Nonlinear Hybrid Control Systems

Ricardo G. Sanfelice

Electrical and Computer Engineering Department
University of California, Santa Cruz, USA

Article Outline

Notation

- $\mathbb{R}^n$ denotes n-dimensional Euclidean space and $\mathbb{R}$ denotes the real numbers.
- $\mathbb{R}_{\geq 0}$ denotes the nonnegative real numbers, i.e., $\mathbb{R}_{\geq 0} = [0, \infty)$.
- $\mathbb{Z}$ denotes the integers.
- $\mathbb{N}$ denotes the natural numbers including 0, i.e., $\mathbb{N} = \{0, 1, \ldots\}$.
- Given a set C, $\overline{C}$ denotes its closure and $\text{int}C$ its interior.
- The double-arrow notation, e.g., $F : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$, indicates a set-valued mapping, in contrast to a single arrow used for functions.
- Given $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^n$, (x, y) is equivalent to $[x^\top y^\top]^\top$.

I Introduction

This entry provides an introduction to hybrid control. Following the outline in the entry entitled *10008.Control of hybrid dynamical systems*, several frameworks for studying hybrid dynamics are presented and selected approaches to control a given system, referred to as the plant, using hybrid control are introduced.

Hybrid control algorithms are versatile due to allowing variables that change continuously and, at times, are updated to new values, instantaneously. This feature provides unique capabilities for control, leading to algorithms that can surpass the capabilities of controllers designed using classical control theory. For instance, a hybrid control algorithm can

- update the control input when the a new measurement arrives;
- reset a memory variable stored in the controller when computations terminates (e.g., the solution to an optimal control problem providing the control input value to use ends);
- change the gains of a controller architecture based on previous measurements.

Logic variables, timers, and memory states as state variables in a hybrid controller enable these unique features.

A control problem that challenges classical control theory is the design of an algorithm that robustly and globally asymptotically stabilizes a point mass evolving on the unit circle to a desired point. Specifically, the plant is given by

$$\dot{z} = u \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} z \qquad z \in \mathbb{S}^1, u \in \mathbb{R}$$

where $z = (z_1, z_2)$ is the state and u is the control input. This system is such that the state z remains in the unit circle $\mathbb{S}^1$ regardless of the choice of u. Suppose that the desired point is $z^* = (1, 0)$ and suppose one is able to design a static state-feedback controller of the form $u = \kappa(z)$ guaranteeing that, from every initial condition from $\mathbb{S}^1$, the resulting trajectories asymptotically converge to z^*. Without loss of generality, suppose that, from points with positive z_2, trajectories converge to z^* by rotating clockwise while from points with negative z_2, they converge by rotating counter clockwise. Then, either $-z^*$ is an equilibrium, in which case the convergence property is not global, or κ is discontinuous at z^*, in which case convergence is global but not robust as small noise in the state would lead to trajectories that stay nearby $-z^*$ for all time. In fact, it is easy to verify that there is no continuous κ that induces global convergence to the desired point. Fortunately, as shown in Section III.1, it is possible to design a hybrid controller that solves the problem. In simple words, the hybrid controller employs two static state-feedback laws in subsets of the unit circle and selects them using hysteresis, so as to not switch between the two laws at the same point.

A hybrid control approach to the design of classical sample-and-hold control and to event-triggered control guarantees asymptotic convergence not only at the event times but also in between those events, as the state evolves continuously. Characterizing such inter-event behavior is difficult using classical control theory. On the other hand, by using hybrid control theory one might be able to construct a decreasing Lyapunov function along both regimes. Another advantage of hybrid control is that it can implement both continuous-time and discrete-time feedback laws, in this way, being able to combine control strategies that are designed using classical control theory. Furthermore, a hybrid controller can effectively coordinate multiple control strategies, leading to a "divide and conquer" approach to modular control design.

Building from the outline in entry *10008.Control of hybrid dynamical systems*, the next section introduces frameworks that are suitable for modeling and analyzing systems exhibiting hybrid dynamics.

II Frameworks

This section introduces frameworks for the study of dynamical systems with some of the features seen in the examples in Section I. A brief outline of these frameworks is in the entry entitled *10008.Control of hybrid dynamical systems*.

II.1 Switched Systems

Switched systems are multi-mode continuous-time systems. Denoting the different modes by the discrete set $Q \subset \mathbb{N}$ and the state of the system by z, a switched system is modeled as

$$\dot{z} = f_{\sigma(t)}(z, u) \tag{1}$$

where $\{f_i\}_{i \in Q}$ is a family of functions governing the continuous evolution of z and $t \mapsto \sigma(t) \in Q$ is a continuous-time signal that changes its values over time. The control input is denoted by u. When the switching signal σ remains constant—for instance, suppose that $\sigma(t) = \sigma^*$ over the time interval

$[0, t_1]$, $t_1 > 0$—the state z evolves continuously according to the differential equation (1) with the right-hand side f_{σ^*}. When the switching signal changes value, the right-hand side associated to its new value is used to govern the continuous evolution of the state z. The state of the switched system does not exhibit a jump when the switching signal changes—its rate of may jumps due to the change of the right-hand side.

As expected, the rate of change of the switching signal is critical in the study of dynamical properties of a switched system. At times, the switching signal is considered to be the control input. A class of switching signals for which this rate is bounded is dwell-time switching signals. A switching signal is dwell time if the time elapsed between consecutive switching times is uniformly lower bounded by a positive number; that is, given a switching signal $t \mapsto \sigma(t)$, there exists $\tau^* > 0$ such that the switching times t_j of σ satisfy $t_{j+1} - t_j \geq \tau^*$ for all $j \in \mathbb{N}$ such that t_j and t_{j+1} are switching times. On the other hand, arbitrary switching signals are those for which switching times can occur arbitrarily fast. The literature considers classes of switching signals for which, on average, over a finite time window, the number of switches is bounded by a known constant – known as average dwell-time switching signals – at for which the dwell time bound hold every so often as time evolves – known as persistent dwell time switching signals.

Switched systems are particularly useful in situations where the switching signal is an external quantity, perhaps unknown (e.g., systems that may exhibit failures), and one is interested in guaranteeing that a desired system behavior is preserved regardless of the value of the switching signal.

II.2 Impulsive Systems

As a difference to switched systems, impulsive systems exhibit state jumps at pre-determined times $t_1, t_2, t_3, \ldots$. Denoting by z the state of the impulsive system, its dynamics are given by

$$\dot{z} = f(z, u_c) \qquad t \notin \mathcal{T} := \{t_1, t_2, t_3 \ldots\}$$

where f is the right-hand side, with impulses instantaneously changing the state via

$$z^+(t_i) = z(t_i) + g(z, u_d, t_i)$$

where g is a map capturing the change in z at the impulses. Here, the notation $z^+(t_i)$ corresponds to the right limit of the (left continuous) state trajectory $t \mapsto z(t)$. When the map g is nonzero at an impulse time, the state exhibits an instantaneous change of an amount characterized by g. Note that two types of inputs are included in the model above. The control input u_c affects the continuous evolution of the state in between the impulse times. The input u_d affects the state z at impulses only.

Impulsive system models emerge in optimal control problems, where the impulses typically corresponds to instantaneous changes in the control signal. These models are also useful when studying systems with state resets for which the impulse times are known in advance, for example, when they are scheduled or known as a function of the initial condition.

II.3 Hybrid Automata

Hybrid automata models allow for multiple modes of operation and instantaneous changes of the state. Similar to switched systems, a hybrid automata includes a logic mode state indicating the operating mode. It is a discrete state that can take values representing modes such as “on” or “off”; “low” or “high”; and “controller 1” or “controller 2.” The model also includes a “continuous state” collecting all variables that exhibit changes in between jumps. Similar to impulsive systems, the continuous state can also change at jumps. As a difference to both switched and impulsive systems, the jumps of the hybrid automata are triggered by state conditions that may involve both the discrete and continuous state.

The discrete state, denoted q, of the hybrid automata takes values from the finite set $Q := \{1, 2, \dots, q_{\max}\}$ where $q_{\max}$ indicates the number of modes of the system The continuous state, denoted by z, takes value from the Euclidean space $\mathbb{R}^n$, a subset of it, or more generally, a manifold of some finite dimension. For each $q \in Q$, the continuous state evolves according to

$$\dot{z} = f_q(z, u_c)$$

as long as it satisfies

$$z \in \mathcal{I}_q$$

The map f_q represents the right-hand side, depending on the state and input u_c affecting the continuous evolution. The set $\mathcal{I}_q$, a subset of $\mathbb{R}^n$, defines the set of points over which z is allowed to evolve continuously. It is sometimes referred to as the invariant set for each mode. Jumps in (q, z) are possible when q and z are such that

$$z \in \text{Guard}_{q,q'}$$

for some $q' \in Q$, at which event, the state (q, z) is reset via

$$q^+ = q', \qquad z^+ \in \text{Reset}(z, q, q')$$

The set $\text{Guard}_{q,q'}$ collects the points from which a transition from mode q to mode q' is possible. It is sometimes referred to as the guard set. The set-valued map Reset resets the continuous state according to the current value of (q, z) and of the value of q' for which the guard condition $z \in \text{Guard}_{q,q'}$ holds. Note that in this model, q remains constant during continuous evolution

II.4 Hybrid Inclusions

Hybrid inclusions are mathematical models used to describe systems that exhibit both continuous and discrete behavior. They extend the frameworks outlined above by explicitly incorporating set-valued maps to accommodate uncertainties and non-deterministic behaviors, nonunique solutions, solutions that end prematurely, and Zeno behavior. The evolution of a system with inputs modeled by a hybrid inclusion is governed by two types of dynamics:

I. Flow dynamics: The state x evolves according to $\dot{x} \in F(x, u_c)$ when $(x, u_c) \in C$.

II. Jump dynamics: The state x can instantaneously change to a new state $x^+ \in G(x, u_d)$ when $(x, u_d) \in D$.

A hybrid inclusion with inputs is described as

$$\mathcal{H} = (C, F, D, G)$$

where, when the state has dimension n, the input for the flows has dimension m_c, and the input for the jumps has dimension m_d, the data (C, F, D, G) is defined as follows:

- $C \subset \mathbb{R}^n \times \mathbb{R}^{m_c}$ is the flow set, which collect states and inputs for which the system can evolve continuously;
- $F : \mathbb{R}^n \times \mathbb{R}^{m_c} \rightrightarrows \mathbb{R}^n$ is the flow map, a set-valued map describing continuous evolution of the state;

- $D \subset \mathbb{R}^n \times \mathbb{R}^{m_d}$ is the jump set, which collects the set of states and inputs where jumps are allowed;
- $G : \mathbb{R}^n \times \mathbb{R}^{m_d} \rightrightarrows \mathbb{R}^n$ is the jump map, a set-valued map describing the jumps of the state.

This formulation allows hybrid inclusions to handle uncertainty captured by the set-valued nature of the model and nondeterminism, allowing for the system to evolve along multiple trajectories depending on the flow and jump sets.

The state $x \in \mathbb{R}^n$ captures all of the variables associated to the system. Its time derivative is denoted $\dot{x}$, while x^+ denotes its value after jumps. In several cases, the state x of the hybrid system can contain logic states that take value in discrete sets, as in hybrid automata.

Two parameters are used to specify "time" in solutions to hybrid systems: t, taking values in $\mathbb{R}_{\geq 0}$, and representing the elapsed "real" time; and j, taking values in $\mathbb{N}$, and representing the number of jumps that have occurred. For each solution, the combined parameters (t, j) will be restricted to belong to a hybrid time domain, a particular subset of $\mathbb{R}_{\geq 0} \times \mathbb{N}$. Hybrid time domains corresponding to different solutions may differ.

A set $E \subset \mathbb{R}_{\geq 0} \times \mathbb{N}$ is a hybrid time domain if it is the union of a nondecreasing sequence of compact hybrid time domains, namely, E is the union of compact hybrid time domains E_j with the property that $E_0 \subset E_1 \subset E_2 \subset \ldots \subset E_j \ldots$. In simple words, a hybrid time domains is the union of infinitely many intervals of the form $[t_j, t_{j+1}] \times \{j\}$, where $0 = t_0 \leq t_1 \leq t_2 \leq \ldots$, or of finitely many such intervals, with the last one possibly of the form $[t_j, t_{j+1}] \times \{j\}$, $[t_j, t_{j+1}) \times \{j\}$, or $[t_j, \infty) \times \{j\}$. On each hybrid time domain there is a natural ordering of points; namely, $(t, j) \preceq (t'j')$ for $(t, j), (t', j') \in E$ if $t \leq t'$ and $j \leq j'$.

A solutions to hybrid inclusion with inputs is given by a state-input pair. These functions are defined on hybrid time domains and satisfy the dynamics and the constraints given by the data of the hybrid system. The state component of a solution is a hybrid arc, which is a function $x : \operatorname{dom} x \to \mathbb{R}^n$ with $\operatorname{dom} x$ being a hybrid time domain and, for each fixed j, $t \mapsto x(t, j)$ being locally absolutely continuous function. The input component, $u = (u_c, u_d)$, is also defined on a hybrid time domain, and such that $t \mapsto u(t, j)$ is Lebesgue measurable for each j. A pair (x, u) with these properties is a solution to $\mathcal{H} = (C, F, D, G)$ if $\operatorname{dom}(x, u) = \operatorname{dom} x = \operatorname{dom} u$, $(x(0,0), u(0,0)) \in \overline{C} \cup D$, and the following hold:

Flow condition: for each $j \in \mathbb{N}$ such that $I^j := \{t \ : \ (t,j) \in \mathrm{dom}(x,u)\ \}$ has a nonempty interior

$$\dot{x}(t,j) \in F(x(t,j), u_c(t,j)) \qquad \text{for almost all } t \in I^j \tag{2}$$

and

$$(x(t,j), u_c(t,j)) \in C \qquad \text{for all } t \in \mathrm{int} I^j \tag{3}$$

Jump condition: for each $(t,j) \in \mathrm{dom}(x,u)$ such that $(t,j+1) \in \mathrm{dom}(x,u)$,

$$x(t,j+1) \in G(x(t,j), u_d(t,j)) \ \text{ and } \ (x(t,j), u_d(t,j)) \in D \tag{4}$$

When the maps F and G are single valued, (2) consists of a differential equation and (4) of a difference equation.
Further Reading: [22, 20, 60, 58, 59]

III Hybrid Control Approaches

A plant or a control algorithm with hybrid dynamics can be modeled as those of $\mathcal{H}$ in Section II.4. For instance, the impulsive neuron model and the walking robot model in entry entitled *10008.Control of hybrid dynamical systems* can be modeled as $\mathcal{H}$. We denote plants with hybrid dynamics as $\mathcal{H}_P$, with associated data (C_P, F_P, D_P, G_P, h), where h models the plant measurements available for control. The state of the hybrid plant is denoted z and its input u. Then, $\mathcal{H}_P$ is modeled as

$$\mathcal{H}_P \ : \ \begin{cases} (z,u) \in C_P & \dot{z} \ \in \ F_P(z,u) \\ (z,u) \in D_P & z^+ \ \in \ G_P(z,u) \\ & y \ = \ h(z) \end{cases}$$

Specifically, denoting the output of the hybrid plant by y, it defines it as $y = h(z,u)$.

Similarly, a hybrid controller is denoted $\mathcal{H}_K$ and its data as $(C_K, F_K, D_K, G_K, \kappa)$, where κ defines its output, which is denoted ζ. The state of the hybrid controller is η and its input is v. The model of the hybrid controller is

$$\mathcal{H}_K \ : \ \begin{cases} (v,\eta) \in C_K & \dot{\eta} \ \in \ F_K(v,\eta) \\ (v,\eta) \in D_K & \eta^+ \ \in \ G_K(v,\eta) \\ & \zeta \ = \ \kappa(v,\eta) \end{cases}$$

A simple example of a hybrid controller modeled as $\mathcal{H}_K$ is sample-and-hold control, which is introduced in a tutorial manner in [59]. Selected hybrid control strategies are introduced next. An outline of these strategies and several others are the entry entitled *10008.Control of hybrid dynamical systems*.

III.1 Uniting Control

Uniting control coordinates two feedback controllers to achieve robust and effective control. Using a logic-based rule, uniting control determines which controller to apply based on the current state of the plant. In this strategy, one of the controllers is considered to be suitable for use nearby the desired set-point z^*. This controller is referred to as the local controller. Another controller, referred to as the global controller,

The key idea of uniting control is to exploit the performance of the local controller near z^* and the global stabilizing capabilities of a global controller across the state space. Specifically:

- A local controller achieves efficient transient responses near z^* but may fail to stabilize the set-point globally.
- A global controller ensures convergence from any initial condition but may not be able to stabilize z^* or exhibit suboptimal performance around it— for instance, it might guarantee convergence to z^* but it might be too slow.

In simple words, uniting control globally asymptotically stabilizes a system to a set-point z^* by

- Ensuring local stability and convergence to z^* using a controller κ_0 for states near z^*.
- Leveraging a controller κ_1 to ensure global convergence to a neighborhood of z^* from which κ_0 can take over.

A hybrid controller implementing the logic above and modeled by $\mathcal{H}_K$ is given as follows:

- The state of the controller is a logic variable q taking values from $\{0, 1\}$. Hence, $\eta = q$.

- When the controller is of state-feedback type, it measures the state the of the plant. Hence, $v = z$.
- The logic-based rule implemented by the controller is as follows:

  I. If q is zero, apply κ_0 until z leaves a region nearby z^*. In the event that z leaves that region, reset q via

  $$q^+ = 1$$

  so that the global controller κ_1 can be applied to bring back the state close to z^*.

  II. If q is one, apply κ_1 until z reaches the region where κ_0 induces desired properties. Upon reaching that region, reset q via

  $$q^+ = 0$$

  so that the control law applied is κ_0 and the state asymptotically converges to z^* with good performance.

- In any other condition for z and q, keep the logic state q constant and allow z to evolve according to the dynamics of the plant under the effect of the controller κ_q.

The regions need to be carefully designed to avoid a large number of switches between the two controllers. Details on design can be found in the references cited below.

With uniting control within our control design toolbox, the problem of robustly and globally asymptotically stabilizing a point on the circle introduced in Section I can be solved. The two control laws in the hybrid control solution proposed therein can be designed as follows:

- The control law κ_0 almost globally asymptotically stabilizes $z^* = (1, 0)$;
- The control law κ_1 almost globally asymptotically stabilizes $z^\circ = (0, 1)$.

For this choice, a suitable region of operation for κ_0 is given by the points z in $\mathbb{S}^1$ with $z_1 \geq -\frac{1}{2}$. Denote this region by C_0. The region where κ_1 is used can be given by the points z in $\mathbb{S}^1$ with $z_1 \leq \frac{-1}{4}$. Denote this region by C_1. Then, switching from $q = 0$ to $q = 1$ should occur at points in $\overline{\mathbb{S}^1 \setminus C_0}$, region that is denoted by D_0. From those points, κ_1 takes the state towards z°, hence, eventually reaching C_0 in finite time. Hence, switching from $q = 1$ to $q = 0$ should occur at points in $\overline{\mathbb{S}^1 \setminus C_1}$, denoted D_1. Then, the closed-loop system is given by a hybrid system (without inputs) with state $(z, q) \in \mathbb{R}^2 \times \{0, 1\}$ with the following dynamics:

$$\begin{aligned} \dot{z} = \kappa_q(z) \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \ \dot{q} = 0 \qquad & z \in C_q \\ z^+ = z, \ q^+ = 1 - q \qquad & z \in D_q \end{aligned}$$

Using Lyapunov methods for hybrid systems, it can be shown that this hybrid closed-loop system has the set $\mathcal{A} := \{z_p^*\} \times \{0\}$ robustly globally asymptotically stable.

Further Reading: [51, 15, 5, 62, 68, 58, 29, 30]

III.2 Event-Triggered Control

As outlined in Section I, perhaps the most elementary algorithm with events is perhaps a sample-and-hold controller κ with periodic events associated to updating the control input. This hybrid controller features

- a memory variable ℓ_v that stores the latest value of the input applied to the plant;
- a resettable timer τ that triggers events every $T^* > 0$ seconds.

These events are triggered when

$$\tau \geq T^*$$

upon which the input to the plant is instantaneously updated to the value provided by κ at the current state, as follows:

$${\ell_v}^+ = \kappa(z)$$

and the plant input is assigned via

$$u = \ell_v$$

Conveniently, the condition $\tau \geq T^*$ triggering the events can be captured by the *event-triggering function*

$$\gamma(y, \eta) := \tau - T^*$$

which can be used to trigger events when

$$\gamma(y, \eta) \geq 0 \tag{5}$$

In general, an event-triggered controller features multiple event triggering conditions, which can be written using event-triggering functions. In principle, the following two types of events are of interest:

- *input events* triggered by the event-triggering function γ_u;
- *output events* triggered by the event-triggering function γ_y.

These functions may involve the state of the plant and of the controller, plus an additional state capturing extra dynamics of the controller, such as the timer τ triggering the events in sample-and-hold control.

A hybrid controller modeled as $\mathcal{H}_K$ implementing the event-trigger control logic is as follows:

- The state of the controller includes
  - states ℓ_y and ℓ_u storing the output of the plant and the value of control law, respectively;
  - an auxiliary state χ included to capture mechanisms involved in the event-triggered strategy.

  Hence, $\eta = (\ell_y, \ell_u, \chi)$, where
- The controller input measures the output of the plant. Hence, $v = y$.
- The logic-based rule implemented by the controller is as follows:

  I. If γ_y is larger than or equal to zero, then trigger an output event. At such event, update ℓ_y appropriately.

  II. If γ_u is larger than or equal to zero, then trigger an input event. At such event, update ℓ_u appropriately.

  At each of the events, the auxiliary variable χ is reset appropriately as well.
- In any other condition for y, ℓ_y, ℓ_u, and χ, the states are allowed to evolve continuously appropriately.

To ground this general construction, consider the control the plant

$$\dot{z} = \widetilde{f}(z, u) \qquad (z, u) \in \mathbb{R}^{n_z} \times \mathbb{R}^{m_z}$$

using the static state-feedback law κ via sample-and-hold control. The state of the hybrid controller is given by

$$\eta = (\ell_v, \tau)$$

and the resulting hybrid closed-loop system has state (z, ℓ_v, τ) with the following dynamics:

$$\begin{aligned} (\dot{z}, \dot{\ell}_v, \dot{\tau}) &= (\widetilde{f}(z, \ell_v), 0, 1) \qquad \tau \in [0, T^*] \\ (z^+, \ell_v^+, \tau^+) &= (z, \kappa(z), 0) \qquad \tau \geq T^* \end{aligned}$$

This hybrid closed-loop system can be expressed as a hybrid system $\mathcal{H}$ without inputs having state $x = (z, \eta)$ and the following data:

- The flow set C is given by

$$C := \{x \in \mathbb{R}^n \ : \ \tau \in [0, T^*] \ \} = \mathbb{R}^{n_\xi} \times \mathbb{R}^{m_\xi} \times [0, T^*]$$

- The flow map F is given by

$$F(x) := (\widetilde{f}(\xi, \ell_v), 0, 1)$$

- The jump set D is given by

$$D := \{x \in \mathbb{R}^n \ : \ \tau = T^* \ \} = \mathbb{R}^{n_z} \times \mathbb{R}^{m_z} \times \{T^*\}$$

  where $n = n_z + m_z + 1$.

- The jump map is given by

$$G(x) := (\xi, \kappa_c(\xi), 0)$$

Further Reading: [70, 27, 50, 71, 11, 48, 58, 12, 31, 8, 49]

III.3 Synergistic Control

Synergistic control is a hybrid strategy designed to achieve global asymptotic stabilization of a desired compact set, denoted $\mathcal{A}$, using multiple feedback laws and Lyapunov certificates. This method uses a family of Lyapunov functions and their corresponding state-feedback laws to guide the system toward the goal while overcoming challenges such as local minima in individual Lyapunov functions. The primary idea is to construct a finite set of pairs

$$\{(V_i, \kappa_i)\}_{i \in \{0,1,\ldots,N\}}$$

where each pair consists of a Lyapunov-like function V_i and a state-feedback law κ_i, with N being an integer larger than one. Each Lyapunov function V_i is such that

I. $V_i(x) = 0$ on $\mathcal{A}$ and $V_i(x) > 0$ elsewhere;

II. V_i decreases along system trajectories under the feedback law κ_i, except at certain problematic points (e.g., local minima or critical points).

At points where V_i no longer decreases, the synergistic control strategy switches to another pair (V_k, κ_k), where $V_k(x) < V_i(x)$. This ensures continued progress toward $\mathcal{A}$. To manage the switching between feedback laws, a logic variable $q \in \{0, 1, \ldots, N\}$ indexes the currently active Lyapunov function-feedback pair (V_q, κ_q).

A hybrid controller implementing the logic above and modeled by $\mathcal{H}_K$ is given as follows:

- The state of the controller is a logic variable q taking values from $\{0, 1, \ldots, N\}$. Hence, $\eta = q$.
- When the controller is of state-feedback type, it measures the state the of the plant. Hence, $v = z$.
- The logic-based rule implemented by the controller is as follows:

  I. If the Lyapunov function V_q satisfies

  $$V_q(x) \geq V_k(x) + \delta \quad \text{for some } k \in \{0, 1, \ldots, N\},$$

  where $\delta > 0$ is a positive parameter, the logic variable is updated via

  $$q^+ = k$$

II. The feedback law κ_k ensuring a decrease in the Lyapunov function by at least δ is used.

The mechanism implemented by the logic is crucial for maintaining progress, especially at points where the Lyapunov function associated to the current feedback law stops decreasing. In fact, the logic ensures that the Lyapunov function decreases consistently along the hybrid closed-loop system's trajectories.
Further Reading: [39, 38, 37, 7, 9, 58, 10]

IV Conclusion

Hybrid control extends classical control theory by allowing the use of variables that can change continuously or discretely. The unique combination of continuous and discrete dynamics allowed by hybrid control enables the solution of control problems that classical continuous-time and discrete-time control design tools cannot resolve. In recent years, the field of hybrid systems and control has advanced significantly, introducing powerful control strategies. These strategies enable systematic analysis and control design for systems with intermittent information, multiple modes of operation, or abrupt state changes?such as impacts, faults, and resets. Hybrid control theory is a dynamic and exciting research area, offering abundant opportunities for developing new theoretical insights and advancing emerging applications in science and engineering.

References

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

[2] B. Altin and R. G. Sanfelice. Model predictive control for hybrid dynamical systems: Sufficient conditions for asymptotic stability with persistent flows or jumps. In *Proceedings of the American Control Conference*, pages 1791–1796, July 2020.

[3] A. D. Ames, K. Galloway, and J.W. Grizzle. Control Lyapunov functions and hybrid zero dynamics. In *Proceedings of American Control Conference*, pages 6837–6842, December 2012.

[4] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada. Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control*, 62(8):3861–3876, 2017.

[5] V. Andrieu and C. Prieur. Uniting two control Lyapunov functions for affine systems. *IEEE Transactions on Automatic Control*, 55(8):1923–1927, 2010.

[6] K. J. Astrom and K. Furuta. Swinging up a pendulum by energy control. *Automatica*, 36:287–295, 2000.

[7] P. Casau, C. G. Mayhew, R. G. Sanfelice, and C. Silvestre. Exponential stabilization of a vectored-thrust vehicle using synergistic potential functions. In *Proceedings of American Control Conference*, pages 6042–6047, 2016.

[8] P. Casau, R.G.Sanfelice, and C. Silvestre. On the robustness of nominally well-posed event-triggered controllers. In *Proceedings of the 60th IEEE Conference on Decision and Control*, December 2021.

[9] P. Casau, R. G. Sanfelice, and C. Silvestre. Adaptive backstepping of synergistic hybrid feedbacks with application to obstacle avoidance. In *Proceedings of the American Control Conference*, pages 1730–1735, July 2019.

[10] P. Casau, R. G. Sanfelice, and C. Silvestre. Robust synergistic hybrid feedback. *IEEE Transactions on Automatic Control*, pages 1–16, December 2024.

[11] J. Chai, P. Casau, and R. G. Sanfelice. Analysis and design of event-triggered control algorithms using hybrid systems tools. In *Proceedings of the 2017 IEEE Conference on Decision and Control*, pages 6057–6062, 2017.

[12] J. Chai, P. Casau, and R. G. Sanfelice. Analysis and design of event-triggered control algorithms using hybrid systems tools. *International Journal of Robust and Nonlinear Control*, April 2020.

[13] J. Chai and R. G. Sanfelice. Forward invariance of sets for hybrid dynamical systems (Part I). *IEEE Transactions on Automatic Control*, 64:2426–2441, 06/2019 2019.

[14] J. Chai and R. G. Sanfelice. Forward invariance of sets for hybrid dynamical systems (part ii). *To appear in IEEE Transactions on Automatic Control*, 2020.

[15] D. V. Efimov. Uniting global and local controllers under acting disturbances. *Automatica*, 42:489–495, 2006.

[16] G. E. Fainekos, A. Girard, H. Kress-Gazit, and G. J. Pappas. Temporal logic motion planning for dynamic robots. *Automatica*, 45(2):343–352, 2009.

[17] I. Fantoni, R. Lozano, and M. W. Spong. Energy based control of the pendubot. *IEEE Transactions on Automatic Control*, 45:725–729, 2000.

[18] R. Goebel, C. Prieur, and A. R. Teel. Smooth patchy control Lyapunov functions. *Automatica*, 45(3):675–683, 2009.

[19] R. Goebel, R. G. Sanfelice, and A. R. Teel. Invariance principles for switching systems via hybrid systems techniques. *Systems & Control Letters*, 57(12):980–986, December 2008.

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

[21] R. Goebel, R. G. Sanfelice, and A. R. Teel. Some converse lyapunov-like results for strong forward invariance. In *Proceedings of the 63rd IEEE Conference on Decision and Control*, December 2024.

[22] R. Goebel, R. G. Sanfelice, and A.R. Teel. Hybrid dynamical systems. *IEEE Control Systems Magazine*, 29(2):28–93, April 2009.

[23] W. M. Haddad, S. G. Nersesov, and V. Chellaboina. Energy-based control for hybrid port-controlled Hamiltonian systems. *Automatica*, 39(8):1425–1435, 2003.

[24] H. Han, M. Maghenem, and R. G. Sanfelice. Certifying the ltl formula p until q in hybrid systems. *IEEE Transactions on Automatic Control*, 68(7):4451–4458, September 2022.

[25] H. Han and R. G. Sanfelice. Linear temporal logic for hybrid dynamical systems: Characterizations and sufficient conditions. *Nonlinear Analysis: Hybrid Systems*, 36, 05/2020 2020.

[26] H. Han and R. G. Sanfelice. Exploiting invariance properties to certify always and eventually signal temporal logic operators for hybrid dynamical systems. *IEEE Control Systems Letters*, 7:2809–2814, June 2023.

[27] W.P.M.H. Heemels, K. H. Johansson, and P. Tabuada. An introduction to event-triggered and self-triggered control. In *Proceedings of the IEEE 51st Annual Conference on Decision and Control*, pages 3270–3285. IEEE, 2012.

[28] J. P. Hespanha, D. Liberzon, and A. S. Morse. Supervision of integral-input-to-state stabilizing controllers. *Automatica*, 38:1327–335, 2002.

[29] D. Hustig-Schultz and R. G. Sanfelice. Uniting nesterov's accelerated gradient descent and the heavy ball method for strongly convex functions with exponential convergence rate. In *Proceedings of the American Control Conference*, May 2021.

[30] D. Hustig-Schultz and R. G. Sanfelice. Uniting nesterov and heavy ball methods for uniform global asymptotic stability of the set of minimizers. *Automatica*, 160:111389, February 2024.

[31] D. Kooi and R. G. Sanfelice. A self-triggered control strategy to guarantee forward invariance. In *Proceedings of the American Control Conference*, May 2021.

[32] X.D. Koutsoukos, P.J. Antsaklis, J.A. Stiver, and M.D. Lemmon. Supervisory control of hybrid systems. *Proc. IEEE*, 88(7):1026–1049, July 2000.

[33] M. Maghenem and R. G. Sanfelice. Multiple barrier function certificates for weak forward invariance in hybrid inclusions. In *Proceedings of*

the 2019 IEEE Conference on Decision and Control, pages 6319–6324, December 2019.

[34] M. Maghenem and R. G. Sanfelice. Sufficient conditions for forward invariance and contractivity in hybrid inclusions using barrier functions. *Automatica*, 124, 2021.

[35] B. P. Malladi, R. G. Sanfelice, E. Butcher, and J. Wang. Robust hybrid supervisory control for rendezvous and docking of a spacecraft. In *In Proceedings of the Conference on Decision and Control*, pages 3325 – 3330, 2016.

[36] B. P. Malladi, R. G. Sanfelice, and E.A. Butcher. Robust hybrid supervisory control for spacecraft close proximity missions. *To appear in Annual Reviews in Control*, 52, January 2021.

[37] C. G. Mayhew, R. G. Sanfelice, and A. R. Teel. Further results on synergistic Lyapunov functions and hybrid feedback design through backstepping. In *Proc. Joint Conference on Decision and Control and European Control Conference*, page 7428–7433, 2011.

[38] C. G. Mayhew, R. G. Sanfelice, and A. R. Teel. Synergistic Lyapunov functions and backstepping hybrid feedbacks. In *Proc. 30th American Control Conference*, page 3203–3208, 2011.

[39] C. G. Mayhew and A. R. Teel. Hybrid control of rigid-body attitude with synergistic potential functions. In *Proceedings of the American Control Conference*, pages 287–292, 2011.

[40] A. S. Morse. Supervisory control of families of linear set-point controllers - Part 1: Exact matching. *IEEE Transactions on Automatic Control*, 41:1413–1431, 1996.

[41] A. S. Morse. Supervisory control of families of linear set-point controllers - Part 2: Robustness. *IEEE Transactions on Automatic Control*, 42:1500–1515, 1997.

[42] R. Naldi and R. G. Sanfelice. Passivity-based controllers for a class of hybrid systems with applications to mechanical systems interacting with their environment. In *Proc. Joint Conference on Decision and Control and European Control Conference*, page 7416–7421, 2011.

[43] R. Naldi and R. G. Sanfelice. Passivity-based control for hybrid systems with applications to mechanical systems exhibiting impacts. *Automatica*, 49(5):1104–1116, May 2013.

[44] R. Naldi and R. G. Sanfelice. Sufficient conditions for passivity and stability of interconnections of hybrid systems using sums of storage functions. In *Proceedings of the 2014 American Control Conference*, pages 1432–1437, 2014.

[45] P. Nanez, R. G. Sanfelice, and N. Quijano. Notions and a passivity tool for switched dae systems. In *Proceedings of the 2017 IEEE Conference on Decision and Control*, pages 3612–3617, 2017.

[46] R. O'Flaherty, R. G. Sanfelice, and A. R. Teel. Robust global swing-up of the pendubot via hybrid control. In *Proc. 27th American Control Conference*, page 1424–1429, 2008.

[47] A.Y. Pogromsky, M. Jirstrand, and P. Spangeous. On stability and passivity of a class of hybrid systems. In *Proc. 37th IEEE Conference on Decision and Control*, pages 3705–3710, Tampa, Florida, USA, 1998.

[48] R. Postoyan, W. P. M. H. Heemels, and R. G. Sanfelice. Inter-event times analysis for planar linear event-triggered controlled systems (i). In *Proceedings of the 2019 IEEE Conference on Decision and Control*, pages 1662 – 1667, December 2019.

[49] R. Postoyan, R.G. Sanfelice, and W. P. M. H. Heemels. Explaining the "mystery" of periodicity in inter-transmission times in two-dimensional event-triggered controlled system. *To appear in the IEEE Transactions on Automatic Control*, February 2023.

[50] R. Postoyan, P. Tabuada, D. Nešić, and A. Anta. A framework for the event-triggered stabilization of nonlinear systems. *IEEE Transactions on Automatic Control*, 60(4):982–996, 2015.

[51] C. Prieur. Uniting local and global controllers with robustness to vanishing noise. *Math. Control Signals Systems*, 14:143–172, 2001.

[52] R. G. Sanfelice. Control Lyapunov functions and stabilizability of compact sets for hybrid systems. In *Proceedings of the Joint Conference on Decision and Control and European Control Conference*, page 7404–7409, 2011.

[53] R. G. Sanfelice. On the existence of control Lyapunov functions and state-feedback laws for hybrid systems. *IEEE Transactions on Automatic Control*, 58(12):3242–3248, December 2013.

[54] R. G. Sanfelice. Pointwise minimum-norm control laws for hybrid systems. In *Proceedings of the IEEE Conference on Decision and Control*, page 2665–2670, 2013.

[55] R. G. Sanfelice. A computationally tractable implementation of pointwise minimum norm state-feedback laws for hybrid systems. In *Proceedings of American Control Conference*, pages 4257–4262, 2016.

[56] R. G. Sanfelice. Robust asymptotic stabilization of hybrid systems using control lyapunov functions. In *Proceedings of the 19th International Conference on Hybrid Systems: Computation and Control*, pages 235–244, April 2016.

[57] R. G. Sanfelice. *Hybrid Model Predictive Control*, pages 199–220. Birkhäuser, Basel, edition 1 edition, 09/2018 2018.

[58] R. G. Sanfelice. *Hybrid Feedback Control*. Princeton University Press, New Jersey, 2021.

[59] R. G. Sanfelice, P. Casau, and F. Ferrante. Hybrid feedback control design. In *Proceedings of the 63rd IEEE Conference on Decision and Control*, December 2024.

[60] R. G. Sanfelice, D. A. Copp, and P. Nanez. A toolbox for simulation of hybrid systems in Matlab/Simulink: Hybrid Equations (HyEQ) Toolbox. In *Proceedings of Hybrid Systems: Computation and Control Conference*, page 101–106, 2013.

[61] R. G. Sanfelice, R. Goebel, and A. R. Teel. Invariance principles for hybrid systems with connections to detectability and asymptotic stability. *IEEE Transactions on Automatic Control*, 52(12):2282–2297, 2007.

[62] R. G. Sanfelice and C. Prieur. Robust supervisory control for uniting two output-feedback hybrid controllers with different objectives. *Automatica*, 49(7):1958–1969, July 2013.

[63] R. G. Sanfelice and A. R. Teel. A "throw-and-catch" hybrid control strategy for robust global stabilization of nonlinear systems. In *Proc. 26th American Control Conference*, page 3470–3475, 2007.

[64] R. G. Sanfelice and A. R. Teel. On invariants for open hybrid systems and their interconnections. In *Proceedings of the 62nd IEEE Conference on Decision and Control*, pages 1859–1864, December 2023.

[65] R. G. Sanfelice, A. R. Teel, and R. Goebel. Supervising a family of hybrid controllers for robust global asymptotic stabilization. In *Proc. 47th IEEE Conference on Decision and Control*, page 4700–4705, 2008.

[66] R.G. Sanfelice. *Hybrid Model Predictive Control*. Springer, 2020.

[67] A. Saoud and R.G. Sanfelice. Computation of controlled invariants for nonlinear systems: Application to safe neural networks approximation and control. In *Proceedings of the 7th Analysis and Design of Hybrid Systems*, volume 54, July 2021.

[68] D. W. Smith and R. G. Sanfelice. A hybrid feedback control strategy for autonomous waypoint transitioning and loitering of unmanned aerial vehicles. *Nonlinear Analysis: Hybrid Systems*, 26:115–136, 2017.

[69] M. W. Spong, J. K. Holm, and D. Lee. Passivity-based control of bipedal locomotion. *IEEE Robotics & Automation Magazine*, pages 30–40, June 2007.

[70] P. Tabuada. Event-triggered real-time scheduling of stabilizing control tasks. *IEEE Transactions on Automatic Control*, 52(9):1680–1685, 2007.

[71] T. A. F. Theunisse, J. Chai, R. G. Sanfelice, and M. Heemels. Robust global stabilization of the DC-DC boost converter via hybrid control. *IEEE Transactions on Circuits and Systems I*, 62:1052–1061, April 2015.

[72] P. K. Wintz and R.G. Sanfelice. Forward invariance-based hybrid control using uncertified controllers (i). In *Proceedings of the 62nd IEEE Conference on Decision and Control*, December 2023.

[73] G. Zucchini, B. P. Malladi, R. G. Sanfelice, and E. Butcher. Robust hybrid supervisory control for a 3-dof spacecraft in close-proximity operations. In *Networked & Autonomous Air & Space Systems*, pages 88–93, 06/2018 2018.