Asymptotically Stabilizing Model Predictive Control for Hybrid Dynamical Systems

Berk Altın and Ricardo G. Sanfelice

***Abstract*— We present a model predictive control (MPC) algorithm for hybrid dynamical systems. The proposed algorithm relies on a terminal constraint and a cost function, as well as a set-based notion of prediction horizon, reminiscent of free end-time optimal control problems. When the terminal cost is a control Lyapunov function (CLF) on the terminal constraint set, and the prediction horizon has a certain geometry, under standard assumptions from conventional MPC, the closed-loop system governed by MPC is shown to have an asymptotically stable compact set using the value function. A numerical example using the prototypical hybrid model of a bouncing ball demonstrates the effectiveness of the proposed algorithm.**

I. INTRODUCTION

In the context of model predictive control (MPC), the term hybrid has been frequently used to refer to systems with continuous- and discrete-valued states [1, Sec. 2.2.5], or discontinuities in the control algorithm or the plant dynamics [1], [2], [3]. Works in the former category often consider discrete-time models derived from the differential equation governing the continuous-valued states, while the discrete-valued states correspond to logical variables [1, Sec. 2.2.5]. Many systems labeled as hybrid in the broader control literature (*hybrid dynamical systems*, or simply hybrid systems) do not follow such a partition, and possess continuously evolving states that are subject to discrete transitions (*jumps*) at times. MPC strategies for such systems have been limited, with the most relevant publications being the impulsive and measure-driven frameworks in [4] and [5]. See [6] for a recent survey.

The objective of this work is to present a stabilizing MPC strategy for hybrid dynamical systems. Similar to existing continuous-time MPC variants [7], the proposed strategy (hereinafter referred to as the *hybrid MPC algorithm*) relies on the solution of an optimal control problem (OCP) at isolated time instants, when state measurements are available. In a receding horizon fashion, the optimal control signal is applied to the system until the next measurement.

Unlike the literature that partitions the state into continuous- and discrete-valued components, our analysis takes place in the mathematical formalism of [8], where a hybrid system is identified by a combination of constrained differential and difference equations (and more generally, inclusions). In addition to a well-developed robust stability theory, as noted in [1], the hybrid systems framework in [8] is highlighted by its simplicity, as well as its ability to describe numerous models of a hybrid nature, such as impulsive [4] and switched [3] systems, and hybrid automata [9]. In conducting the analysis in this setting, we formulate an MPC scheme applicable to various hybrid modeling paradigms, thereby laying the theoretical foundations of a general hybrid MPC framework.

In contrast to some of the existing work, the hybrid MPC algorithm proposed in this paper does not rely on the discretization of the differential equations. Apart from the semantics of the term *hybrid*, which, in our setting, implies the interaction of continuous and discrete dynamics, a major reason for this approach is the fact that discretization is a nontrivial task. Although discretization is somewhat straightforward in certain cases (e.g. sampled-data control systems and switched systems), it is a delicate matter for systems like hybrid automata, in which the state can undergo nonperiodic jumps. For such models, which include mechanical systems with impacts, the time between consecutive jumps can be noninteger multiples of the sampling time, and become arbitrarily small. This issue arises in the prototypical hybrid model of a bouncing ball, modeling the vertical motion of a ball bouncing on a horizontal flat surface. Modeled as a unit point-mass with height x_1 and velocity x_2, the motion of the ball can be represented by the differential equations

$$\dot{x}_1 = x_2 \text{ and } \dot{x}_2 = -\gamma \text{ when } x_1 \geq 0, \tag{1}$$

and the difference equations

$$x_1^+ = x_1 \text{ and } x_2^+ = -\lambda x_2 + u \text{ when } x_1 = 0 \text{ and } x_2 \leq 0. \tag{2}$$

Above, $\gamma > 0$ is the gravitational constant, $\lambda \in [0, 1]$ is the coefficient of restitution, and $u \geq 0$ is an input affecting the post-impact velocity. In the autonomous case with dissipative impacts (in other words, when $u = 0$ and $\lambda < 1$), state trajectories of the bouncing ball are characterized by Zeno phenomenon, where the time between consecutive jumps tends toward zero [8, Example 2.12].

As summarized in Section III, the OCP associated with our proposed algorithm minimizes a cost functional weighting the state during both the continuous and discrete phases, and imposes constraints on the terminal state and time. The assumptions on the OCP listed in Section IV are similar to their counterparts in the continuous/discrete-time MPC literature. However, the notion of terminal time differs significantly from conventional finite-horizon optimal control. Akin to free end-time problems, the proposed notion allows

*This research has been partially supported by the National Science Foundation under CAREER Grant no. ECS-1450484, Grant no. ECS-1710621, and Grant no. CNS-1544396, by the Air Force Office of Scientific Research under Grant no. FA9550-16-1-0015, by the Air Force Research Laboratory under Grant no. FA9453-16-1-0053, and by CITRIS and the Banatao Institute at the University of California.

Berk Altın and Ricardo G. Sanfelice are with the Department of Electrical and Computer Engineering, University of California, Santa Cruz, CA 95064, USA; berkaltin@ucsc.edu, ricardo@ucsc.edu.

for state trajectories with terminal times belonging to a set, called the *prediction horizon*. To account for hybrid time domains, which are introduced in Section II, a hybrid time domain-like geometry is assumed for the prediction horizon.

Section V is devoted to the relevant properties of the OCP. Recursive feasibility of the proposed hybrid MPC algorithm is shown by exploiting the geometry of the prediction horizon. Then, using the assumptions in Section IV, which require the terminal cost to be a control Lyapunov function (CLF) on the terminal constraint set with respect to a set $\mathcal{A}$, it is proven that the value function is continuous on $\mathcal{A}$ and positive definite with respect to it. Since the value function is upper bounded by a decreasing function along optimal solution pairs, asymptotic stability of $\mathcal{A}$ can be established, as detailed in Section VI. A numerical example using the bouncing ball system is presented in Section VII to showcase the algorithm. Due to space constraints, proofs of the technical results are omitted and will be published in another venue. An outline of this work without formal statements and more restrictive assumptions can be found in [10]. Note that as opposed to the case study reported in [10], where the OCP is solved analytically, the bouncing ball example reported here relies on *numerical solutions* of the OCP, demonstrating the applicability of the algorithm to a broader class of problems.

II. Preliminaries

We use $\mathbb{R}$ to represent real numbers and $\mathbb{R}_{\geq 0}$ its nonnegative subset. The set of nonnegative integers is denoted $\mathbb{N}$. The notation $S_1 \subset S_2$ indicates S_1 is a subset of S_2, not necessarily proper. The Euclidean norm is denoted $|.|$. The distance of a vector $x \in \mathbb{R}^n$ to a nonempty set $\mathcal{A} \subset \mathbb{R}^n$ is $|x|_{\mathcal{A}} := \inf_{a\in\mathcal{A}} |x - a|$. The interior of a set $S \subset \mathbb{R}^n$ is denoted $\operatorname{int} S$. We denote by $\pi : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ the standard projection onto $\mathbb{R}^n$ such that $\pi(x, y) = x$. A strictly increasing continuous function $\alpha : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is said to belong to class-$\mathcal{K}_\infty$ if $\alpha(0) = 0$.

A. Hybrid Control Systems

In this paper, a hybrid control system $\mathcal{H}$ is defined by its data (C, f, D, g), and represented as follows:

$$\mathcal{H} \begin{cases} \dot{x} = f(x,u) & (x,u) \in C \\ x^+ = g(x,u) & (x,u) \in D. \end{cases} \tag{3}$$

Above, $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$ denote the state and input of $\mathcal{H}$, respectively. The *flow map* $f : C \to \mathbb{R}^n$ describes the continuous evolution (*flow*) of the state x when $(x, u) \in C$. The *flow set* $C \subset \mathbb{R}^n \times \mathbb{R}^m$ is a mixed state-input constraint that defines where flows are allowed. In a similar manner, the *jump map* $g : D \to \mathbb{R}^n$ describes the discrete evolution (*jump*) of the state x when $(x, u) \in D$, with the *jump set* $D \subset \mathbb{R}^n \times \mathbb{R}^m$ defining where jumps are allowed.

Assumption 2.1: The sets C and D are closed. The functions f and g are continuous.

Example 2.2 (Bouncing Ball): Consider the bouncing ball with actuated jumps evolving according to (1) and (2). The dynamics of the bouncing ball in Section I can be represented in the form of (3) with state $x = (x_1, x_2)$ and input u by incorporating the constraints therein. The flow map is given as $f(x, u) = (x_2, -\gamma)$ on the flow set $C = C' \times U_C$, where $C' = \{x \in \mathbb{R}^2 : x_1 \geq 0\}$ and the nonempty compact set $U_C \subset \mathbb{R}^m$ is arbitrary, as f does not depend on u. The jump map is given as $g(x, u) = (0, -\lambda x_2 + u)$ on the jump set $D = D' \times \mathbb{R}_{\geq 0}$, where $D' = \{x \in \mathbb{R}^2 : x_1 = 0, x_2 \leq 0\}$.

Solution pairs of $\mathcal{H}$ are defined on *hybrid time domains* and parametrized by two independent variables, $t \in \mathbb{R}_{\geq 0}$ and $j \in \mathbb{N}$. A hybrid time domain E is a subset of $\mathbb{R}_{\geq 0} \times \mathbb{N}$ with the property that for each $(T, J) \in E$, there exists a finite nondecreasing sequence $\{t_j\}_{j=0}^{J+1}$ such that $t_0 = 0$ and

$$E \cap ([0, T] \times \{0, 1, \ldots, J\}) = \bigcup\nolimits_{j=0}^{J} ([t_j, t_{j+1}] \times \{j\}).$$

Given a solution pair with domain E, at any $(t, j) \in E$, t denotes the ordinary time elapsed and j denotes the number of jumps that have occurred. The subsequence $\{t_j\}_{j=1}^{J}$ corresponds to when jumps occur, as formalized next.

Definition 2.3: Given a pair of functions $x : \operatorname{dom} x \to \mathbb{R}^n$ and $u : \operatorname{dom} u \to \mathbb{R}^m$ representing the state trajectory and the input, respectively, (x, u) is said to be a solution pair of (3) if $\operatorname{dom}(x, u) := \operatorname{dom} x = \operatorname{dom} u$ is a hybrid time domain, $(x(0, 0), u(0, 0)) \in \operatorname{cl}(C) \cup D$, where cl denotes closure, and the following hold:

- For all $j \in \mathbb{N}$ such that $I^j := \{t : (t, j) \in \operatorname{dom}(x, u)\}$ has a nonempty interior, 1) the function $t \mapsto x(t, j)$ is locally absolutely continuous, 2) $(x(t, j), u(t, j)) \in C$ for all $t \in \operatorname{int} I^j$, 3) the function $t \mapsto u(t, j)$ is Lebesgue measurable and locally essentially bounded, and 4) for almost all $t \in I^j$,
$$\dot{x}(t, j) = f(x(t, j), u(t, j)). \tag{4}$$
- For all $(t, j) \in \operatorname{dom}(x, u)$ such that $(t, j + 1) \in \operatorname{dom}(x, u)$,
$$\begin{aligned} &(x(t, j), u(t, j)) \in D, \\ &\qquad x(t, j + 1) = g(x(t, j), u(t, j)). \end{aligned} \tag{5}$$

We denote by $\widehat{\mathcal{S}}_{\mathcal{H}}(S)$ the set of solution pairs of $\mathcal{H}$ originating from a set $S \subset \mathbb{R}^n$, with $\widehat{\mathcal{S}}_{\mathcal{H}} := \widehat{\mathcal{S}}_{\mathcal{H}}(\mathbb{R}^n)$.[1] That is, given any $(x, u) \in \widehat{\mathcal{S}}_{\mathcal{H}}(S)$, $x(0, 0) \in S$. Given a solution pair (x, u), $(T, J) \in \operatorname{dom}(x, u)$ is said to be the terminal (hybrid) time of (x, u) if $T \geq t$ and $J \geq j$ for all $(t, j) \in \operatorname{dom}(x, u)$. The pair (x, u) is said to be complete if $\operatorname{dom}(x, u)$ is unbounded.

In contrast with autonomous hybrid systems, which require further conditions for uniqueness of state trajectories [8, Proposition 2.11], the following assumption is necessary and sufficient for uniqueness, as the jump times are determined by the domain of the input.

Assumption 2.4: Given (classical) solution pairs $t \mapsto (x(t), u(t))$ and $t \mapsto (x'(t), u'(t))$ to the constrained differential equation $\dot{x} = f(x, u)$, $(x, u) \in C$, if $u(t) = u'(t)$ almost everywhere and $x(0) = x'(0)$, then $x = x'$.

[1] We use $\widehat{\mathcal{S}}_{\mathcal{H}}(S)$ to avoid confusion with the notation in [8] for sets of *maximal* state trajectories [8, Definition 2.7] of autonomous hybrid systems.

B. Hybrid Control Systems under Static State Feedback

Given the *feedback pair* $\kappa := (\kappa_C, \kappa_D)$, defined by the functions $\kappa_C : \mathbb{R}^n \to \mathbb{R}^m$ and $\kappa_D : \mathbb{R}^n \to \mathbb{R}^m$, let

$$\mathcal{H}_\kappa \begin{cases} \dot{x} = f_\kappa(x) := f(x, \kappa_C(x)) & x \in C_\kappa \\ x^+ = g_\kappa(x) := g(x, \kappa_D(x)) & x \in D_\kappa, \end{cases} \tag{6}$$

where

$$C_\kappa := \{x \in \mathbb{R}^n : (x, \kappa_C(x)) \in C\},$$
$$D_\kappa := \{x \in \mathbb{R}^n : (x, \kappa_D(x)) \in D\},$$

and $\mathcal{H}_\kappa$ arises from the application of κ on $\mathcal{H}$.

Trajectories of (6) are defined over hybrid time domains via Definition 2.3 as follows: a function x is a state trajectory of (6) if there exists a solution pair (x, u) (said to be *generated by* κ) that satisfies (4) with $u(t, j) = \kappa_C(x(t, j))$ for all $t \in \operatorname{int} I^j$ and (5) with $u(t, j) = \kappa_D(x(t, j))$.

III. Hybrid MPC

As in conventional continuous/discrete-time MPC, the algorithm proposed in this paper is implemented by measuring the state of the plant $\mathcal{H}$ in (3) and minimizing a cost functional. Each time the state is measured, the algorithm finds an optimal control that is applied to $\mathcal{H}$ until the next measurement, leading to a moving horizon implementation. Similar to free end-time optimal control, the minimization is performed over a nontrivial prediction horizon, in that the terminal time is allowed to vary within a set.

The set-based generalization of the notion of prediction horizon accounts for hybrid time domains and maximizes the set of initial conditions such that the OCP is feasible. Indeed, for the bouncing ball model in Example 2.2, it could be impossible to stabilize the origin by hybrid MPC when the associated OCP is restricted to a fixed prediction horizon of the form (T, J) for some $T > 0$ and $J \geq 0$: since flows are not possible from the origin, given any solution pair (x, u) with terminal time (T, J), there exists $(t, j) \in \operatorname{dom}(x, u)$ such that $x(t, j) \neq (0, 0)$. Hence, the algorithm would not be able to make the origin forward invariant. On the other hand, if the prediction horizon were to be fixed at $(0, J)$ for some $J > 0$, the OCP would only allow jumps and therefore be unsolvable from any initial condition with positive position or velocity, which, again, would prohibit stabilization.

Another subtlety arises in the moving horizon implementation of the algorithm, where a periodic measurement and optimization scheme may not be meaningful. This issue can be observed with the aforementioned scenario for the bouncing ball, where the time between consecutive jumps converges to zero for any state trajectory converging to the origin, and suggests the usage of a similar set-based notion of *control horizon* to regulate the optimization times [10]. An example implementation using the control horizon approach is shown at the end of this section. Since this approach can result in the optimization times being nonperiodic, the proposed hybrid MPC algorithm allows for the online selection of each optimization time, as discussed in Section VI.

A. Finite-Horizon Hybrid Optimal Control

We now detail the formulation of the OCP associated with the hybrid MPC algorithm. Since the flow set C and jump set D define mixed state-input constraints on solution pairs of $\mathcal{H}$, no additional state-input constraints will be given for the problem. Instead, constraints on the terminal time and state will be specified by

- the *prediction horizon* $\mathcal{T} \subset \mathbb{R}_{\geq 0} \times \mathbb{N}$, and
- the *terminal constraint set* $X \subset \pi(C \cup D)$.

1) The Cost Functional: Given a solution pair (x, u) of $\mathcal{H}$ with terminal time (T, J), let $\{t_j\}_{j=0}^{J+1}$ be the sequence such that $\operatorname{dom}(x, u) = \cup_{j=0}^{J}([t_j, t_{j+1}] \times \{j\})$, where $t_{J+1} = T$. We define the cost functional $\mathcal{J}$ such that

$$\mathcal{J}(x, u) := \left(\sum_{j=0}^{J} \int_{t_j}^{t_{j+1}} L_C(x(t, j), u(t, j))\, dt \right) + \left(\sum_{j=0}^{J-1} L_D(x(t_{j+1}, j), u(t_{j+1}, j)) \right) + V(x(T, J)),$$

where $L_C : C \to \mathbb{R}_{\geq 0}$ is the *flow cost*, $L_D : D \to \mathbb{R}_{\geq 0}$ is the *jump cost*, and $V : X \to \mathbb{R}_{\geq 0}$ is the *terminal cost*. The second sum is to be interpreted as zero if $J = 0$.

2) The Prediction Horizon: As discussed before, the notion of a prediction horizon given by a singleton can be overly restrictive and prevent a reasonable formulation of MPC for hybrid systems in the form (3). To maximize feasibility of the OCP, an appropriate selection of $\mathcal{T}$ should ensure that it intersects with unbounded hybrid time domains. A natural way of handling this is to let

$$\mathcal{T} := \{(T, J) \in \mathbb{R}_{\geq 0} \times \mathbb{N} : \max\{T/\delta_p, J\} = \tau_p\} \tag{7}$$

for some $\delta_p > 0$ and $\tau_p \in \{1, 2, \dots\}$. Above, the parameters τ_p and δ_p define a "rectangle" of height τ_p and width $\tau_p \delta_p$ in the (t, j) plane. Observing that $\mathcal{T}$ "connects" the two axes of $\mathbb{R}_{\geq 0} \times \mathbb{N}$ under this choice, we generalize its structure as follows.

Assumption 3.1: There exists a finite nonincreasing sequence $\{t_j\}_{j=0}^{J+1}$ such that $t_0 > 0$, $t_{J+1} = 0$, and

$$\mathcal{T} := \bigcup\nolimits_{j=0}^{J}([t_{j+1}, t_j] \times \{j\}).$$

Assumption 3.1 imposes a specific structure on $\mathcal{T}$, with its geometry resembling a "reverse" hybrid time domain. When $\mathcal{T}$ is selected according to (7), Assumption 3.1 holds with $J = \tau_p$ and $t_0 = t_1 = \cdots = t_J = \delta_p \tau_p$. This assumption is used to prove recursive feasibility.

3) The Constrained OCP: With the constraints X and $\mathcal{T}$ already defined, the OCP to be solved is presented next.

Problem 3.2: Given an initial condition $x_0 \in \mathbb{R}^n$,

$$\begin{aligned} \underset{(x,u) \in \widehat{\mathcal{S}}_{\mathcal{H}}(x_0)}{\text{minimize}} \quad & \mathcal{J}(x, u) \\ \text{subject to} \quad & (T, J) \in \mathcal{T} \\ & x(T, J) \in X, \end{aligned} \tag{8}$$

where (T, J) denotes the terminal time of (x, u).

A solution pair (x, u) is said to be *feasible* if it satisfies the constraints of (8) for some x_0. The feasible set $\mathcal{X}$ is the set of all x_0 so that there exists a feasible $(x, u) \in \widehat{\mathcal{S}}_{\mathcal{H}}(x_0)$. The *value function* $\mathcal{J}^* : \mathcal{X} \to \mathbb{R}_{\geq 0}$ is defined as

$$\mathcal{J}^*(x_0) := \inf_{\substack{(x,u)\in\widehat{\mathcal{S}}_{\mathcal{H}}(x_0)\\ (T,J)\in\mathcal{T}\\ x(T,J)\in X}} \mathcal{J}(x, u) \quad \forall x_0 \in \mathcal{X}, \tag{9}$$

where (T, J) is the terminal time of (x, u). A feasible (x, u) attaining the infimum is said to be *optimal*.

While the existence of optimal solution pairs is addressed in Section IV, their computation is a nontrivial task. In certain cases, Problem 3.2 can be solved by converting it into a finite-dimensional nonlinear program, as we shall see in Section VII. The development of general numerical methods to solve Problem 3.2 is the current object of research, and can take the form of the maximum principle—see [11] and the references therein.

B. An Example Implementation of Hybrid MPC

Having formulated the OCP in Problem 3.2 we present an example implementation of the proposed algorithm. For this implementation, we suppose

$$\mathcal{T} = \{(T, J) \in \mathbb{R}_{\geq 0} \times \mathbb{N} : T + J \in [\tau, \tau + 1]\} \tag{10}$$

for some constant $\tau > 0$, which satisfies Assumption 3.1.

Algorithm 1 Hybrid MPC Implementation

```
i = 0, (T_0, J_0) = (0, 0), x_0 = x(0, 0).
while true do
    Solve Problem 3.2 to obtain an optimal pair (x_i^*, u_i^*).
    while (t − T_i) < τ/2 and j − J_i < 1 do
        Apply u_i^* to H to generate the state trajectory x.
    end while
    i = i + 1, (T_i, J_i) = (t, j), x_0 = x(T_i, J_i).
end while
```

In the implementation in Algorithm 1, the state trajectory x results from the application of a sequence of optimal control inputs $\{u_i^*\}_{i=0}^{\infty}$ to $\mathcal{H}$ from x_0. The initial optimization occurs at time $(0, 0)$, and the initial optimal control input u_0^* is applied until $\tau/2$ units of ordinary time elapse or a jump occurs, whichever occurs first. At this point, Problem 3.2 is re-solved to find the new input u_1^*, which is again applied for $\tau/2$ units of ordinary time or until the next jump. The portion of the state trajectory x from (T_i, J_i) to (T_{i+1}, J_{i+1}) corresponds to the optimal state trajectory x_i^* computed at time $(T_i, J_i) \in \mathrm{dom}(x, u)$. Note that each (x_i^*, u_i^*) flows for at least τ units of ordinary time, or jumps at least once, due to (10). This implementation is generalized in Section VI.

IV. Basic MPC Assumptions for Hybrid Systems

Now we list the assumptions imposed on Problem 3.2 to ensure feasibility and stability properties. The first assumption concerns the existence of optimal controls. Sufficient conditions for this assumption can be found in [12], where standard conditions for existence in continuous time are used.

Assumption 4.1: For any $x_0 \in \mathcal{X}$, an optimal solution pair $(x, u) \in \widehat{\mathcal{S}}_{\mathcal{H}}(x_0)$ exists.

Because of the existence of state variables that do not necessarily converge to equilibria (e.g., timers and logic variables), stability theory for hybrid systems consider sets as opposed to singletons [8, Ch. 3]. As such, the conditions of Assumption 4.2 are stated with respect to a set $\mathcal{A} \subset X$.

Assumption 4.2: Given the compact set $\mathcal{A} \subset X$, the following hold:

- (O1) There exists a class-$\mathcal{K}_\infty$ function α_C such that for every $(x, u) \in C$, $L_C(x, u) \geq \alpha_C(|x|_{\mathcal{A}})$.
- (O2) There exists a class-$\mathcal{K}_\infty$ function α_D such that for every $(x, u) \in D$, $L_D(x, u) \geq \alpha_D(|x|_{\mathcal{A}})$.
- (O3) $V(x) = 0$ if and only if $x \in \mathcal{A}$.
- (O4) The inclusion $S \cap (\pi(C \cup D)) \subset X$ holds for some open set $S \supset \mathcal{A}$.
- (O5) There exists a continuous function $\sigma : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ such that $|f(x, u)| \leq \sigma(|x|_{\mathcal{A}})$ for all $(x, u) \in C$.

The positive definite properties imposed in Conditions (O1)-(O3) mirror those encountered in the MPC literature. On the other hand, Condition (O4) is weaker than the typical assumption in the MPC literature, which would translate to our setting by requiring $\mathcal{A}$ to be contained in the interior of X, a restrictive assumption when $\pi(C \cup D) \neq \mathbb{R}^n$. This property is utilized to establish continuity of the value function on $\mathcal{A}$. The last condition of Assumption 4.2, (O5), implies the existence of a uniform upper bound on the magnitude of the velocity vector $\dot{x} = f(x, u)$. It is satisfied when $C = C' \times U$ for a closed $C' \subset \mathbb{R}^n$ and compact $U \subset \mathbb{R}^m$, which corresponds to the typical continuous-time MPC assumption on state and input constraints.

The final assumption concerns the feedback κ and the terminal cost V over the terminal constraint set X. For the term *forward invariant* used here, see [13, Definition 3.1].

Assumption 4.3: The terminal constraint set X is closed, and forward invariant for the hybrid system $\mathcal{H}_\kappa$ in (6). Moreover, 1) the flow cost L_C, jump cost L_D, terminal cost V, and feedback pair κ are continuous, and 2) there exists an open set $S \supset (X \cap C_\kappa)$ on which the terminal cost V is differentiable and the following hold:

$$\begin{aligned} \langle \nabla V(x), f_\kappa(x)\rangle &\leq -L_C(x, \kappa_C(x)) \quad \forall x \in X \cap C_\kappa, \\ V(g_\kappa(x)) - V(x) &\leq -L_D(x, \kappa_D(x)) \quad \forall x \in X \cap D_\kappa, \end{aligned} \tag{11}$$

$$\begin{aligned} L_C(x, \kappa_C(x)) &= 0 \quad \forall x \in \mathcal{A} \cap C_\kappa, \\ L_D(x, \kappa_D(x)) &= 0 \quad \forall x \in \mathcal{A} \cap D_\kappa. \end{aligned} \tag{12}$$

Assumption 4.3 is an extension of the familiar CLF-like assumption in the MPC literature (see, e.g., [7]) and is the main stabilizing ingredient of the hybrid MPC algorithm. As per conventional continuous/discrete-time MPC, solutions of the feedback-controlled system $\mathcal{H}_\kappa$ are used to establish desired properties of the OCP. Note that forward invariance of X for $\mathcal{H}_\kappa$ necessitates $X \subset C_\kappa \cup D_\kappa$, and, as a result of Assumption 3.1, $X \subset \mathcal{X}$.

V. Properties of the OCP

This section presents the relevant properties of Problem 3.2, exploited to show asymptotic stability of the compact set $\mathcal{A}$ via the proposed hybrid MPC algorithm.

We begin the section with a couple of feasibility properties. The first result shows that when the set $\mathcal{A}$ is in the relative interior of X, Assumption 4.3 ensures that there exists a relative neighborhood of $\mathcal{A}$, where feasible solution pairs exist everywhere.

Proposition 5.1: Suppose Assumptions 2.1, 3.1 and 4.3 hold. Then, $X \subset \mathcal{X}$. If, in addition, Condition (O4) of Assumption 4.2 holds, there exists $\delta > 0$ such that $x \in X$ for every $x \in \pi(C \cup D)$ satisfying $|x|_{\mathcal{A}} \leq \delta$.

The next result extends the typical forward/recursive feasibility (see [7]) result in continuous/discrete-time MPC to the hybrid case. The proof relies on extending feasible solution pairs from the terminal constraint set via the CLF property. That is, as in conventional MPC, feasible solution pairs can be extended by concatenation. To facilitate the discussion, we formally define what we mean by concatenation.

Definition 5.2: A solution pair (x, u) is said to be the concatenation of solution pairs (x_0, u_0) and (x_1, u_1) with terminal times (T_0, J_0) and (T_1, J_1), respectively, if

$$\operatorname{dom}(x, u) = \operatorname{dom}(x_0, u_0) \cup \{(t + T_0, j + J_0) : (t, j) \in \operatorname{dom}(x_1, u_1)\},$$

and for every $(t, j) \in \operatorname{dom}(x, u)$, the following holds:

$$(x(t, j), u(t, j)) = (x_0(t, j), u_0(t, j)),$$

if $t + j < T_0 + J_0$, otherwise,

$$(x(t, j), u(t, j)) = (x_1(t - T_0, j - J_0), u_1(t - T_0, j - J_0)).$$

Proposition 5.3: Suppose Assumptions 2.1, 3.1 and 4.3 hold. Let (x, u) be a feasible solution pair. Then, for any $(t, j) \in \operatorname{dom}(x, u)$, $x(t, j)$ belongs to the feasible set $\mathcal{X}$, i.e., $x(t, j) \in \mathcal{X}$.

Next, we establish $\mathcal{J}^*$ as a candidate Lyapunov function.

Lemma 5.4: Under Assumptions 2.1, 3.1, 4.2, and 4.3, the value function $\mathcal{J}^*$ in (9) satisfies the following:

1) *For all $\varepsilon > 0$, there exists $\delta > 0$ such that $\mathcal{J}^*(x) \leq \varepsilon$ for every $x \in \mathcal{X}$ with $|x|_{\mathcal{A}} \leq \delta$.*
2) *There exists a continuous function $\alpha : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ such that $\mathcal{J}^*(x) \geq \alpha(|x|_{\mathcal{A}})$ for every $x \in \mathcal{X}$, $\alpha(r) = 0$ if and only if $r = 0$, and $\liminf_{r \to \infty} \alpha(r) > 0$.*

Finally, using the CLF-like conditions and analyzing the feasible solution obtained by concatenation in Proposition 5.3, it can be shown that the value function is upper bounded by a decreasing function along optimal solutions.

Lemma 5.5: Suppose Assumptions 2.1, 3.1 and 4.3, and Conditions (O1)-(O2) of Assumption 4.2 hold. Let (x, u) be an optimal solution pair. Then, for any $(t, j) \in \operatorname{dom}(x, u)$,

$$\mathcal{J}^*(x(t, j)) \leq \mathcal{J}^*(x_0) - \left(\sum_{i=0}^{j} \int_{s_i}^{s_{i+1}} \alpha_C(|x(s, i)|_{\mathcal{A}})\, ds + \sum_{i=0}^{j-1} \alpha_D(|x(s_{i+1}, i)|_{\mathcal{A}}) \right),$$

where $\{s_i\}_{i=0}^{j+1}$ is the sequence satisfying

$$\operatorname{dom}(x, u) \cap ([0, t] \times \{0, 1, \ldots, j\}) = \cup_{i=0}^{j} ([s_i, s_{i+1}] \times \{i\}).$$

VI. Hybrid MPC: Open-loop Solutions and Asymptotic Stability

This section details asymptotic stability of the hybrid MPC algorithm. Recalling that a periodic implementation of hybrid MPC is not always possible, we introduce a notion of solution pairs for $\mathcal{H}$ under open-loop optimal control signals derived from Problem 3.2, generalizing Algorithm 1. Below, the concatenation of $\{(\tilde{x}_i, \tilde{u}_i)\}_{i=0}^{\infty}$ is to be understood in the sense of Definition 5.2, and the truncation of (x_i, u_i) is its restriction to $([0, t] \times \{0, 1 \ldots, j\}) \cap \operatorname{dom}(x_i, u_i)$ for some $(t, j) \in \operatorname{dom}(x_i, u_i)$.

Definition 6.1: A complete solution pair (x, u) is said to be generated by the hybrid MPC algorithm if it is the concatenation of a sequence of solution pairs $\{(\tilde{x}_i, \tilde{u}_i)\}_{i=0}^{\infty}$, where for each $i \in \mathbb{N}$, $(\tilde{x}_i, \tilde{u}_i)$ is the truncation of an optimal solution pair (x_i, u_i).

In establishing stability properties for hybrid MPC, the stability notions in [8, Definition 7.1] are adapted for solution pairs described in Definition 6.1. Note that by definition, every (x, u) generated by the hybrid MPC algorithm is complete and satisfies $x(0, 0) \in \mathcal{X}$.

Definition 6.2: The hybrid MPC algorithm is said to render the set $\mathcal{A}$ asymptotically stable if the following hold:

- For all $\varepsilon > 0$, there exists $\delta > 0$ such that given any solution pair (x, u) generated by the hybrid MPC algorithm, $|x(0, 0)|_{\mathcal{A}} \leq \delta$ implies $|x(t, j)|_{\mathcal{A}} \leq \varepsilon$ for all $(t, j) \in \operatorname{dom}(x, u)$.
- There exists $\mu > 0$ such that given any solution pair (x, u) generated by the hybrid MPC algorithm, $|x(0, 0)|_{\mathcal{A}} \leq \mu$ implies $\lim_{t+j \to \infty} |x(t, j)|_{\mathcal{A}} = 0$.

Definition 6.2 is meaningful under Assumptions 2.4 and 4.1, and the conditions of Propositions 5.1 and 5.3. Indeed, there exists a neighborhood of $\mathcal{A}$ so that from any x_0 in this neighborhood, either no solution pairs exist or an optimal $(x, u) \in \widehat{\mathcal{S}}_{\mathcal{H}}(x_0)$ can be found. In addition, Proposition 5.3 ensures that solutions generated by the hybrid MPC algorithm stay in $\mathcal{X}$. Finally, Assumption 2.4 guarantees that given any $(x, u) \in \widehat{\mathcal{S}}_{\mathcal{H}}(x_0)$, the application of the open-loop input u to $\mathcal{H}$ with initial condition x_0 results in the state trajectory x as desired. This allows us to use the value function $\mathcal{J}^*$, which is characterized as a Lyapunov function by Lemmas 5.4-5.5, to show asymptotic stability.

Theorem 6.3: Suppose Assumptions 2.1, 2.4, 3.1 and 4.1-4.3 hold. Then, the hybrid MPC algorithm renders the compact set $\mathcal{A}$ asymptotically stable for the hybrid system $\mathcal{H}$. Furthermore, $\lim_{t+j \to \infty} |x(t, j)|_{\mathcal{A}} = 0$ for every solution pair (x, u) generated by the hybrid MPC algorithm.

VII. Numerical Example

This section demonstrates an implementation of the hybrid MPC algorithm with the bouncing ball in Example 2.1.

Consider the total energy $W(x) := \gamma x_1 + x_2^2/2$ of the ball for all $x \in C'$, and let $\mathcal{A} = \{x \in C' : W(x) = \gamma h\}$ for some given constant $h \geq 0$, which trivially satisfies Condition (O5). When $\lambda = 1$, $\mathcal{A}$ corresponds to the limit cycle of the autonomous bouncing ball originating from $(h, 0)$.

Let $X = C'$ and fix $\theta \in (0, (2/\pi)(1-\lambda^4)/(1+\lambda^4))$. Let

$$V(x) = (1 + \theta \arctan x_2)(W(x) - \gamma h)^2 \quad \forall x \in X,$$

which satisfies Condition (O3), due to the assumption on θ. Furthermore, Condition (O4) holds with $S = \mathbb{R}^2$. For the closed-loop system $\mathcal{H}_\kappa$ in (6), we choose κ_C as an arbitrary function with its range in U_C, which leads to the set $C_\kappa = C'$ and mapping $f_\kappa(x) = (x_2, -\gamma)$, and κ_D such that

$$\kappa_D(x) = \max\{\lambda x_2 + \sqrt{2\gamma h}, 0\} \quad \forall x \in \mathbb{R}^2,$$

which leads to the set $D_\kappa = D'$ and mapping

$$g_\kappa(x) = \begin{cases} (0, -\lambda x_2) & \text{if } x_2 \leq -\sqrt{2\gamma h}/\lambda \\ (0, \sqrt{2\gamma h}) & \text{otherwise.} \end{cases}$$

We select the flow cost so that

$$L_C(x, u) = \theta\gamma(W(x) - \gamma h)^2/(1 + 2W(x)) \quad \forall (x, u) \in C,$$

which satisfies Condition (O1) due to *radial unboundedness* of W in C'. The jump cost is chosen so that

$$L_D(x, u) = (1 - \theta\pi/2)\gamma h(x_2 + \sqrt{2\gamma h})^2/2$$

if $x_2 \geq -\sqrt{2\gamma h}/\lambda$, and

$$L_D(x, u) = \min\Big\{(1 - \theta\pi/2)\gamma h(x_2 + \sqrt{2\gamma h})^2/2,$$
$$(1 - \theta\pi/2)(x_2^2/2 - \gamma h)^2 - (1 + \theta\pi/2)(\lambda^2 x_2^2/2 - \gamma h)^2\Big\}$$

otherwise, which satisfies Condition (O2) due to the assumption on θ. Note that (12) holds since $W(x) = \gamma h$ on $\mathcal{A}$ and $x_2 = -\sqrt{2\gamma h}$ on $\mathcal{A} \cap D_\kappa$. By routine algebraic manipulations, it can also be shown that (11) holds. Finally, one can adapt the strategy in [8, Example 2.12] to show that $X = C'$ is forward invariant for $\mathcal{H}_\kappa$.

Simulation results[2] of the bouncing ball under hybrid MPC with $\gamma = 9.81$, $\lambda = 0.8$, $h = 2$, and a prediction horizon of the form (7) with $\tau_p = 6$, $\delta_p = 0.4$ are presented in Figure 1. For this simulation, Problem 3.2 is solved in MATLAB using the `fmincon` command by converting it into a finite-dimensional nonlinear program. Such an approach is possible due to conservation of energy during flows (the total energy W, and therefore L_C is invariant during flows) and the fact that the state trajectory of the bouncing ball during flows can be written in closed form. After every optimization, if the predicted trajectory jumps, the next optimization is triggered at the next jump time; otherwise, it occurs at the terminal time of the predicted trajectory. It can be seen that state trajectories from different initial conditions all converge to $\mathcal{A}$ after a few jumps.

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

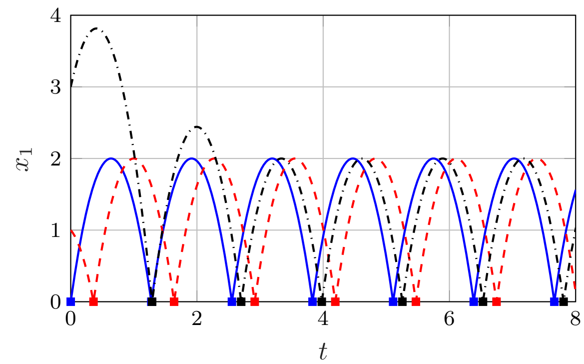

Fig. 1: Position trajectories of the bouncing ball from different initial conditions under hybrid MPC, projected onto t. Markers denote optimization times.

VIII. Conclusion

Borrowing tools from continuous/discrete-time MPC, we presented a CLF-based stabilizing MPC algorithm for hybrid systems, based on finite-horizon hybrid optimal control. A particular feature of the optimal control problem, which stands out in comparison to the continuous/discrete-time case, is the need to reconsider the notion of a terminal time to account for hybrid time domains. Future work will focus on the solution of the optimal control problem.

References

[1] D. Q. Mayne, "Model predictive control: Recent developments and future promise," *Automatica*, vol. 50, no. 12, pp. 2967–2986, 2014.

[2] E. Camacho, D. Ramirez, D. Limon, D. Muñoz de la Peña, and T. Alamo, "Model predictive control techniques for hybrid systems," *Annual Reviews in Control*, vol. 34, no. 1, pp. 21–31, 2010.

[3] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*. New York, NY: Cambridge University Press, 7 2017.

[4] P. Sopasakis, P. Patrinos, H. Sarimveis, and A. Bemporad, "Model predictive control for linear impulsive systems," *IEEE Transactions on Automatic Control*, vol. 60, no. 8, pp. 2277–2282, Aug 2015.

[5] F. L. Pereira, F. A. C. C. Fontes, A. P. Aguiar, and J. B. de Sousa, *An Optimization-Based Framework for Impulsive Control Systems*. Cham: Springer International Publishing, 2015, pp. 277–300.

[6] R. G. Sanfelice, *Hybrid Model Predictive Control*. Basel: Birkhäuser, 2019, pp. 199–220.

[7] H. Chen and F. Allgöwer, "A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability," *Automatica*, vol. 34, no. 10, pp. 1205–1217, 1998.

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

[9] J. Lygeros, K. H. Johansson, S. N. Simic, , and S. S. Sastry, "Dynamical properties of hybrid automata," *IEEE Transactions on Automatic Control*, vol. 48, no. 1, pp. 2–17, Jan 2003.

[10] B. Altın, P. Ojaghi, and R. G. Sanfelice, "A model predictive control framework for hybrid dynamical system," in *6th IFAC Conference on Nonlinear Model Predictive Control NMPC 2018*, 8 2018, pp. 128–133.

[11] A. Pakniyat and P. E. Caines, "On the relation between the minimum principle and dynamic programming for classical and hybrid control systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 9, pp. 4347–4362, Sept 2017.

[12] R. Goebel, "Existence of optimal controls on hybrid time domains," *Nonlinear Analysis: Hybrid Systems*, vol. 31, pp. 153–165, 2019.

[13] J. Chai and R. G. Sanfelice, "Forward invariance of sets for hybrid dynamical systems (Part I)," *IEEE Transactions on Automatic Control*, 2019.