{f2}(\ell) - u_{f1}(\ell)u_{f2}(\ell)) \in \{0, 1\}$, i.e., $(\tilde{x}, \tilde{u}) \in C \cap D$.

This corresponds exactly to the definition of a solution to $\tilde{\mathcal{H}}_d$ given in (56). Therefore, the function $\ell \mapsto (\tilde{x}(\ell), \tilde{u}(\ell), u_{f1}(\ell), u_{f2}(\ell))$ defined in (57) is a solution to $\tilde{\mathcal{H}}_d$.

Considering the proof of the first part of the lemma, the proof of the second part is straightforward. □

Appendix A.1 : Converting the hybrid optimal control problem into a nonlinear discrete-time optimal control problem

To enforce the prediction horizon constraint, we add two auxiliary variables r_c and r_d to the proposed $\tilde{\mathcal{H}}_d$ system in (55). By adding these two auxiliary variables we can keep track of flows and the number of jumps elapsed. To this end, we rewrite the nonlinear discrete-time system with new variables as follows:

$$\tilde{\mathcal{H}}_{\mathcal{D}} : \begin{cases} \zeta^+ = \begin{bmatrix} \tilde{x}^+ \\ r_c^+ \\ r_d^+ \end{bmatrix} \in \begin{bmatrix} \bigcup\limits_{\substack{u_{f1} \in \mathcal{U}_{f1}(\tilde{x}, \tilde{u}) \\ u_{f2} \in \mathcal{U}_{f2}(\tilde{x}, \tilde{u})}} \left\{\rho(u_{f1}, u_{f2}) f(\tilde{x}, \tilde{u}) + (1 - \rho(u_{f1}, u_{f2})) g(\tilde{x}, \tilde{u})\right\} \\ \rho(u_{f1}, u_{f2}) + r_c \\ 1 - \rho(u_{f1}, u_{f2}) + r_d \end{bmatrix} \\ (\zeta, \tilde{u}) = (\tilde{x}, r_c, r_d, \tilde{u}) \in \widetilde{C} \cup \widetilde{D}, \end{cases} \tag{59}$$

where $\zeta := (\tilde{x}, r_c, r_d)$, $\widetilde{C} := \{\zeta \in \mathbb{R}^n \times \mathbb{N} \times \mathbb{N} : (\tilde{x}, \tilde{u}) \in C\}$ and $\widetilde{D} := \{\zeta \in \mathbb{R}^n \times \mathbb{N} \times \mathbb{N} : (\tilde{x}, \tilde{u}) \in D\}$.

Considering the quadratic cost functions defining $\mathcal{J}$ in (8), the following cost functional is defined for a given solution $\ell \mapsto (\zeta(\ell), u_{f1}(\ell), u_{f2}(\ell), \tilde{u}(\ell))$ to $\tilde{\mathcal{H}}_{\mathcal{D}}$ with terminal time N

$$
\begin{aligned}
\tilde{\mathcal{J}}(\zeta, u_{f1}, u_{f2}, \tilde{u}) = \sum_{\ell=0}^{N-1} & \rho(u_{f1}(\ell), u_{f2}(\ell))\Big(\tilde{x}(\ell)^\top Q_c \tilde{x}(\ell) + \tilde{u}(\ell)^\top R_c \tilde{u}(\ell)\Big) \\
& + \big(1 - \rho(u_{f1}(\ell), u_{f2}(\ell))\big) \Big(\tilde{x}(\ell)^\top Q_d \tilde{x}(\ell) + \tilde{u}(\ell)^\top R_d \tilde{u}(\ell)\Big) \\
& + \tilde{x}(N)^\top P \tilde{x}(N).
\end{aligned}
\tag{60}
$$

With the data of $\tilde{\mathcal{H}}_{\mathcal{D}}$ already defined in (59), the nonlinear optimal problem to be solved is as follows:

Problem 4 Given an initial condition $\zeta_0 = (\tilde{x}_0, r_{c_0}, r_{d_0}) \in \mathbb{R}^n \times \{0\} \times \{0\}$

$$
\begin{aligned}
\text{minimize} \quad & \widehat{\mathcal{J}}(\zeta, u_{f1}, u_{f2}, \tilde{u}) \\
\text{subject to} \quad & (\zeta, u_{f1}, u_{f2}, \tilde{u}) \in \widehat{\mathcal{S}}_{\mathcal{H}_{\mathcal{D}}}(z_0) \\
& \tilde{x}(N) \in X \\
& (r_c(N), r_d(N)) \in \mathcal{T},
\end{aligned}
\tag{61}
$$

where N is the terminal time of $(\zeta, u_{f1}, u_{f2}, \tilde{u})$ and satisfies $N \in [\tau_p, 2\tau_p]$, and $\widehat{\mathcal{S}}_{\mathcal{H}_{\mathcal{D}}}(\zeta_0)$ is the set of solution pairs of $\tilde{\mathcal{H}}_{\mathcal{D}}$ from ζ_0.