A Timer-Based Hybrid Supervisor for Robust, Chatter-Free Policy Switching

Anonymous authors
Paper under double-blind review

Keywords: Chattering, robustness, value function-based switching, hybrid control.

Summary

We address the challenge of switching among multiple learned policies in reinforcement learning control systems, where conventional value function–based methods can lead to chattering in the presence of small measurement noise. Our goal is to design a switching logic that preserves the asymptotic stability of each individual policy, achieves overall performance that is at least as good as any fixed policy, and maintains a robustness margin so that rapid switching is prevented for any bounded measurement noise. To this end, we propose a timer-based hybrid supervisor that integrates a resettable timer that enforces a minimum dwell time on the active policy. This dwell time is adaptively adjusted by predicting the evolution of the state of the system, ensuring that a switch occurs only when a significantly better alternative is predicted.

We derive sufficient conditions under which the hybrid supervisor is guaranteed to exhibit non-Zeno behavior and render a compact set robustly globally asymptotically stable in the presence of bounded measurement noise. Simulation results on representative decision-making problems demonstrate that our hybrid supervisors maintain performance and robustness under noisy conditions where a conventional switching strategy fails.

Contribution(s)

1. This paper presents a hybrid supervisor that maintains the globally asymptotic stability properties of the underlying policies and prevents chattering between the policies under bounded measurement noise. The hybrid supervisor deploys a timer-based mechanism to predict and enforce a dwell period between policy switches.
   Context: Chattering refers to the phenomenon of a system rapidly switching its decision due to measurement noise that results in inefficient or destabilizing behavior. Prior solutions rely on spatial or temporal regulation, such as hierarchical methods. In contrast to existing spatial regulation approaches, the proposed method can be applied to high-dimensional state spaces. Compared to existing temporal regulation approaches, the proposed method guarantees that chattering cannot occur.

A Timer-Based Hybrid Supervisor for Robust, Chatter-Free Policy Switching

Anonymous authors
Paper under double-blind review

Abstract

We address the challenge of switching among multiple learned policies in reinforcement learning control systems, where conventional value function–based methods can lead to chattering in the presence of small measurement noise. Our goal is to design a switching logic that preserves the asymptotic stability of each individual policy, achieves overall performance that is at least as good as any fixed policy, and maintains a robustness margin so that rapid switching is prevented for any bounded measurement noise. To this end, we propose a timer-based hybrid supervisor that integrates a resettable timer that enforces a minimum dwell time on the active policy. This dwell time is adaptively adjusted by predicting the evolution of the state of the system, ensuring that a switch occurs only when a significantly better alternative is predicted. We derive sufficient conditions under which the hybrid supervisor is guaranteed to exhibit non-Zeno behavior and render a compact set robustly globally asymptotically stable in the presence of bounded measurement noise. Simulation results on representative decision-making problems demonstrate that our hybrid supervisors maintain performance and robustness under noisy conditions where a conventional switching strategy fails.

1 Introduction

In many real-world decision-making problems, a reinforcement learning (RL) agent is faced with a choice among multiple strategies or actions, each with its own advantages and trade-offs. Such problems arise not only in control systems and robotics (Hwangbo et al., 2019), but also in areas as diverse as financial portfolio management (Bartram et al., 2021), cybersecurity (Alpcan & Baar, 2010), and video game strategy selection (Yannakakis & Togelius, 2018). In these settings, the agent must evaluate limited, often noisy information in order to select the best course of action, balancing short-term rewards against long-term objectives. This dynamic, multi-strategy decision-making process is inherently challenging, particularly in RL, where the optimal strategy may not be immediately apparent and can depend on subtle aspects of the current state or adversarial influences in the environment.

One particularly challenging phenomenon in this context is the tendency for the system to "chatter" between strategies (Prieur et al., 2007; Mayhew et al., 2011; de Priester et al., 2022; 2024). Chattering occurs when an agent rapidly switches its decision—often due to small measurement errors or environmental perturbations—resulting in inefficient or even destabilizing behavior. For example, in video game scenarios, a player or AI might oscillate between aggressive, defensive, and resource-gathering strategies in response to transient changes in the game state. Although such switching might appear adaptive in the short term, it can prevent the full exploitation of a promising strategy, ultimately leading to suboptimal performance. Furthermore, in competitive environments, adversaries can deliberately manipulate the situation to induce premature or frequent switching, thereby exploiting the hesitation or uncertainty of the decision maker (de Priester et al., 2022; 2024).

Existing approaches to mitigate chattering typically rely on spatial regulation or temporal abstraction, as seen in traditional Hierarchical Reinforcement Learning (HRL) frameworks (Sutton et al., 1999; Makumi et al., 2023). Prior works (de Priester et al., 2022; 2024) have explored robustifying the switching logic through spatial regulation, which involves defining specific regions in the state space where policy switching is permitted or restricted. However, spatial regulation may not be ideal when defining such regions is challenging or impractical—for instance, in environments with high-dimensional state spaces. Policy parameterizations are often continuous or memoryless discrete, and even strategies that incorporate state or input memory struggle because the required memory length depends on the magnitude of the perturbations; insufficient memory can trap the system in critical areas. More complex solutions—such as adaptive input spaces—further reduce policy interpretability. Moreover, even when temporal regulation is employed in such ways, the execution duration of a policy or option is not explicitly linked to robustness against chattering, offering no guarantees that rapid switching will be prevented. To address these limitations, we complement the prior works by developing a new state value function-based approach that robustifies RL-based supervisory policies.

Motivated by these challenges, we propose a novel hybrid supervisor that maintains the performance of the underlying policies and prevents chattering. In particular, the hybrid supervisor deploys a timer-based mechanism to predict and enforce a dwell period between policy switches. In Section 4, we provide a formal analysis of the proposed hybrid supervisor, establishing properties such as non-Zeno behavior and robustness to measurement noise. In Section 5, we validate our approach through numerical simulations on a representative decision-making problem.[1] Section 3 further motivates our work by presenting a detailed problem formulation and an illustrative example of the chattering issues observed with conventional switching strategies.

2 Preliminaries

2.1 Notation

The following notation is used throughout the paper. The n-dimensional Euclidean space is denoted by $\mathbb{R}^n$. The real numbers are denoted by $\mathbb{R}$. The nonnegative real numbers are denoted by $\mathbb{R}_{\geq 0}$, i.e., $\mathbb{R}_{\geq 0} := [0, \infty)$. The positive real numbers are denoted by $\mathbb{R}_{>0}$, i.e., $\mathbb{R}_{>0} := (0, \infty)$. The natural numbers including 0 are denoted by $\mathbb{N}$, i.e., $\mathbb{N} := \{0, 1, 2, ...\}$. The natural numbers excluding 0 are denoted by $\mathbb{N}_{>0}$, i.e., $\mathbb{N}_{>0} := \{1, 2, ...\}$. The empty set is denoted by $\varnothing$. The interior of the set S is denoted by $\text{int}\,(S)$. The boundary of the set S is denoted by ∂S. The closed unit ball, of appropriate dimension and centered at the origin, in the Euclidean norm is denoted by $\mathbb{B}$. The Euclidean norm of the vector x is denoted by $|x|$. The distance from x to the nonempty set S is denoted by $|x|_S$, which is given by $\inf_{y \in S} |x - y|$. The domain of a map f is denoted by $\text{dom}\ f$. The signum function is denoted by sgn and is defined as $\text{sgn}(\chi) := -1$ if $\chi < 0$ and $\text{sgn}(\chi) := 1$ if $\chi \geq 0$. The tangent cone to the set $S \subset \mathbb{R}^n$ at $\chi \in \mathbb{R}^n$ is denoted by $T_S(\chi)$ and defined as the set of all vectors $w \in \mathbb{R}^n$ for which there exists sequences $\chi_i \in S$, $\tau_i > 0$ with $\chi_i \to \chi$, $\tau_i \searrow 0$, and $w = \lim_{i \to \infty} \frac{\chi_i - \chi}{\tau_i}$. A set-valued map F is outer semicontinuous if for each $x \in \text{dom}\ F$, every sequence of points $x_i \in \text{dom}\ F$ converging to x is such that every sequence $y_i \in F(x_i)$ converging to y satisfies $y \in F(x)$. A set-valued map F is outer semicontinuous if and only if its graph, denoted by $\text{gph}(F)$, is closed. If F is single-valued and continuous, it is also outer semicontinuous.

2.2 Reinforcement Learning Framework

Markov decision processes (MDPs) are used as a formalism for RL (Puterman, 1994). In an MDP, the learner/controller is referred to as the agent and interacts with an environment. The state of the agent $z \in \mathcal{Z}$, where $\mathcal{Z} \subset \mathbb{R}^n$ is a set of states, evolves according to its continuous-time dynamics

$$\dot{z} = f(z, u), \tag{1}$$

[1] All simulation files are available in the supplementary material.

where $f : \mathcal{Z} \times \mathcal{U} \to \mathcal{Z}$ and $u \in \mathcal{U}$ is the control input, where $\mathcal{U} \subset \mathbb{R}^m$ is a set of actions. For computational purposes, the continuous-time dynamics (1) are discretized to yield the discrete-time system

$$z^+ = z + f(z, u)\Delta t, \tag{2}$$

where $z^+ \in \mathcal{Z}$ is the next value of the state z and $\Delta t \in \mathbb{R}_{>0}$ is the sampling time in seconds. A solution pair to (2) of length $J \in \mathbb{N} \cup \{\infty\}$ is denoted by $(z, u) = \{(z_0, u_0), (z_1, u_1), \ldots, (z_j, u_j)\} = \{(z_j, u_j)\}_{j=0}^{J} \in \mathcal{S}$, where $\mathcal{S}$ is the set of solution pairs to (2). The *discounted reward functional* $\mathcal{V} : \mathcal{S} \to \mathbb{R}$ maps solutions of (2) to a scalar, that is, the discounted reward, and is defined as

$$\mathcal{V}(z, u) := \sum_{j=0}^{J} \gamma^j R(z_j, u_j), \tag{3}$$

where (z, u) is a solution pair to (2), $\gamma \in (0, 1)$ is the discount factor, and $R : \mathcal{Z} \times \mathcal{U} \to \mathbb{R}$ is the reward function. The *state value function* for a policy $\pi : \mathcal{Z} \to \mathcal{U}$ is given by

$$\tilde{V}^\pi(z_0) = \mathcal{V}(z, \pi(z)) = \sum_{j=0}^{J} \gamma^j R(z_j, \pi(z_j)), \tag{4}$$

where (z, u) is the unique solution pair of (2) starting from z_0 under the policy π. In this work, we apply value iteration to obtain an approximate state value function $V : \mathcal{Z} \to \mathbb{R}$ that approximates the true state value function $\tilde{V}^\pi$ defined in (4). We represent V using a multi-layer perceptron (MLP) with L layers and continuously differentiable activation functions, such as the sigmoid or hyperbolic tangent function. The approximate state value function is given by

$$V(z; \theta) = W_L\, h\Big(W_{L-1}\, h\Big(\cdots\, h\big(W_1 z + b_1\big)\cdots\Big) + b_{L-1}\Big) + b_L, \tag{5}$$

where $\theta = \{W_1, W_2, \ldots, W_L, b_1, b_2, \ldots, b_L\}$ denotes the collection of weights and biases, and $h : \mathbb{R} \to \mathbb{R}$ is a continuously differentiable activation function. For conciseness, we omit the explicit dependency on the network parameters θ in the remainder of the paper. The value iteration algorithm implemented is analogous to the training of the critic network in actor-critic methods, such as Proximal Policy Optimization (PPO) (Schulman et al., 2017).[2]

2.3 Hybrid Systems

A hybrid system $\mathcal{H} = (C, F, D, G)$ is defined as

$$\mathcal{H} : \begin{cases} \dot{x} &= F(x) \quad x \in C \\ x^+ &= G(x) \quad x \in D \end{cases} \tag{6}$$

where $x \in \mathbb{R}^n$ denotes the state variable, x^+ the state variable after a jump, $F : C \to \mathbb{R}^n$ is a function referred to as the flow map, $C \subset \mathbb{R}^n$ is the set of points referred to as the flow set, $G : D \to \mathbb{R}^n$ the jump map, and $D \subset \mathbb{R}^n$ is the jump set. When the state is in the flow set, the state is allowed to evolve continuously and is described by the differential equation defined by the flow map. When the state is in the jump set, the state is allowed to be updated using the difference equation defined by the jump map. In this way, with some abuse of notation, the solution to (6) is given by a function $(t, j) \mapsto x(t, j)$ defined on a hybrid time domain, which properly collects values of the ordinary time variable $t \in \mathbb{R}_{\geq 0}$ and of the discrete jump variable $j \in \mathbb{N}$. The hybrid system $\mathcal{H}$ allows for the combination of continuous-time behavior (flow) with discrete-time behavior (jumps). For more details on hybrid dynamical systems, see Goebel et al. (2012); Sanfelice (2021).

[2] For details on the implementation of the value iteration algorithm, see the attached .zip file.

3 Motivation

3.1 Problem Definition

We consider systems described by the dynamics in (1). The control input to this system is provided by a continuous policy $\pi_q \in \Pi := \{\pi_1, \pi_2, \ldots, \pi_N\}$, $\pi_q : \mathcal{Z} \to \mathcal{U}$ selected from a policy bank Π, where each policy π_q maps states to control actions and $N \in \mathbb{N}_{>0}$ is the number of policies in the policy bank Π. Furthermore, each policy asymptotically stabilizes a compact set. A continuously differentiable approximate state value function $V_q : \mathcal{Z} \to \mathbb{R}$ of the form (5) is obtained for each policy $\pi_q \in \Pi$ for $q \in \mathcal{Q} := \{1, 2, \ldots, N\}$ via RL. The problem to solve is defined as follows:

Problem ($\star$) Design a state value function-based switching logic that:

(P1) Preserves the properties of the individual policies, namely, inducing asymptotic stability of a compact set;

(P2) Achieves overall performance, as measured by the state value function, that is at least as good as—and, if possible, better than—that obtained by holding any individual policy; and

(P3) Prevents chattering under measurement noise by guaranteeing a robustness margin $\varepsilon > 0$ such that for every measurement noise m satisfying $|m| \leq \varepsilon$, the switching mechanism prevents rapid switching (chattering).

A straightforward switching logic that selects the value of q corresponding to the highest state value function V_q for the current state z solves (P1) and (P2); however, it may not be robust against measurement noise as required in (P3), as illustrated in the following example.

Example 1 (Stabilizing two disconnected points on a line)**.** *We consider a system evolving on a line with the state $z \in \mathcal{Z} \subset \mathbb{R}$ and dynamics $\dot{z} = u$, where $u \in [-1, 1]$ is the control input. The problem to solve consists of robustly globally asymptotically stabilizing the set $\mathcal{Z}^* := \{z_1^*, z_2^*\} := \{-1, 1\} \subset \mathcal{Z}$, which consists of two disconnected setpoints, by designing a supervisory policy to select between the two available policies, π_1 and π_2, based on the observation vector*

$$o(z + m) = \begin{bmatrix} z + m - z_1^* \\ z + m - z_2^* \end{bmatrix}, \tag{7}$$

where $m \in \mathbb{R}$ represents the measurement noise. The policies are given by

$$\pi_q(z) = z_q^* - z, \tag{8}$$

for each $z \in \mathcal{Z}$ and each value of the logic variable $q \in \{1, 2\} := \mathcal{Q}$. It can be shown that each policy in (8) globally asymptotically stabilizes one of the setpoints, namely, π_1 globally asymptotically stabilizes z_1^* and π_2 globally asymptotically stabilizes z_2^*. The value iteration algorithm is applied to find the approximate state value functions V_q for $q \in \mathcal{Q}$ subject to the reward function

$$R(z) = -c_1 |z|_{\mathcal{Z}^*}, \tag{9}$$

which has a global maximum for $z = z^*$, where $c_1 \in \mathbb{R}_{>0}$ is a constant, discount factor $\gamma = 0.9$, sampling time of $\Delta t = 0.05$ seconds, and a horizon of 100 time steps.

The supervisory policy $Q^* : \mathcal{Z} \to \mathcal{Q}$ that maps the state z to the logic variable q is given by

$$Q^*(z) := \{q \in \mathcal{Q} : V_q(z) = \max_{\bar{q} \in \mathcal{Q}} V_{\bar{q}}(z)\} \tag{10}$$

namely, the value of q corresponds to the highest approximate state value function V_q for the current state z.[3] *For the state value function V_{Q^*} corresponding to deploying the supervisory policy Q^* it follows by the definition of Q^* that $V_{Q^*}(z) \geq V_q(z)$ for all $z \in \mathcal{Z}$. Figure 1 shows the approximate*

[3] For certain states, multiple maximizers may exist for the state value functions in the policy bank.

Figure 1: Left, the approximate state value function V_1, in blue, and V_2, in green, for the policies π_1 and π_2, respectively. Right, the resulting control policy by applying the supervisory policy Q^*. The setpoints $\mathcal{Z}^*$ are denoted by the red stars.

state value function V_1 and V_2 for the policies π_1 and π_2, respectively, and the resulting control policy by applying the supervisory policy Q^*. Figure 1 shows that in the region $z \in [z_1^*, z_2^*]$, the supervisory policy Q^* selects the logic variable q corresponding to the closest setpoint: policy π_1 ($q = 1$) when $z \leq 0$ to stabilize z_1^*, and policy π_2 ($q = 2$) when $z > 0$ to push solutions towards z_2^*. Conversely, in the regions $z \in [-3, -1.6) \cup (1.6, 3]$, the supervisory policy Q^* opts for opposing policies: policy π_2 ($q = 2$) when $z < -1.6$, and policy π_1 ($q = 1$) when $z > 1.6^*$. This exploitative selection can be attributed to the opposing policies in these regions yielding a larger control input than the corresponding policies while maintaining the same sign of the control input. Specifically, $\pi_2(z) > \pi_1(z)$ for all $z \in [-3, -1.6)$ and $\pi_1(z) < \pi_2(z)$ for all $z \in (1.6, 3]$, as can be seen in Figure 1. Hence, solutions evolve towards z^* quicker under the exploitative selection.[4]

Figure 1 shows that the resulting control policy by applying the supervisory policy Q^* is piecewise continuous. In particular, the resulting control policy changes its decision for a small change in the state z near $z_c := 0$, referred to as a critical state. Recall that $\dot{z} = u$, hence near this critical state, $\dot{z} > 0$ for $z \in (z_c, z_2^*)$ and $\dot{z} < 0$ for $z \in (z_1^*, z_c)$. To highlight the issue near z_c, suppose the system is in the region $z \in (z_c - \varepsilon, z_c)$, where $\varepsilon > 0$. Without measurement noise, the supervisory policy Q^* selects policy π_1 as the system is in the subset of the region $z \in (z_1^*, z_c)$. However, with a small perturbation $m = \varepsilon$, the measured state is in the region $z + m \in (z_c, z_c + \varepsilon)$, placing it in the subset of the region $z \in (z_c, z_2^*)$ and causing the supervisory policy Q^* to select policy π_2. At the next sampling interval, that is, when the supervisory policy Q^* makes its next decision, the system moves to the region $z \in (z_c, z_c + \varepsilon)$ due to the previous selection of policy π_2. This time, with a perturbation $m = -\varepsilon$, the measured state is $z + m \in (z_c - \varepsilon, z_c)$, resulting in the supervisory policy Q^* selecting policy π_1 and pushing the system back to into the region $(z_c - \varepsilon, z_c)$. Repetition of this pattern causes the system to chatter around the critical state z_c.[5] *Figure 2b illustrates this chattering behavior. In Figure 2, the solutions of the closed-loop system using the supervisory policy Q^* are shown for various initial conditions in the presence of the measurement noise signal given by*

$$m(t) = \varepsilon m_{\text{sgn}}(t), \tag{11}$$

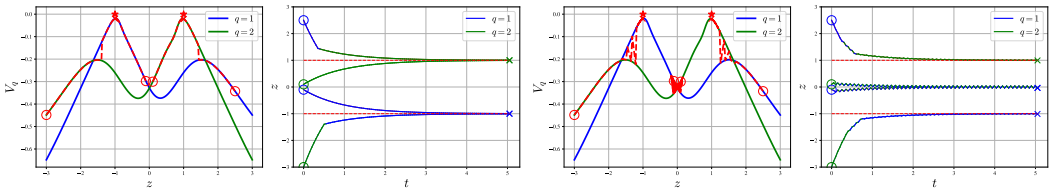
where $\varepsilon \in \mathbb{R}_{\geq 0}$ is the magnitude of the measurement noise and m_{sgn} is a function that changes its sign at every sampling time interval $\Delta t = 0.05$ and is given by

$$m_{\text{sgn}}(t) = \text{sgn}\left(\cos\left(\frac{\pi t}{\Delta t}\right)\right) \quad \forall t \geq 0, \tag{12}$$

where sgn *is the signum function. Figure 2b shows that the solutions starting away from z_c converge to the set $\mathcal{Z}^*$. However, for the solutions starting near z_c, the measurement noise (11) causes solutions to chatter around z_c and prevents convergence to the set $\mathcal{Z}^*$. Figure 2 shows that the supervisory policy Q^* indeed causes the chattering behavior due to the applied measurement noise as the supervisory policy Q^* switches its decision from $q = 1$ to $q = 2$, and vice versa, for each sampling time interval near $z = z_c$.*

[4] The supervisory policy Q^* is not necessarily the optimal switching strategy; however, is better or equal to a non-switching strategy.

[5] The decision of the supervisory policy Q^* changes near $z \in \{-1.6, 1.6\}$ as well; however, as both policies have an equal sign for these points, chattering does not occur at these points.



(a) Noise magnitude $\varepsilon = 0$.

(b) Noise magnitude $\varepsilon = 0.3$.

Figure 2: The solutions of the closed-loop system using the supervisory policy Q^* plotted over the approximate state value function V_1, in blue, and V_2, in green, for the policies π_1 and π_2, respectively, and over time under the measurement noise signal (11) of magnitude $\varepsilon \in \{0, 0.3\}$, for Example 1. The solutions plotted over the state value functions are displayed by the dashed red lines with initial conditions denoted by the circles and terminal conditions by the crosses. The setpoints $\mathcal{Z}^*$ are denoted by the red stars.

To address the issue of chattering in the supervisory policy, we propose a state value function-based switching approach that ensures the resulting hybrid closed-loop system is robust to measurement noise and robustly globally asymptotically stabilizes a compact set.

4 Hybrid Supervisors

In this section, we first define a class of systems and establish state value function-based conditions that characterize *critical areas*, where multiple policies appear equally "good", and small perturbations can trigger undesired chattering behavior and performance degradation. Using these conditions, we then propose a hybrid supervisor that deploys a timer-based mechanism to predict and enforce a dwell period between policy switches.

4.1 Class of Systems

The focus is on systems where, in certain regions of the state space, multiple policies are locally equally optimal choices. However, small perturbations in observations within these regions can lead to rapid switching between decisions, resulting in undesired chattering behavior and performance degradation. These regions are referred to as *critical areas*. From a state value function-based perspective, critical areas occur in regions of the state space where small perturbations in observations can lead to rapid switching between decisions, causing chattering and performance degradation. Given a policy bank Π with continuous policies $\pi_q : \mathcal{Z} \to \mathcal{U}$ for $q \in \mathcal{Q}$, continuously differentiable approximate state value functions $V_q : \mathcal{Z} \to \mathbb{R}$ of the form (5), and continuous-time dynamics as in (1), where $\mathcal{Z}$ is closed, critical areas are identified by the following conditions:

($\circ$) for each $z \in \mathcal{M}^* \subset \mathcal{Z}$, there exist $q, p \in Q^*(z)$, $q \neq p$, such that

$$V_q(z) = V_p(z), \tag{13}$$

$$\langle \nabla V_q(z), f(z, \pi_q(z)) \rangle > 0 \text{ and } \langle \nabla V_p(z), f(z, \pi_p(z)) \rangle > 0, \tag{14}$$

$$\langle \nabla V_q(z), f(z, \pi_p(z)) \rangle < 0 \text{ and } \langle \nabla V_p(z), f(z, \pi_q(z)) \rangle < 0. \tag{15}$$

Therefore, the set $\mathcal{M}^*$ is defined as:

$$\begin{aligned} \mathcal{M}^* := \{ z \in \mathcal{Z} : & \exists q \in Q^*(z), p \in Q^*(z) \setminus \{q\} : V_p(z) = V_q(z), \\ & \langle \nabla V_q(z), f(z, \pi_q(z)) \rangle > 0, \langle \nabla V_p(z), f(z, \pi_p(z)) \rangle > 0, \\ & \langle \nabla V_q(z), f(z, \pi_p(z)) \rangle < 0, \langle \nabla V_p(z), f(z, \pi_q(z)) \rangle < 0 \}. \end{aligned} \tag{16}$$

The conditions in ($\circ$) can be interpreted as follows. Condition (13) identifies states z where multiple policies π_q and π_p are equally optimal, allowing a supervisory policy to select either q or q'.

Condition (14) indicates that the state value functions V_q and V_p increase locally when following their respective policies π_q and π_p . Conversely, condition (15) shows that the state value functions decrease when following the opposing policies π_p and π_q , respectively. Together, these conditions are used to define the following partitions of the state space $\mathcal{Z}$ near the critical areas $\mathcal{M}^*$.

Lemma 1. *Under the conditions in* $(\circ)$*, there exists* $\delta > 0$ *such that the set* $\mathcal{M}^* + \delta\mathbb{B}$ *can be written as*

$$\bigcup_{q \in \mathcal{Q}'} \mathcal{Z}_q \subset \mathcal{M}^* + \delta\mathbb{B}, \tag{17}$$

where $\mathcal{Q}' := Q^*(\mathcal{M}^*) = \bigcup_{z \in \mathcal{M}^*} Q^*(z) \subset \mathcal{Q}$ *and, for each* $q \in \mathcal{Q}'$*, the partition*

$$\mathcal{Z}_q := \bigcap_{p \in \mathcal{Q}' \setminus \{q\}} \{z \in \mathcal{Z} \cap (\mathcal{M}^* + \delta\mathbb{B}) : V_q(z) - V_p(z) \geq 0\} \tag{18}$$

satisfies

(A1) $\mathcal{Z}_q$ *is closed for each* $q \in \mathcal{Q}'$*;*

(A2) $V_q(z) > V_p(z)$ *for all* $z \in \operatorname{int}(\mathcal{Z}_q)$ *and* $p \in \mathcal{Q}' \setminus \{q\}$*; and*

(A3) $\operatorname{int}(\mathcal{Z}_q) \cap \operatorname{int}(\mathcal{Z}_p) = \varnothing$ *for all* $q, p \in \mathcal{Q}'$*,* $p \neq q$*.*

The proof of Lemma 1 can be found in the supplementary material.

By Lemma 1, there exists an arbitrarily small $\delta > 0$ such that the neighborhood $\mathcal{M}^* + \delta\mathbb{B}$ is partitioned into regions $\mathcal{Z}_q$ with disjoint interiors, where each $\mathcal{Z}_q$ corresponds to a region where V_q is strictly larger than V_p for all $q, p \in \mathcal{Q}'$, $p \neq q$. Suppose, without loss of generality, that $z \in \mathcal{Z}_q$ for some $q \in \mathcal{Q}'$. Then, due to the properties in Lemma 1, every point $z \in \mathcal{M}^* + \delta\mathbb{B}$ is δ close to some boundary between $\mathcal{Z}_q$ and $\mathcal{Z}_p$. Therefore, there exists a perturbation $m \in \delta\mathbb{B}$, for which the perturbed state $z + m$ belongs to $\mathcal{Z}_p$. Consequently, even though z initially lies in $\mathcal{Z}_q$, the perturbation leads to the selection of policy π_p. Since $z + m$ is in $\mathcal{M}^* + \delta\mathbb{B}$, conditions (14) and (15) ensure that if p is chosen in $\mathcal{Z}_q$, V_p increases while V_q decreases, effectively making $\mathcal{M}^*$ attractive. That is, the system is "pulled" back toward the critical area. Subsequent perturbations can then induce a switch back to q, leading to chattering between policies near the critical area.

4.2 Timer-based Hybrid Supervisor

The hybrid supervisor employs a timer to prevent chattering near critical areas. This timer-based approach sets a dwell time parameter based on the state value functions and a noise-free model of the system dynamics, ensuring a minimum waiting period before another policy switch is allowed.

The hybrid closed-loop system $\mathcal{H} = (C, F, D, G)$ models the continuous evolution (flow) and discrete updates (jumps) of its state, which consists of the system state z of (1), a timer $\tau \in \mathbb{R}_{\geq 0}$, the logic variable $q \in \{1, 2, \ldots, N\} =: \mathcal{Q}$, where $N \in \mathbb{N}_{>0}$ is the number of policies in the policy bank, and an adjustable dwell time parameter $\delta_d \in \mathbb{R}_{>0}$. The adjustable dwell time parameter δ_d dictates the amount of time that needs to elapse before the applied policy can be changed, namely, to change q. The state is defined as $x = (z, \tau, q, \delta_d) \in \mathcal{X} := \mathcal{Z} \times \mathbb{R}_{\geq 0} \times \mathcal{Q} \times \mathbb{R}_{\geq 0}$. The flow map F governs the continuous evolution of each state component, the flow set C specifies the conditions under which the state components can flow, the jump map G governs the discrete updates, and the jump set D defines the conditions under which the state components can jump.

During flows, the state z evolves according to its dynamics (1) under the selected policy π_q from the policy bank. Furthermore, the policy decision stored in q and the adjustable dwell time parameter δ_d do not change during flows. On the other hand, the timer τ evolves linearly with a constant rate of one so as to count ordinary time. The *flow map* F that captures this behavior is given by

$$\begin{bmatrix} \dot{z} \\ \dot{\tau} \\ \dot{q} \\ \dot{\delta}_d \end{bmatrix} = F(x) := \begin{bmatrix} f(z, \pi_q(z)) \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad x \in C. \tag{19}$$

The system flows whenever the timer τ is less than or equal to the adjustable dwell time parameter δ_d, or when the logic variable q corresponds to the highest state value function for the current value of the state z, leading to the *flow set* $C := C_0 \cup C_1$, where

$$C_0 := \{x \in \mathcal{X} : \tau \leq \delta_d\}, \tag{20}$$

and

$$C_1 := \{x \in \mathcal{X} : q \in Q^*(z)\}, \tag{21}$$

where Q^* is given by (10). The *jump map* G is given by

$$\begin{bmatrix} z^+ \\ \tau^+ \\ q^+ \\ \delta_d^+ \end{bmatrix} \in G(x) := \begin{bmatrix} z \\ 0 \\ \left\{ \begin{bmatrix} q' \\ \min\left(\mathcal{T}(z,q') \cup \{\bar{\delta}_d\}\right) \end{bmatrix} : q' \in Q^*(z) \right\} \end{bmatrix} \quad x \in D, \tag{22}$$

where

$$\mathcal{T}(z,q) := \Big\{ \eta \in [0, \bar{\delta}_d] : \frac{\max\limits_{\bar{q} \in (\mathcal{Q} \setminus \{q\})} V_{\bar{q}}(\chi(\eta)) - V_q(\chi(\eta))}{|V_q(\chi(\eta))| + \epsilon} \geq \mu, \\ \text{where } \dot{\chi} = f(\chi, \pi_q(\chi)), \quad \chi(0) = z \Big\}, \tag{23}$$

where $\epsilon \in \mathbb{R}_{>0}$ is a very small constant (e.g., $\epsilon \ll 1$) used to avoid division by zero, $\mu \in \mathbb{R}_{>0}$ is a constant, and $\bar{\delta}_d \in \mathbb{R}_{>0}$ is the maximum value of the dwell time parameter. The set-valued map $\mathcal{T} : \mathcal{Z} \rightrightarrows [0, \bar{\delta}_d]$ gathers all time horizons up to $\bar{\delta}_d$ at which a *significant* relative improvement (as determined by μ) over the *current* optimal policy $q'(z)$ is predicted. To clarify the construction of the ratio in (23), suppose that, for some $q \in \mathcal{Q}$ and for some $\eta \in [0, \bar{\delta}_d]$, we have $V_q(\chi(\eta)) > 0$. Then the ratio in (23) can be written as

$$V_{\bar{q}}(\chi(\eta)) \geq \mu(|V_q(\chi(\eta))| + \epsilon) + V_q(\chi(\eta)) > (1+\mu)V_q(\chi(\eta)), \tag{24}$$

where $\bar{q} \in \mathcal{Q} \setminus \{q\}$. This inequality indicates that the policy with index $\bar{q}$ yields a return that is more than $1+\mu$ times the return of policy q at η seconds in the future. Next, suppose that, for some $q \in \mathcal{Q}$ and for some $\eta \in [0, \bar{\delta}_d]$, we have $V_q(\chi(\eta)) < 0$. In this case, the ratio in (23) can be written as

$$V_{\bar{q}}(\chi(\eta)) - V_q(\chi(\eta)) \geq \mu(|V_q(\chi(\eta))| + \epsilon) > \mu|V_q(\chi(\eta))|, \tag{25}$$

where again $\bar{q} \in \mathcal{Q} \setminus \{q\}$. Here, the inequality shows that the difference in return between the policy with index $\bar{q}$ and the policy with index q is greater than μ times the return of policy q at η seconds in the future. Notice that the absolute value in the term $|V_q(\chi(\eta))|$ is essential. Suppose again that, for some $q \in \mathcal{Q}$ and for some $\eta \in [0, \bar{\delta}_d]$, we have $V_q(\chi(\eta)) < \epsilon$. In this case, if we were to omit the absolute value, the ratio in (23) would be written as

$$V_{\bar{q}}(\chi(\eta)) \leq \mu\Big(V_q(\chi(\eta)) + \epsilon\Big) + V_q(\chi(\eta)) = (\mu+1)V_q(\chi(\eta)) + \mu\epsilon. \tag{26}$$

Since $V_q(\chi(\eta)) < \epsilon$, the sum $V_q(\chi(\eta)) + \epsilon$ is negative, which makes the right-hand side of (26) negative as well. This means that the inequality would hold even if the return of policy $\bar{q}$ is actually *worse* than $1+\mu$ times the return of policy q at η seconds in the future. By including the absolute value, we ensure a meaningful comparison of returns regardless of the sign of $V_q(\chi(\eta))$.

The constant $\epsilon > 0$ ensures the ratio is well-defined even when $V_q(\chi(\eta))$ is close to zero. In (23), the trajectory $\chi(\eta)$ with $\chi(0) = z$ is a solution of $\dot{\chi} = f(\chi, \pi_q(\chi))$ for $\eta \in [0, \bar{\delta}_d]$. For each alternative policy $\pi_{\bar{q}}$ with $\bar{q} \in Q \setminus \{q\}$, we compare $V_q(\chi(\eta))$ with $V_{\bar{q}}(\chi(\eta))$. Whenever the resulting ratio meets or exceeds the threshold μ, the corresponding η is included in $\mathcal{T}(z,q)$. In (22), it can be seen that, at jumps, the state z remains unchanged, and the logic variable q is updated to correspond to the index of one of the state value functions with largest value, that is q is reset to a point in $Q^*(z)$, the

timer τ is reset to 0, and the adjustable dwell time parameter δ_d is reset to the minimum between the smallest time horizon in $\mathcal{T}(z, q)$ or to the maximum dwell time $\bar{\delta}_d$. If the set $\mathcal{T}(z, q)$ is empty, which is possible when no significant improvement is observed over the current optimal policy within the time horizon δ_d, then δ_d is reset to $\bar{\delta}_d$.

The system jumps whenever the state is in the *jump set* D, which is defined as

$$D := \{x \in \mathcal{X} : \tau \geq \delta_d \, , q \in \mathcal{Q} \setminus Q^*(z)\}. \tag{27}$$

In (27), it can be seen that the jump in (22) occurs whenever the timer is greater than or equal to the adjustable dwell time parameter δ_d and the logic variable q does not correspond to a state value function with the largest value.

Next, the key properties of the hybrid closed-loop system $\mathcal{H}$ in (19)-(27) are discussed. The first property, namely the hybrid basic conditions, is a set of mild conditions on the data (C, F, D, G) of the hybrid closed-loop system $\mathcal{H}$. These conditions are required to ensure that asymptotically stable compact sets of the hybrid closed-loop system $\mathcal{H}$ are robust against disturbances.

Proposition 1. *The hybrid closed-loop system $\mathcal{H}$ in (19)-(27) satisfies the hybrid basic conditions if, for each $q \in \mathcal{Q}$, the approximate state value function $V_q : \mathcal{Z} \to \mathbb{R}$ in (5) is continuous and the policy $\pi_q : \mathcal{Z} \to \mathcal{U}$ and $f : \mathcal{Z} \times \mathcal{U} \to \mathcal{Z}$ are Lipschitz continuous.*

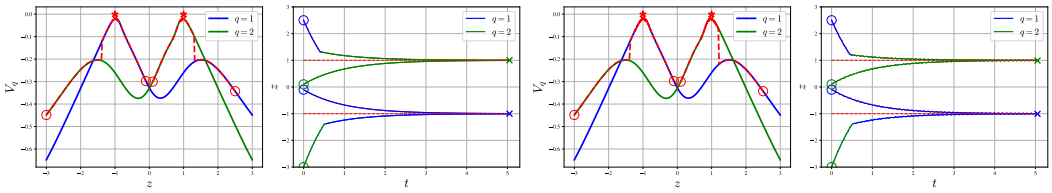
Proof Sketch. We show that the hybrid system $\mathcal{H}$ satisfies the required properties. First, both the flow set C and the jump set D are closed subsets of $\mathcal{X}$. Next, the flow map F is single-valued and constructed from Lipschitz continuous policies π_q and dynamics f, which implies its continuity on C. This continuity ensures that F is outer semicontinuous, locally bounded and that its values are convex. For the jump map G, the analysis is performed for each state component. The z and τ components are continuous. The q component is outer semicontinuous as Q^* is shown to be outer semicontinuous. The δ_d component is determined by a set-valued map given as the inverse image of a closed set under a continuous function and is thus outer semicontinuous. Together with the appropriate boundedness properties, these facts establish that $\mathcal{H}$ meets the desired conditions. For the detailed proof, see the supplementary material. □

The following proposition shows that under mild continuity and Lipschitz conditions, the hybrid closed-loop system $\mathcal{H}$ in (19)-(27) has nontrivial solutions for every initial state in the flow or jump set. Without this guarantee, the algorithm could get stuck, preventing further progress.

Proposition 2. *Suppose, for each $q \in \mathcal{Q}$, the approximate state value function $V_q : \mathcal{Z} \to \mathbb{R}$ in (5) is continuous and the policy $\pi_q : \mathcal{Z} \to \mathcal{U}$ and $f : \mathcal{Z} \times \mathcal{U} \to \mathcal{Z}$ are Lipschitz continuous. Then, there exists a nontrivial solution x to the hybrid closed-loop system $\mathcal{H}$ in (19)-(27) for each $x_\circ \in C \cup D$ with $x(0, 0) = x_\circ$. Furthermore, every maximal solution x to $\mathcal{H}$ is complete, not Zeno, and every bounded solution x to $\mathcal{H}$ has jump times that are uniformly lower bounded by a positive constant, that is, for each bounded solution x to $\mathcal{H}$ there exists $\gamma > 0$ such that $t_{j+1} - t_j \geq \gamma$ for all $j \geq 1$, where t_j denotes the time at jump j.*

Proof Sketch. We show that every initial state yields a well-behaved solution. First, during continuous evolution, the state either lies in an open region when $\tau < \delta_d$, or, if on the boundary when $\tau \geq \delta_d$ with $q \in Q^*(z)$, the allowed directions, described by the tangent cone, still contain the flow map F. This ensures that the system can always flow. When a jump occurs, the jump map resets the timer and updates q to an optimal value while keeping z unchanged so that the state is moved back into a region where continuous evolution can resume and jumps cannot occur in rapid succession. Moreover, under the global Lipschitz condition on F, solutions can be extended for all time, ruling out issues like Zeno behavior. For the detailed proof, see the supplementary material. □

Theorem 1. *Suppose, for each $q \in \mathcal{Q}$, the approximate state value function $V_q : \mathcal{Z} \to \mathbb{R}$ in (5) is continuous and the policy $\pi_q : \mathcal{Z} \to \mathcal{U}$ and $f : \mathcal{Z} \times \mathcal{U} \to \mathcal{Z}$ are Lipschitz continuous. Furthermore, each policy $\pi_q \in \Pi$ globally asymptotically stabilizes a compact set $\mathcal{Z}_q^*$, where $\mathcal{Z}^* := \bigcup_{q \in \mathcal{Q}} \mathcal{Z}_q^*$,*



(a) Noise magnitude $\varepsilon = 0$.

(b) Noise magnitude $\varepsilon = 0.3$.

Figure 3: The solutions of the hybrid closed-loop system $\mathcal{H}$ in (19)-(27), where $\mu = 0.1$ and $\bar{\delta}_d = 0.5$, over the approximate state value function V_1, in blue, and V_2, in green, for the policies π_1 and π_2, respectively, and over time under the measurement noise signal (11) of magnitude $\varepsilon \in \{0, 0.3\}$, for Example 1. The solutions plotted over the state value functions are displayed by the dashed red lines with initial conditions denoted by the circles and terminal conditions by the crosses. The setpoints $\mathcal{Z}^*$ are denoted by the red stars.

the reward function in (4) *has a global maximum at* $\mathcal{Z}^*$*, and the reward function is strictly decreasing as the distance from* $\mathcal{Z}^*$ *increases. Then, the hybrid closed-loop system* $\mathcal{H}$ *in* (19)-(27) *solves Problem* $(\star)$.

Proof Sketch. We construct a hybrid Lyapunov function based on the state value functions, which is positive outside the desired set and decreases strictly along both flows and jumps. During flows, the continuous dynamics under each policy drive the state toward the target set $\mathcal{Z}^*$, while at jumps, the switching mechanism selects a policy that maximizes the value function. In addition, the jump mechanism enforces a positive dwell time, ensuring that switches cannot occur too rapidly. Together, these properties ensure that the hybrid closed-loop system maintains or improves performance and prevents chattering. For the detailed proof, see the supplementary material. □

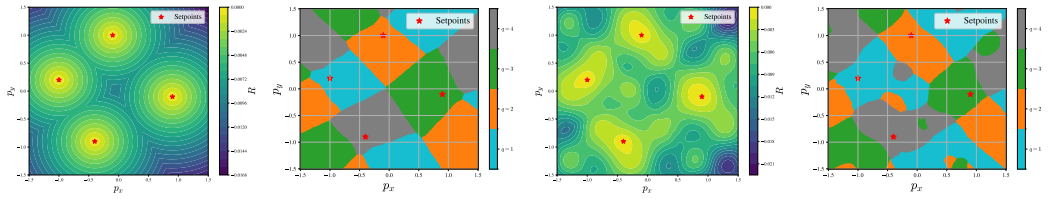
Example 2 (Stabilizing two disconnected points on a line, revisited)**.** *The solutions of the hybrid closed-loop system* $\mathcal{H}$ *in* (19)-(27) *are shown in Figure 3 for various initial conditions in the presence of the measurement noise signal* (11) *of magnitude* $\varepsilon \in \{0, 0.3\}$*. Furthermore, the parameters in the jump set* (27) *are chosen as* $\mu = 0.1$ *and* $\bar{\delta}_d = 0.5$.[6] *Figure 3b shows that the hybrid closed-loop system* $\mathcal{H}$ *in* (19)-(27) *is robust against the perturbation that caused the switching logic* (10) *in Example 1 Figure 2b to chatter. Furthermore, the rapid policy switching that occurs near* $z \in \{-1.5, 1.5\}$ *in Figure 2b is also prevented by the hybrid switching logic.*

4.3 Design Considerations

The timer-based approach relies on predicting the system state using a dynamic model, which can be computationally expensive. The maximum dwell time $\bar{\delta}_d$ in (22) sets the upper limit for the prediction horizon. If $\bar{\delta}_d$ is set too low, the prediction horizon may be insufficient for the system to exit critical regions, thereby reducing robustness; if it is set too high, the increased computational cost—and, when combined with a high threshold parameter μ in (23), the potential for prolonged adherence to a suboptimal policy—may degrade performance. The parameter μ serves as a safeguard against rapid switching when the state value functions of competing policies are nearly equal. Together, $\bar{\delta}_d$ and μ balance the trade-off between computational efficiency, exploitative policy switching, and robustness.

In practice, the parameters $\bar{\delta}_d$ and μ can be determined from prior knowledge of the system dynamics or tuned iteratively based on observed chattering behavior. Alternatively, these parameters may be made state-dependent and integrated into the learning process as part of the overall control strategy.

[6] The design/choice of μ and $\bar{\delta}_d$ is discussed in Section 4.3.



(a) Without periodic variations.

(b) With periodic variations.

Figure 4: Visualization of the reward function 28 (located left in each sub-figure) and Q^* (located right in each sub-figure) without and with the periodic variations. The setpoints $\mathcal{Z}^*$ are denoted by the red stars.

5 Application

We consider a system evolving on a plane with the state $z = (z_x, z_y) \in \mathcal{Z} \subset \mathbb{R}^2$, with $z_x \in \mathbb{R}$ and $z_y \in \mathbb{R}$ being the coordinates along the $x-$ and $y-$axes, respectively, and dynamics $\dot{z} = u$, where $u \in [-1, 1]^2$ is the control input. The problem to solve consists of robustly globally asymptotically stabilizing the set $\mathcal{Z}^* := \{z_1^*, z_2^*, z_3^*, z_4^*\} := \{(-1, 0.2), (-0.1, 1), (0.9, -0.1), (-0.4, -0.9)\} \subset \mathcal{Z}$, which consists of four disconnected setpoints, by designing a supervisory policy to select between the four available policies, π_1, π_2, π_3 and π_4, based on the observation vector $o(z + m) = z + m$, where $m \in \mathbb{R}^2$ represents the measurement noise. The policies are given by (8) for each $z \in \mathcal{Z}$ and each value of the logic variable $q \in \{1, 2, 3, 4\} := \mathcal{Q}$. It can be shown that each policy globally asymptotically stabilizes one of the setpoints, namely, π_1 globally asymptotically stabilizes z_1^*, π_2 globally asymptotically stabilizes z_2^*, π_3 globally asymptotically stabilizes z_3^*, and π_4 globally asymptotically stabilizes z_4^*. The value iteration algorithm is applied to find the approximate state value functions V_q for $q \in \mathcal{Q}$ subject to the reward function

$$R(z) = -c_1 |z|_{\mathcal{Z}^*} \left(1 + \frac{1}{2} \sin(c_2 z_x) \cos(c_3 z_y)\right), \tag{28}$$

which has a global maximum for $z = z^*$, where $c_1, c_2, c_3 \in \mathbb{R}_{>0}$ are constants, discount factor $\gamma = 0.9$, sampling time of $\Delta t = 0.025$ seconds, and a horizon of 100 time steps. The term $1 + \frac{1}{2} \sin(c_2 z_x) \cos(c_3 z_y)$ introduces periodic variations in the reward function, creating local minima and maxima across the state space. As a result, following the shortest Euclidean path to a setpoint is not necessarily optimal—certain indirect trajectories may yield a higher long-term return. This makes determining the optimal regions for each policy—the regions where a given policy has the highest corresponding state value function, namely $\mathcal{Q}^*$ (10)—nontrivial. Consequently, this leads to more complex switching boundaries and decision regions, as shown in Figure 4.

Figure 5 compares the closed-loop system solutions under the two different supervisors. The first row displays the supervisory policy Q^*, given by (10), while the second displays the timer-based hybrid supervisor. The timer-based hybrid supervisor, introduced in Section 4.2, is given by (19)-(27) with parameters $\bar{\delta}_d = 0.5$ and $\mu = 0.15$. The trajectories are plotted over the optimal policy regions $\mathcal{Q}^*$ and over time under the influence of a measurement noise signal, given by

$$m(t) = \begin{bmatrix} \varepsilon \\ -\varepsilon \end{bmatrix} m_{\text{sgn}}(t), \tag{29}$$

where the noise magnitude is $\varepsilon \in \{0, 0.1\}$. The first row of Figure 5 shows that the supervisory policy Q^* is highly sensitive to measurement noise, leading to chattering and, in some cases, failure to reach the target set $\mathcal{Z}^*$. In fact, even one initial condition exhibits chattering in the absence of noise. In contrast, the second row demonstrates that the hybrid supervisor successfully mitigates chattering and consistently guides the system to $\mathcal{Z}^*$ for all considered initial conditions, even under the same measurement noise that caused instability in Q^*.

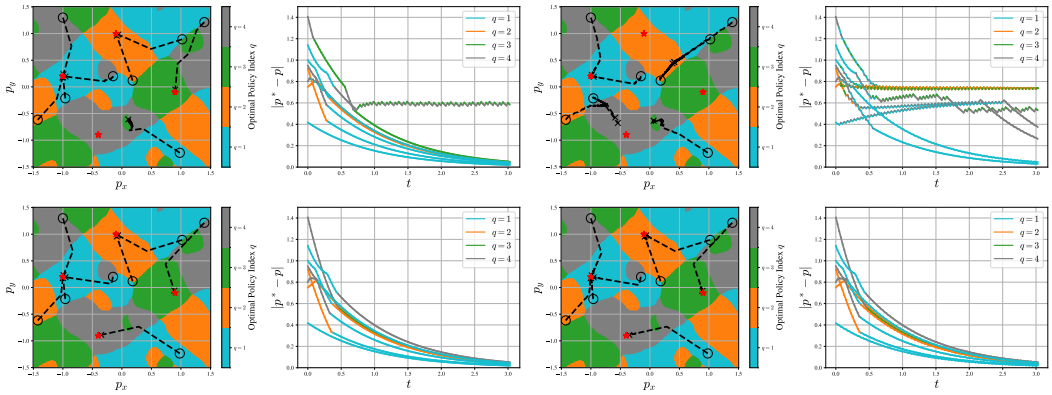


Figure 5: The solutions of the (hybrid) closed-loop system using the supervisory policy Q^* (first row) and the timer-based hybrid supervisor from Section 4.2 (second row), plotted over Q^*, and over time under the measurement noise signal (29) of magnitude $\varepsilon \in \{0, 0.1\}$. The solutions plotted over the optimal regions for each policy are displayed by the black dashed lines with initial conditions denoted by the circles and terminal conditions by the crosses. The solutions plotted over time illustrate at what time the switches between policies occur. The setpoints $\mathcal{Z}^*$ are denoted by the red stars.

Remark. *The problem setup in this section generalizes to a wide range of real-world decision-making problems where multiple strategies are available, but selecting the best one at any moment is challenging due to limited information and long-term trade-offs. Furthermore, adversaries or external agents can deliberately influence the environment to induce frequent or premature switching. In such cases, an agent may be misled into abandoning a good strategy too soon or committing to a suboptimal one, leading to inefficient or unintended behavior. An example is video game strategy selection, where a player or AI agent must decide when to commit to a specific tactic—such as aggressive attacks, defensive positioning, or resource gathering. While frequent switching may seem advantageous in response to short-term rewards, it can lead to inefficiencies or poor overall performance. Moreover, in competitive settings, a clever opponent may manipulate the game state to bait the player into unnecessary switching, exploiting hesitation or uncertainty to gain an advantage. This mirrors real-world decision-making problems where adversarial environments or strategic opponents introduce uncertainty, making robust strategy selection crucial.*

6 Conclusion

This paper presents a novel timer-based hybrid supervisor that leverages state value functions for robust switching among multiple policies. The supervisor predicts and enforces a minimum dwell time between policy switches, thereby preventing chattering even under bounded measurement noise. It also ensures that overall performance is maintained or improved relative to any fixed policy while preserving the asymptotic stability of the individual policies. Sufficient conditions are presented for non-Zeno behavior and robust asymptotic stability of the hybrid closed-loop system. Numerical simulations on representative decision-making problems demonstrate that the supervisor effectively mitigates rapid switching and drives the system toward the desired target set even under noisy conditions, where a conventional switching strategy fails.

References

Tansu Alpcan and Tamer Baar. *Network Security: A Decision and Game-Theoretic Approach*. Cambridge University Press, USA, 1st edition, 2010. ISBN 0521119324.

Söhnke M Bartram, Jürgen Branke, Giuliano De Rossi, and Mehrshad Motahari. Machine learning for active portfolio management. *Journal of Financial Data Science*, 3(3):9–30, 2021.

Jan de Priester, Ricardo G. Sanfelice, and Nathan van de Wouw. Hysteresis-based rl: Robustifying reinforcement learning-based control policies via hybrid control. In *2022 American Control Conference (ACC)*, pp. 2663–2668, 2022. DOI: 10.23919/ACC53348.2022.9867627.

Jan de Priester, Zachary Bell, Prashant Ganesh, and Ricardo Sanfelice. MultiHyRL: Robust hybrid RL for obstacle avoidance against adversarial attacks on the observation space. *Reinforcement Learning Journal*, 4:2017–2040, 2024.

Rafal Goebel, Ricardo G. Sanfelice, and Andrew R. Teel. *Hybrid Dynamical Systems: Modeling, Stability, and Robustness*. Princeton University Press, 2012. ISBN 9780691153896. URL http://www.jstor.org/stable/j.ctt7s02z.

Jemin Hwangbo, Joonho Lee, Alexey Dosovitskiy, Dario Bellicoso, Vassilios Tsounis, Vladlen Koltun, and Marco Hutter. Learning agile and dynamic motor skills for legged robots. *Science Robotics*, 4(26):eaau5872, 2019. DOI: 10.1126/scirobotics.aau5872. URL https://www.science.org/doi/abs/10.1126/scirobotics.aau5872.

Hassan K Khalil. *Nonlinear systems; 3rd ed.* Prentice-Hall, Upper Saddle River, NJ, 2002. URL https://cds.cern.ch/record/1173048. The book can be consulted by contacting: PH-AID: Wallet, Lionel.

Wanjiku A. Makumi, Max L. Greene, Zachary I. Bell, Scott Nivison, Rushikesh Kamalapurkar, and Warren E. Dixon. Hierarchical reinforcement learning-based supervisory control of unknown nonlinear systems. *IFAC-PapersOnLine*, 56(2):6871–6876, 2023. ISSN 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2023.10.485. URL https://www.sciencedirect.com/science/article/pii/S2405896323008522. 22nd IFAC World Congress.

C. G. Mayhew, R. G. Sanfelice, and A. R. Teel. Quaternion-based hybrid controller for robust global attitude tracking. *IEEE Transactions on Automatic Control*, 56(11):2555–2566, November 2011. DOI: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5701762. URL https://hybrid.soe.ucsc.edu/files/preprints/50.pdf.

Christophe Prieur, Rafal Goebel, and Andrew R. Teel. Hybrid feedback control and robust stabilization of nonlinear systems. *IEEE Transactions on Automatic Control*, 52(11):2103–2117, 2007. DOI: 10.1109/TAC.2007.908320.

Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., USA, 1st edition, 1994. ISBN 0471619779.

R.T. Rockafellar, M. Wets, and R.J.B. Wets. *Variational Analysis*. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg, 2009. ISBN 9783540627722. URL https://books.google.com/books?id=w-NdOE5fD8AC.

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

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017. URL http://arxiv.org/abs/1707.06347.

Richard S. Sutton, Doina Precup, and Satinder Singh. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1):181–211, 1999. ISSN 0004-3702. DOI: https://doi.org/10.1016/S0004-3702(99)00052-1. URL https://www.sciencedirect.com/science/article/pii/S0004370299000521.

Georgios N. Yannakakis and Julian Togelius. *Artificial Intelligence and Games*. Springer Publishing Company, Incorporated, 1st edition, 2018. ISBN 3319635182.

Supplementary Materials

The following content was not necessarily subject to peer review.

Proof of Lemma 1. For each fixed $q \in \mathcal{Q}'$, consider any $p \in \mathcal{Q}' \setminus \{q\}$. Since both V_q and V_p are continuous, the function

$$g_p(z) := V_q(z) - V_p(z) \quad \forall z \in \mathcal{Z} \tag{30}$$

is continuous. Moreover, the set $\mathcal{Z}$ is closed, and since $\mathbb{B}$ is closed, the set $\mathcal{M}^* + \delta\mathbb{B}$ is closed as well. The intersection of these two closed sets is, therefore, closed. Consequently, the set

$$S_p := \{z \in \mathcal{Z} \cap (\mathcal{M}^* + \delta\mathbb{B}) : g_p(z) \geq 0\} \tag{31}$$

is closed, as it is the preimage of the closed set $[0, \infty)$ under g_p. Since Z_q in (18) is an intersection of closed sets, it follows that $\mathcal{Z}_q$ is closed for each $q \in \mathcal{Q}'$. Thus, condition (A1) is satisfied.

Furthermore, the set $\mathcal{Z}_q$ can be written as

$$\mathcal{Z}_q := \bigcap_{p \in \mathcal{Q}' \setminus \{q\}} \{z \in \mathcal{Z} \cap (\mathcal{M}^* + \delta\mathbb{B}) : V_q(z) - V_p(z) \geq 0\} = \bigcap_{p \in \mathcal{Q}' \setminus \{q\}} S_p. \tag{32}$$

The interior of $\mathcal{Z}$ is therefore

$$\begin{aligned}
\operatorname{int}(\mathcal{Z}_q) &= \operatorname{int}\left(\bigcap_{p \in \mathcal{Q}' \setminus \{q\}} S_p\right); && \text{(33a)}\\
&= \bigcap_{p \in \mathcal{Q}' \setminus \{q\}} \operatorname{int}(S_p); && \text{(33b)}\\
&= \bigcap_{p \in \mathcal{Q}' \setminus \{q\}} \operatorname{int}(\{z \in \mathcal{Z} \cap (\mathcal{M}^* + \delta\mathbb{B}) : g_p(z) \geq 0\}); && \text{(33c)}\\
&= \bigcap_{p \in \mathcal{Q}' \setminus \{q\}} \{z \in \mathcal{Z} \cap (\mathcal{M}^* + \delta\mathbb{B}) : g_p(z) > 0\}; && \text{(33d)}\\
&= \bigcap_{p \in \mathcal{Q}' \setminus \{q\}} \{z \in \mathcal{Z} \cap (\mathcal{M}^* + \delta\mathbb{B}) : V_q(z) - V_p(z) > 0\}. && \text{(33e)}
\end{aligned}$$

Hence, for each $z \in \operatorname{int}(\mathcal{Z}_q)$, it holds that $V_q(z) > V_p(z)$, proving condition (A2).

Suppose for contradiction that there exists a point

$$z \in \operatorname{int}(\mathcal{Z}_q) \cap \operatorname{int}(\mathcal{Z}_p) \tag{34}$$

for distinct $q, p \in \mathcal{Q}'$. Then by (A2), since $z \in \operatorname{int}(\mathcal{Z}_q)$ we have

$$V_q(z) > V_p(z), \tag{35}$$

and since $z \in \operatorname{int}(\mathcal{Z}_p)$ we have

$$V_p(z) > V_q(z). \tag{36}$$

This contradiction shows that such a z cannot exist, so

$$\operatorname{int}(\mathcal{Z}_q) \cap \operatorname{int}(\mathcal{Z}_p) = \varnothing. \tag{37}$$

Thus, condition (A3) is satisfied. □

Proof of Proposition 1. According to Sanfelice (2021, Theorem 2.20), $\mathcal{H}$ satisfies the hybrid basic conditions if

(B1) C and D are closed subsets of $\mathcal{X}$;

(B2) F is outer semicontinuous and locally bounded relative to C, $C \subset \text{dom } F$, and $F(x)$ is convex for each $x \in C$;

(B3) G is outer semicontinuous and locally bounded relative to D, and $D \subset \text{dom } G$.

By definition, the flow set C in (20)-(21) and the jump set D in (27) are closed subsets of $\mathcal{X}$, satisfying condition (B1).

The flow map F in (19) is a single-valued map. By assumption, the policies π_q and state dynamics f are Lipschitz continuous for each $q \in \mathcal{Q}$. Furthermore, the policies π_q and state dynamics f are defined for each $z \in \mathcal{Z}$ by assumption. These assumptions ensure that F is continuous on $\mathcal{C}$. Consequently:

- The domain of F satisfies $C \subset \text{dom } F = \mathcal{X}$;
- The continuity of F implies that F is outer semicontinuous;
- The continuity of F also guarantees that F is locally bounded relative to C;
- Since F is a single-value map, $F(x)$ is trivially convex for each $x \in \mathcal{C}$.

Thus, condition (B2) is satisfied.

The outer semicontinuity of the jump map G follows from the fact that each state component is outer semicontinuous. Specifically, the z and τ components are trivially continuous and hence outer semicontinuous. For the q component, given $z \in \mathcal{Z}$, $Q^*(z)$ in (10) can be written as

$$Q^*(z) = \{q \in \mathcal{Q} : \nu(z, q) = 0\}, \tag{38}$$

where $\nu(z, q) := V_q(z) - \max_{\bar{q} \in \mathcal{Q}} V_{\bar{q}}(z)$. For each $q \in \mathcal{Q}$, the function ν is continuous as V_q is assumed to be continuous, $\mathcal{Q}$ is a finite set, and the maximum over a finite number of continuous functions is continuous. To show that $Q^* : \mathcal{Z} \rightrightarrows \mathcal{Q}$ is outer semicontinuous, pick any $z \in \mathcal{Z}$ and a sequence $z_i \to z$. We need to show that for the sequence $y_i \to y$, where $y_i \in Q^*(x_i)$, it holds that $y \in Q^*(z)$. For each point z_i, it holds that $y_i \in Q^*(z_i)$ and thus $\nu(z_i, y_i) = 0$. As ν is a continuous function, it holds that $\lim_{i\to\infty} \nu(z_i, y_i) = 0$ and $\nu(\lim_{i\to\infty}(z_i, y_i)) = 0$. Hence, $\nu(z, y) = 0$ and thus $y \in Q^*(z)$.

For the δ_d component, given $z \in \mathcal{Z}$ and $q \in \mathcal{Q}$, $\mathcal{T}(z, q)$ in (23) can be written as

$$\mathcal{T}(z, q) = \Big\{\eta \in [0, \bar{\delta}_d] : \varrho(\chi(0), q, \eta) \geq 0, \quad \text{where } \chi(0) = z\Big\}, \tag{39}$$

where $\varrho(\chi(0), q, \eta) = \max_{\bar{q} \in (\mathcal{Q}\setminus\{q\}),\, \dot{\chi} = f(\chi, \pi_q(\chi))} V_{\bar{q}}(\chi(\eta)) - V_q(\chi(\eta)) - \mu(|V_q(\chi(\eta))| + \epsilon)$. The system dynamics f are Lipschitz continuous in χ and uniformly continuous in η, namely, ordinary time, therefore by Khalil (2002, Theorem 3.5), the function ϱ has a continuous dependency on the initial state $\chi(0) = z$. Now, fix an arbitrary $z \in \mathcal{Z}$ and define

$$g(\eta) = \varrho(z, q, \eta) \quad \text{for } \eta \in [0, \bar{\delta}_d]. \tag{40}$$

By definition, the inverse image of the closed set $[0, \infty)$ under g is

$$g^{-1}([0, \infty)) = \{\eta \in [0, \bar{\delta}_d] \mid g(\eta) \geq 0\}, \tag{41}$$

which is exactly how $\mathcal{T}(z, q)$ is defined:

$$\mathcal{T}(z, q) = \{\eta \in [0, \bar{\delta}_d] : \varrho(z, q, \eta) \geq 0\} = g^{-1}([0, \infty)). \tag{42}$$

Since g is continuous (as ϱ is continuous in both z and η) and $[0, \infty)$ is closed, it follows from standard topological results that $g^{-1}([0, \infty))$ is closed. Hence, $\mathcal{T}(z, q)$ is a closed subset of $[0, \bar{\delta}_d]$. Next, we define the graph of the set-valued map $\mathcal{T}$ as

$$\text{gph}(\mathcal{T}) = \{(z, q, \eta) \in \mathcal{Z} \times \mathcal{Q} \times [0, \bar{\delta}_d] : \eta \in \mathcal{T}(z, q)\}. \tag{43}$$

Substituting the definition of $\mathcal{T}(z,q)$, we have

$$\mathrm{gph}(\mathcal{T}) = \{(z,q,\eta) \in \mathcal{Z} \times \mathcal{Q} \times [0,\bar{\delta}_d] : \varrho(z,q,\eta) \geq 0\}. \tag{44}$$

Since $\varrho(z,q,\eta)$ is continuous in both z and η, and $[0,\infty)$ is a closed subset of $\mathbb{R}$, the set

$$\{(z,q,\eta) \in \mathcal{Z} \times \mathcal{Q} \times [0,\bar{\delta}_d] : \varrho(z,q,\eta) \geq 0\} \tag{45}$$

is the inverse image of the closed set $[0,\infty)$ under the continuous mapping $(z,q,\eta) \mapsto \varrho(z,q,\eta)$, and is therefore closed. By Rockafellar et al. (2009, Theorem 5.7), a set-valued map whose graph is closed is outer semicontinuous. Hence, the set-valued map $\mathcal{T} : \mathcal{Z} \times \mathcal{Q} \rightrightarrows [0,\bar{\delta}_d]$ is outer semicontinuous. To show that

$$\delta_d^+ = \varsigma(z) = \min(\mathcal{T}(z,q') \cup \{\bar{\delta}_d\}) \tag{46}$$

is outer semicontinuous, pick any $z \in \mathcal{Z}$, any $q' \in Q^*(z)$, and a sequence $z_i \to z$. Suppose that for each z_i we choose

$$y_i = \varsigma(z_i) \in \mathcal{T}(z_i, q' \cup \{\bar{\delta}_d\}, \tag{47}$$

and that $y_i \to y$. We need to show that $y = \varsigma(z)$. Since $\mathcal{T}$ is outer semicontinuous, and $y_i \in \mathcal{T}(z_i) \cup \{\bar{\delta}_d\}$ for all i, it follows that $y \in \mathcal{T}(z,q') \cup \{\bar{\delta}_d\}$. Moreover, by definition, for each z_i the number y_i is the minimum element of the set $\mathcal{T}(z_i,q') \cup \{\bar{\delta}_d\}$, namely,

$$y_i \leq \eta \quad \text{for all } \eta \in \mathcal{T}(z_i, q' \cup \{\bar{\delta}_d\}. \tag{48}$$

Taking the limit as $i \to \infty$, and using the fact that inequalities are preserved in the limit, it follows that

$$y \leq \eta \quad \text{for all } \eta \in \mathcal{T}(z,q') \cup \{\bar{\delta}_d\}. \tag{49}$$

Thus, y is a lower bound of $\mathcal{T}(z,q') \cup \{\bar{\delta}_d\}$. Since $\varsigma(z)$ is defined as the minimum of $\mathcal{T}(z,q') \cup \{\bar{\delta}_d\}$, it holds that $y = \varsigma(z)$. Hence, the δ_d component is outer semicontinuous.

As discussed above, the state components z, τ, and δ_d are continuous on $\mathcal{X}$, hence they are bounded relative to D. The component q is also bounded relative to D as q takes values from a finite set $\mathcal{Q}$.

Lastly, for the q component, by assumption, each V_q is continuous and well-defined for all $z \in \mathcal{Z}$, and since $\mathcal{Q}$ is finite and nonempty, the set $Q^*(z)$ is nonempty, and thus the q component is well-defined for all $x \in \mathcal{X}$. For the δ_d component, by definition, $\bar{\delta}_d > 0$. If $\mathcal{T}(z,q')$ is empty, the minimum is $\bar{\delta}_d$, which is trivially defined. If $\mathcal{T}(z,q')$ is nonempty, then $\min(\mathcal{T}(z,q'))$ exists because $\mathcal{T}(z,q')$ is a closed subset of $[0,\bar{\delta}_d]$ as discussed above. Hence, the δ_d is well-defined for all $x \in \mathcal{X}$. Since z remains z, τ resets to 0, each component of $G(x)$ is well-defined for all $x \in \mathcal{X}$, and $D \subset \mathcal{X}$, it holds that $D \subset \mathcal{X} \subset \mathrm{dom}\ G$.

Thus, the condition (B3) is satisfied. □

Proof of Proposition 2. According to Sanfelice (2021, Proposition 2.34), there exists a nontrivial solution x to the hybrid closed-loop system $\mathcal{H}$ in (19)-(27) for each $x_\circ \in C \cup D$ with $x(0,0) = x_\circ$ if

(B1) The hybrid closed-loop system $\mathcal{H}$ satisfies the hybrid basic conditions; and

(B2) For each $x_\circ \in C \setminus D$ there exists a neighborhood U of $x_\circ$ such that for every $x \in U \cap (C \setminus D)$,

$$F(x) \cap T_C(x) \neq \varnothing. \tag{50}$$

As F is single-valued, (50) simplifies to

$$F(x) \in T_C(x). \tag{51}$$

Additionally, every maximal solution x to $\mathcal{H}$ is not Zeno, and every bounded solution x to $\mathcal{H}$ has jump times that are uniformly lower bounded by a positive constant if the conditions (B1) and (B2) hold and if

(B3) $G(D) \cap D = \varnothing$.

Lastly, every maximal solution x to $\mathcal{H}$ is complete if the conditions (B1) and (B2) hold, F is globally Lipschitz on C, and

(B4) $G(D) \subset C \cup D$.

By Proposition 1, the hybrid closed-loop system $\mathcal{H}$ in (19)-(27) satisfies the hybrid basic conditions under the assumptions and therefore condition (B1) is satisfied.

The strict flow set $C \setminus D$ is given by

$$C \setminus D = \{x \in \mathcal{X} : \tau < \delta_d\} \cup \{x \in \mathcal{X} : \tau \geq \delta_d, q \in Q^*(z)\}. \tag{52}$$

Since the set $\{x \in \mathcal{X} : \tau < \delta_d\}$ is open, every point x with $\tau < \delta_d$ is an interior point. Consequently, as such x, there are no constraints on the possible flow directions. For the set $\{x \in \mathcal{X} : \tau \geq \delta_d, q \in Q^*(z)\}$, the value of τ cannot decrease if $\tau = \delta_d$ and the value of q cannot change as $Q^*(z)$ is a finite subset of $\mathbb{N}$. Hence, the q component of the tangent cone for the set $\{x \in \mathcal{X} : \tau \geq \delta_d, q \in Q^*(z)\}$ is equal to zero and the τ component is equal to $\mathbb{R}_{\geq 0}$ if $\tau = \delta_d$. The resulting tangent cone for $C \setminus D$ is given by

$$T_{C\setminus D}(x) = \begin{cases} \mathbb{R}^{n+3} & \text{if } \tau < \delta_d \\ \mathbb{R}^n \times \{0\} \times \mathbb{R}_{\geq 0} \times \mathbb{R} & \text{if } \tau = \delta_d \\ \mathbb{R}^n \times \{0\} \times \mathbb{R}^2 & \text{if } \tau > \delta_d \end{cases}. \tag{53}$$

For the flow map F in (19) it holds that $F(x) \in T_{C\setminus D}(x)$ for $x \in C \setminus D$. Hence, condition $(B2)$ is satisfied.

By the definition of the jump map G in (22), $q^+ \in Q^*(z)$ and $z^+ = z$ for all $x \in D$. As the jump set D in (27) requires $q \in \mathcal{Q} \setminus Q^*(z)$, it holds that $G(D) \cap D = \varnothing$ and condition (B3) is satisfied. Furthermore, by the definition of the flow set C in (20)-(21), $G(D) \subset C$ as $q^+ \in Q^*(z)$, $z^+ = z$, and $\tau^+ = 0$. Hence, $G(D) \subset C \cup D$ and condition (B4) is satisfied.

□

Proof of Theorem 1. To solve Problem $(\star)$, the hybrid closed-loop system $\mathcal{H}$ in (19)-(27) needs to:

(P1) Preserve the properties of the individual policies, namely, inducing asymptotic stability of the compact set $\mathcal{Z}^*$;

(P2) Achieve overall performance, as measured by the state value function, that is at least as good as—and, if possible, better than—that obtained by holding any individual policy; and

(P3) Prevent chattering under measurement noise by guaranteeing a robustness margin $\varepsilon > 0$ such that for every measurement noise m satisfying $|m| \leq \varepsilon$, the switching mechanism prevents rapid switching (chattering).

To prove condition (P1), we will construct a hybrid Lyapunov function and show a strict decrease during flows and jumps unless the state of the system z is in the compact set $\mathcal{A}$, for which the hybrid Lyapunov function equates to zero. In particular, according to Sanfelice (2021, Definition 3.17), the sets $\mathcal{U}, \mathcal{A} \subset \mathcal{X}$ and the function $\mathcal{L} : \mathcal{X} \to \mathbb{R}$ define a Lyapunov function candidate on $\mathcal{U}$ with respect to $\mathcal{A}$ for the hybrid closed-loop system $\mathcal{H} = (C, F, D, G)$ if the following conditions hold:

(L1) $(\overline{C} \cup D \cup G(D)) \cap \mathcal{U} \subset \text{dom } \mathcal{L}$;

(L2) $\mathcal{U}$ contains an open neighborhood of $\mathcal{A} \cap (C \cup D \cup G(D))$;

(L3) $\mathcal{L}$ is continuous on $\mathcal{U}$ and locally Lipschitz on an open set containing $\overline{C} \cap \mathcal{U}$; and

(L4) $\mathcal{L}$ is positive definite on $C \cup D \cup G(D)$ with respect to $\mathcal{A}$.

Consider the Lyapunov candidate function defined by

$$\mathcal{L}(x) = -V_q(z) + V_q(z_q^*), \tag{54}$$

where $x = (z, q, \tau, \delta_d) \in \mathcal{X}$ and $z_q^* \in \mathcal{Z}_q^*$ is the (possibly nonunique) state at which V_q attains its global maximum. To obtain a global result, we choose $\mathcal{U} = \mathcal{X}$. Furthermore, the compact set that we wish to globally asymptotically stabilize is defined as

$$\mathcal{A} := \mathcal{Z}^* \times \mathbb{R}_{\geq 0} \times \mathcal{Q} \times \mathbb{R}_{\geq 0} \subset \mathcal{X}, \quad \text{with} \quad \mathcal{Z}^* = \bigcup_{q \in \mathcal{Q}} \mathcal{Z}_q^*. \tag{55}$$

Condition (L1) is satisfied since $(\overline{C} \cup D \cup G(D)) \subset \mathcal{X}$ and $\operatorname{dom} \mathcal{L} = \mathcal{X}$. Similarly, because both $\mathcal{A}$ and $C \cup D \cup G(D)$ are subsets of $\mathcal{X}$, condition (L2) holds. By assumption, for each $q \in \mathcal{Q}$, V_q is Lipschitz continuous on $\mathcal{Z}$, which ensures that $\mathcal{L}$ is continuous on $\mathcal{X}$ and locally Lipschitz on an open subset containing $\overline{C} \cap \mathcal{X}$; hence, condition (L3) is satisfied. Moreover, by assumption the reward function in (4) attains its global maximum on $\mathcal{Z}^*$ and is strictly decreasing as the distance from $\mathcal{Z}^*$, and hence from $\mathcal{A}$, increases. Consequently, each state value function V_q attains its global maximum at $\mathcal{Z}^*$ and is negative definite with respect to $\mathcal{Z}^*$. Therefore, by construction, for each $q \in \mathcal{Q}$ the Lyapunov function in (54) is positive definite on $\mathcal{X}$, and hence on $C \cup D \cup G(D)$, with respect to $\mathcal{A}$, so that condition (L4) is satisfied.

Next, we use the Lyapunov candidate function in (54) to show that $\mathcal{H}$ globally asymptotically stabilizes $\mathcal{A}$. According to Sanfelice (2021, Theorem 3.19), given sets $\mathcal{U}, \mathcal{A} \subset \mathcal{X}$ and a function $\mathcal{L} : \mathcal{X} \to \mathbb{R}$ that defines a Lyapunov candidate on $\mathcal{U}$ with respect to $\mathcal{A}$ for the hybrid closed-loop system $\mathcal{H} = (C, F, D, G)$ with state $x \in \mathcal{X}$, the set $\mathcal{A}$ is asymptotically stable for $\mathcal{H}$ provided that $\mathcal{A}$ is compact, $\mathcal{H}$ satisfies the hybrid basic conditions, every maximal solution x to $\mathcal{H}$ complete, and the following two conditions hold during flows and jumps:

(L5) $\langle \nabla \mathcal{L}(x), F(x) \cap T_C(x) \rangle < 0$ for all $x \in (C \cap \mathcal{U}) \setminus \mathcal{A}$; and

(L6) $\mathcal{L}(G(x)) - \mathcal{L}(x) < 0$ for all $x \in (D \cap \mathcal{U}) \setminus \mathcal{A}$.

By Proposition 1, the hybrid closed-loop system $\mathcal{H}$ satisfies the hybrid basic conditions under the assumptions. Moreover, by Proposition 2, every maximal solution x to $\mathcal{H}$ is complete.

By assumption, the reward function is strictly decreasing as the distance from $\mathcal{Z}^*$ increases. Therefore, for each $q \in \mathcal{Q}$, the state value function V_q attains its global maximum at $\mathcal{Z}_q^*$ and decreases as the distance from $\mathcal{Z}_q^*$ increases. Moreover, since each policy $\pi_q \in \Pi$ globally asymptotically stabilizes $\mathcal{Z}_q^*$, the state z globally asymptotically converges toward $\mathcal{Z}_q^*$ along flows, resulting in a strict increase in $V_q(z)$ while q remains constant. Consequently, the Lyapunov candidate function in (54) strictly decreases along flows for all $x \in \mathcal{X} \setminus \mathcal{A}$, thereby satisfying condition (L5).

Furthermore, by definition of the jump set D in (27), jumps occur only when $q \in \mathcal{Q} \setminus Q^*(z)$; the jump map G in (22) then updates q to an element of $Q^*(z)$. Since $V_{q'}(z) > V_{\bar{q}}(z)$ for all $q' \in Q^*(z)$ and $\bar{q} \in \mathcal{Q} \setminus Q^*(z)$ for each $z \in \mathcal{Z}$, it follows that the Lyapunov candidate function undergoes a strict decrease at jumps, satisfying condition (L6).

As all the conditions above are satisfied, the hybrid closed-loop system $\mathcal{H}$ globally asymptotically stabilizes the compact set $\mathcal{A}$. This, in turn, renders the compact set $\mathcal{Z}^*$ asymptotically stable, and thus condition (P1) is satisfied.

During flows, the policy index q remains constant, so the hybrid closed-loop system $\mathcal{H}$ behaves exactly as if a single policy were held continuously. At jump instants, the update rule (10) selects a new index $q' \in Q^*(z)$, where $Q^*(z)$ is the set of indices that maximize the state value function at the current state z. This ensures that $V_{q'}(z) \geq V_q(z)$ for all $q \in \mathcal{Q} \setminus \{q'\}$, so the policy chosen at a jump delivers performance that is at least as good as—and potentially better than—that of the policy before the jump. Thus, since the overall performance of $\mathcal{H}$, as measured by the state value function, is maintained or improved during both flows and jumps, condition (P2) is satisfied.

Proposition 2 guarantees that, under its assumptions—that for each $q \in \mathcal{Q}$ the approximate state value function $V_q : \mathcal{Z} \to \mathbb{R}$ is continuous and both the policy $\pi_q : \mathcal{Z} \to \mathcal{U}$ and the dynamics $f : \mathcal{Z} \times \mathcal{U} \to \mathcal{Z}$ are Lipschitz continuous—every maximal solution to the hybrid closed-loop system $\mathcal{H}$ in (19)-(27) is complete, non-Zeno, and has jump times uniformly lower bounded by a positive

constant. Consequently, in the absence of measurement noise, arbitrarily fast switching, that is, chattering, cannot occur.

Now, suppose that measurement noise m is present such that $z + m \in \mathcal{Z}$. Then, it is possible that a policy q belongs to $Q^*(z)$ while $q \notin Q^*(z+m)$; that is, the optimal policy for the measured state $z + m$ may differ from the optimal policy for the true state z. To prevent an instantaneous switch in such cases, we must ensure that the dwell-time parameter δ_d is always reset to a value greater than zero. Since the jump set (27) requires that $\tau \geq \delta_d$ and the timer τ is reset to zero upon each switch, a positive reset value for δ_d guarantees that a nonzero time interval of at least δ_d must elapse before another jump occurs.

To establish that δ_d is always positive, consider its update in (22). Two scenarios arise:

1. If the set $\mathcal{T}$ is empty, then by definition δ_d is reset to $\bar{\delta}_d > 0$.
2. If $\mathcal{T}$ is non-empty, let η^* denote the smallest time horizon in $\mathcal{T}$, that is, the smallest $\eta^* > 0$ for which the ratio in (23) reaches μ. By the definition of Q^* in (10), for every $z \in \mathcal{Z}$ and every $q \in Q^*(z)$ we have
$$V_q(z) \geq V_{\bar{q}}(z) \quad \text{for all } \bar{q} \in \mathcal{Q}.$$
In particular, at $\eta = 0$ we obtain
$$\frac{\max\limits_{\bar{q} \in (\mathcal{Q} \setminus \{q\})} V_{\bar{q}}\big(\chi(0)\big) - V_q\big(\chi(0)\big)}{|V_q\big(\chi(0)\big)| + \epsilon} < 0, \tag{56}$$
for all $q \in Q^*(\chi(0))$ and $\chi(0) \in \mathcal{Z}$. Moreover, as demonstrated in the proof of Proposition 1, the ratio in (23) has a continuous dependency on the initial state $\chi(0)$. Therefore, there exists a smallest time $\eta^* > 0$ such that
$$\frac{\max\limits_{\bar{q} \in (\mathcal{Q} \setminus \{q\})} V_{\bar{q}}\big(\chi(\eta^*)\big) - V_q\big(\chi(\eta^*)\big)}{|V_q\big(\chi(\eta^*)\big)| + \epsilon} = \mu. \tag{57}$$
Since the ratio at $\eta = 0$ is strictly negative and $\eta \in [0, \bar{\delta}_d]$, it follows that $\eta^* > 0$. Consequently, the updated value of δ_d is always positive.

Thus, even in the presence of measurement noise, the requirement $\tau \geq \delta_d > 0$ ensures that a positive dwell time elapses between jumps, thereby preventing arbitrarily fast policy switching. Hence, condition (P3) is satisfied. □