Piecewise Reinforcement Learning for Hybrid Systems

Mi Zhou, Jiazhi Li, Masood Mortazavi, Ning Yan, and Chaouki Abdallah

***Abstract*—In this article, a state-dependent piecewise reinforcement learning (PRL) method is proposed to learn an optimal control policy for state-dependent switched systems. This is an analogy to learning multiple piecewise continuous control inputs. We observe this method's robust and impressive performance, explain it in rigorous mathematical language, and apply it to some canonical examples. Based on the deterministic policy gradient method, the PRL is compared with the vanilla policy gradient method in three customized demonstrative environments with switched dynamics. We theoretically demystify why PRL outperforms RL in these systems.**

***Index Terms*—Reinforcement learning, state-dependent switched system, Hamilton-Jacobi-Bellman equation**

I. Introduction

Optimal control of state-dependent switched systems has long posed significant challenges in the control community. Such systems arise in diverse applications, including fermentation processes [1], temperature control [2], aerospace systems [3], robots [4], [5], and natural phenomena [6]. Under different names, optimal control of differential systems with discontinuous right-hand side [7], optimal control for piecewise smooth systems [8], optimal control for systems with isolated equality constraints [9], optimal control for systems with linear complementarity constraints, and optimal control for hybrid switched systems [10], these problems share common features. Representative examples include YO-YO models, controlled systems with Coulomb friction, electrical relay systems [7], and Spring Loaded Inverted Pendulum (SLIP) models [8]. The theoretical foundation for this problem (hybrid minimum principle) has been rigorously developed in [9]–[13], with costate analysis in [13]–[15]. Numerical methods to solve this problem can be found in [16], [17].

However, if hybrid systems have regional dynamics or a partitioned state space, determining both switching sequence and switching time yields a mixed integer nonlinear programming (MINLP) problem, whose exhaustive solution is computationally intractable [16]. Consequently, most approaches seek feasible suboptimal solutions. Methods include graph search [18], nonlinear programming [19], dynamic programming [20], model predictive control [8], and gradient descent-based schemes [21]. For example, the study of optimal control of multi-region state-dependent switched systems appeared in [18] where the authors used branch-and-bound based model predictive control. This combination of branch-and-bound and model predictive control still demands high computing. [19] proposed a numerical method based on dynamic programming and nonlinear programming after discretizing the state and input spaces. In [22], a hybrid Bellman Equation for systems with regional dynamics was proposed where the switching interface was discretized and then dynamic programming was used at the high level to decide which region to switch to and which two states to be chosen as the boundary condition of a low-level optimal control problem. The time complexity and space complexity of this algorithm are humongous.

With so many numerical methods for optimal control problems, none of them is a very general and convenient algorithm for state-dependent switched systems. As two important theoretical tools in optimal control communities, the Pontryagin minimum (maximum) principle and dynamic programming have their respective limitations. This article, therefore, leverages reinforcement learning to address these problems. Recently, reinforcement learning has been substantially adopted in the control community [23]–[25]. Most of them consider either the infinite horizon reinforcement learning problem [23] or time-dependent switched systems [24]. In [26], the authors examined a finite-horizon reinforcement learning problem. However, the system must be continuous and should have a control-affine form. A recent work [25] proposed a piecewise linear parameterization of policies in reinforcement learning to improve interpretability. However, this mechanism has demonstrated fewer advantages in the simulation environments than vanilla reinforcement learning.

In this article, we first investigate finite-horizon reinforcement learning for state-dependent switched systems. We present some theoretical foundations of finite-horizon RL and state-dependent switched systems and highlight the limitations of applying standard actor-critic methods directly in this hybrid setting. We then introduce a new piecewise reinforcement learning framework inspired by [25]. Finally, we validate our approach on several examples, demonstrating its improved efficiency over the conventional RL framework.

This article is organized as follows: In Section II, we formulate the hybrid optimal control problem. In Section III, we cast the finite-horizon optimal control task in the Hamilton-Jacobi-Bellman framework and provide an error analysis for approximate dynamic programming. In Section IV, we present our piecewise reinforcement learning algorithm. Section V reports three numerical examples that validate our method and

Chaouki Abdallah is with the School of Electrical and Computer Engineering, Georgia Institute of Technology. Mi Zhou, Jiazhi Li, Masood Mortazavi, and Ning Yan are with Futurewei Technologies, San Jose, CA 30332. Email: `mzhou2@futurewei.com`, `jli5@futurewei.com`, `yan.ningyan@futurewei.com`, `masoodmortazavi@gmail.com`, `ccabdallah@gatech.edu`.

compare it to the standard reinforcement learning approach. Finally, Section VI concludes this article and outlines directions for future work.

II. Problem formulated

A general state-dependent switched system is defined as follows:

$$\dot{x}(t) = f_{q(t)}(x(t), u(t)), \quad (x(t), u(t)) \in \mathcal{R}_i \tag{1}$$

where $x \in \mathbb{R}^{n_x}$ denotes the state, $u \in \mathbb{R}^{n_u}$ denotes the continuous control input, $q(t)$ is the index of sub-systems with which we can define a switching sequence $q(t) = \{i | i \in (1, 2, ..., m)\}$, and $f_i(x(t), u(t), t)$ is a continuously differentiable function in the region $\mathcal{R}_i$ which satisfies $\mathcal{R}_i \bigcap \mathcal{R}_j = \{x | g_{ij}(x) = 0\}$[1]. Then the optimal control problem is to determine the optimal control u and switching time instant τ_i to minimize the following objective function:

$$J = \psi(x(t_f)) + \sum_{i=1}^{m} \int_{\tau_i}^{\tau_{i+1}} L_{q(t)}(x(t), u(t)) \mathrm{d}t, \tag{2}$$

where stage cost $L_i(x, u)$ is also piecewise continuously differentiable functions. x_f denotes the terminal state and $\psi(x(t_f))$ is the terminal cost.

We make the following assumptions.

1) There is no jump behavior of the state.
2) In each region, the system is reachable.
3) The optimal controller has bounded variations.

Solving this problem is of significant difficulty, especially when there are multiple regions. What's worse, a nonlinear switching interface will further complicate this problem. Thus, in this article, we use reinforcement learning, a sampling-based method that does not rely on prior knowledge of the system model, to solve this problem. This method combines the high-level graph search and the low-level optimal control problem into one step to construct the value function in the state space.

The formulated problem Eqn. (1) and Eqn. (2) can be transformed into the following problem:

$$\dot{x}(t) = \sum_{i=1}^{m} \mathbb{1}_{x(t) \in \mathcal{R}_i} f_i(x, u), \tag{3}$$

where $\mathbb{1}(\cdot)$ denotes the characteristic function. Similarly, the stage cost is written as

$$L = \sum_{i=1}^{m} \mathbb{1}_{x(t) \in \mathcal{R}_i} L_i(x, u), \tag{4}$$

By reformulating the original problem in this way, the resulting system can be regarded as highly nonlinear.

III. Theoretical error analysis

A. Hamilton-Jacobi-Bellman Equation for Hybrid Systems

Define the following value function

$$V(x(t_f), t) = \min_{u(t)} \int_{t}^{t_f} L(x(t), u(t)) \mathrm{d}t + \psi(x(t_f),$$

where $V(x(t), t_f) = \psi(x(t_f))$. The Hamilton-Jacobi-Bellman equation is given by

$$V(x(t), t) = \\ \min_{u} \{V(x(t + dt), t + dt) + \int_{t}^{t+dt} L(x(s), u(s)) \mathrm{d}s\}. \tag{5}$$

Assuming the cost-to-go function is continuously differentiable in both x and t, we can apply a first-order Taylor expansion to the term $V(x(t + dt), t + dt)$,

$$\begin{aligned} &V(x(t + dt), t + dt) \\ &= V(x(t), t) + \frac{\partial V(x, t)}{\partial t} \mathrm{dt} + \frac{\partial \mathrm{V(x, t)}}{\partial \mathrm{x}} \cdot \mathrm{\dot{x}(t) dt} + \mathrm{o(dt)} \\ &= V(x(t), t) + \frac{\partial V(x, t)}{\partial t} \cdot 1\mathrm{dt} + \frac{\partial \mathrm{V(x, t)}}{\partial \mathrm{x}} \cdot \mathrm{\dot{x}(t) dt} + \mathrm{o(dt)} \\ &= V(x(t), t) + \left[\frac{\partial V(x, t)}{\partial x}, \frac{\partial V(x, t)}{\partial t}\right] \cdot \begin{bmatrix} \dot{x}(t) \\ 1 \end{bmatrix} \mathrm{dt} + \mathrm{o(dt)}, \end{aligned} \tag{6}$$

where $o(dt)$ represents terms of order higher than dt.

Nonetheless, in optimal control for hybrid systems, we must account for a non-differentiable value function at the switching interface. Moreover, the optimal control input u may be discontinuous or non-differentiable at the switching interface.

Theorem III.1. *The value function for the defined optimal control problem Eqn.* (1) *and Eqn.* (2) *is continuous but may not be differentiable at the switching interface with switching state $x(\tau)$ and switching time instant τ.*

$$V(x(\tau_-), \tau_-) = V(x(\tau_+), \tau_+).$$

Proof. Based on the dynamic programming formula,

$$\begin{aligned} V(x(\tau_-), \tau_-) &= V(x(\tau_+), \tau_+) + \int_{\tau_-}^{\tau_+} L(x, u) \mathrm{d}t \\ &\approx V(x(\tau_+), \tau_+) + \tilde{L}(x, u)(\tau_+ - \tau_-) \end{aligned}$$

where $\tilde{L}(x, u)$ is a bounded constant value. Consequently, as $\tau_- \rightarrow \tau_+$, the value function remains continuous $V(x(\tau_-), \tau_-) = V(x(\tau_+), \tau_+)$. □

Recall the definition of Hamiltonian

$$H := \inf_{u \in \mathcal{U}} (\lambda f(x, u) + L(x, u)), \tag{7}$$

where λ is called the co-state. Below, we present two examples illustrating optimal control inputs that are, respectively, (i) continuous but nondifferentiable, and (ii) discontinuous. **Example 1: non-differentiable optimal controller u:** The following is a simple example to show the nondifferentiability

[1] $g_{ij}(x) = 0$ is called the switching interface

of the optimal controller for a hybrid system. Consider a one-dimensional switched system:

$$\min J = \frac{1}{2}\int_0^1 (x^2 + u^2)\mathrm{d}t$$

$$\text{s.t.}, \begin{cases} \dot{x} = 2u, & x > 1 \\ \dot{x} = u, & x < 1 \end{cases}, x(0) = 2.$$

Given the fact that the Hamiltonian is continuous at the switching interface and the optimality conditions in [13], we can obtain [2]

$$\begin{aligned} \frac{1}{2}(x_-^2 + u_-^2) + 2\lambda_- u_- &= \frac{1}{2}(x_+^2 + u_+^2) + \lambda_+ u_+ \\ \frac{1}{2}x_-^2 - \frac{1}{2}u_-^2 &= \frac{1}{2}x_+^2 - \frac{1}{2}u_+^2 \\ &\Rightarrow u_-^2 = u_+^2. \end{aligned}$$

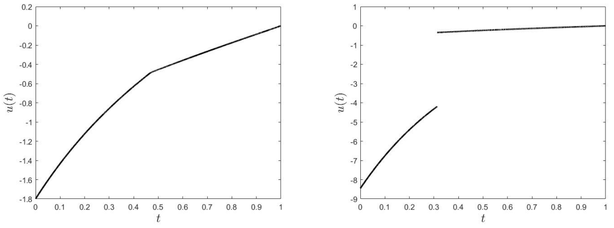
In addition, $x_- = x_+$, $\lambda_- = -\frac{1}{2}u_-$, $\lambda_+ = -u_+$. Based on the Euler-Lagrange equation, we know that, before switching, $\dot{\lambda}(t) = -x(t) \Rightarrow \dot{u}(t) = 2x(t)$, while after switching, $\dot{\lambda}(t) = -x(t) \Rightarrow \dot{u}(t) = x(t)$. Combining these facts, we can derive the continuity but nondifferentiability at the switching state of the optimal controller in this example(the optimal controller is shown in Fig. 1(a)).

Example 2: discontinuous optimal controller u: Consider the following problem

$$\min J = \frac{1}{2}\int_0^1 (x^2 + u^2)\mathrm{d}t,$$

$$\text{s.t.}, \dot{x} = \begin{cases} 2x + u, & x > 1 \\ -x + u, & x < 1 \end{cases}, \ x(0) = 2.$$

For this example, using the optimality conditions derived in [13] and the algorithm proposed in [16], we observe that the optimal controller is discontinuous as shown in Fig. 1(b).



(a) (b)

Fig. 1. (a) Optimal control of Example 1 (switching time $\tau = 0.4694$ and optimal cost $J = 1.0209$); (b) Example 2 (switching time $\tau = 0.3132$ and optimal cost $J = 6.5274$).

A well-known fact is that the HJB equation (5) is a partial differential equation, which is notoriously difficult to solve, especially to obtain the value function. Thus, in this article, we resort to the model-free method called deep deterministic policy gradient in order to solve Eqn. (5). In the following, the notation x is the one after augmenting time t and $f(x, u) = [\dot{x}(t), 1]^\top$.

B. Approximate Dynamic Programming and Finite Horizon RL

Most reinforcement learning research focuses on infinite horizon problems, akin to the linear quadratic regulator in optimal control theory, which admit stationary solutions. In contrast, most optimal control problems have fixed terminal time, requiring optimization of an objective function over a fixed terminal time horizon. Consequently, the value function depends on both state and time, i.e., $V(x,t)$, and the optimal control policy is time-varying $\mu(x,t)$. The corresponding cost-to-go function can be written as $V^\mu(x,t)$.

Theorem III.2 ([27] Chapter VI)**.** *$u = \mu^*(x,t)$ is the optimal control policy if it minimizes the left-hand-side of the HJB equation* (5) *for all x and $t \in [0, t_f]$.*

Function approximators have long been used in optimal control communities. Chebyshev, Legendre, and Fourier basis functions have been commonly used for interpolation [17], [28], see [29], [30] for detailed error bound analysis. Below, we analyze the approximation errors of the optimal control and the value function $V(x,t)$, i.e., $||u - \hat{u}||$ and $||V(x,t) - \hat{V}(x,t)||$. We use the approximants $\hat{u}(t) = \theta\phi(x,t)$ and $\hat{V}(x,t) = w\varphi(x,t)$ where θ, and w are the weights and $\phi(\cdot)$, $\varphi(\cdot)$ are their corresponding basis functions.

Theorem III.3 (Approximation theory of deep neural network [31](Theorem 4.16))**.** *Let $\rho : \mathbb{R} \to \mathbb{R}$ and $n, m, k \in \mathbb{N}$. Let $\mathcal{NN}^\rho_{n,m,k}$ represent the class of functions $\mathbb{R}^n \to \mathbb{R}^m$ described by forward neural networks with n neurons in the input layer, m neurons in the output layer, and an arbitrary number of hidden layer, each with k neurons with activation function ρ. Every function in the output layer has the identity activation function. Let ρ be ReLU and $p \in [1,\infty)$. Then $\mathcal{NN}_{n,m,n+m+1}$ is dense* [3] *in $L^p(\mathbb{R}^n;\mathbb{R}^m)$ with respect to the usual L^p norm* [4]*.*

Theorem III.4. *Assume that the real value function satisfies the global Lipschitz condition $||V(x(t),t) - V(\hat{x}(t),t)|| \leq \gamma||x(t) - \hat{x}(t)||$ where γ is the Lipschitz constant. Then there exists a small $\epsilon' > 0$ such that*

$$||V(x(t),t) - \hat{V}(\hat{x}(t),t)|| \leq \epsilon', \tag{8}$$

where $\hat{V}$ is the value function approximator and $\hat{x}(t)$ is the state generated by the approximator control input $\hat{u}(t)$ that satisfies the state dynamics Eqn. (1)*.*

Proof. Using Theorem III.3, it follows that there exists $\epsilon = \max\{\epsilon_1, \epsilon_2\}$ such that $||u(t) - \hat{u}(t)|| \leq \epsilon_1 \leq \epsilon$ and $||V(x(t),t) - \hat{V}(x(t),t)|| \leq \epsilon_2 \leq \epsilon$. Assuming that $f_q(x,u)$ is Lipschitz, it follows that $x(t) - \hat{x}(t) = x(0) - \hat{x}(0) + \int_0^t f(x(s),u(s)) - f(\hat{x}(s),\hat{u}(s))\mathrm{d}s$. Since the initial state is the same, Hölder's inequality implies $||x(t) - \hat{x}(t)|| \leq M||u - \hat{u}||_1 + M\int_0^t ||x(s) - \hat{x}(s)||\mathrm{d}s$. By the Bellman-Gronwall

[2] To simplify notation, in this article, both $x_\pm$ and $x(\tau\pm)$ means the state at the switching interface. Same for the other variables.

[3] We say ρ networks are dense in $L^p(\mathcal{X};\mathcal{Y})$ if for any $f^* \in L^p(\mathcal{X};\mathcal{Y})$ and $\epsilon > 0$, there exists a ρ network f such that $||f^* - f||_p \leq \epsilon$.

[4] L^p norm is defined as $||f||_p := (\int_\mathcal{X} |f|^p \mathrm{d}x)^{1/p} < \infty$, $p \in [1,\infty)$.

Lemma, we then deduce that $||x(t) - \hat{x}(t)|| \leq M||u(t) - \hat{u}(t)||_1 e^{Mt}$. Hence, by applying the triangle inequality, we obtain

$$
\begin{aligned}
&||V(x,t) - \hat{V}(\hat{x},t)|| \leq \\
&||V(x,t) - V(\hat{x},t)|| + ||V(\hat{x},t) - \hat{V}(\hat{x},t)|| \\
&\leq \gamma||x - \hat{x}|| + \epsilon \leq \gamma\beta||u - \hat{u}||_1 + \epsilon \\
&\leq (\gamma\beta + 1)\epsilon \leq \epsilon',
\end{aligned}
$$

where $\beta = Me^{Mt}$ and $\epsilon' \geq (\gamma\beta + 1)\epsilon$. As $\epsilon' \to 0$, $V(x,t) \to \hat{V}(x,t)$ in L^p. This completes the proof. □

IV. Piecewise reinforcement learning (PRL)

The deterministic policy gradient algorithm utilizes an actor-critic mechanism to learn an optimal policy via environment interactions. An actor neural network parameterizes the policy (i.e., control input in our problem) and a critic neural network parameterizes the Q-function ($V(x_t) = \max_{u_t} Q(x_t, u_t)$)[5]. The parameters of the actor are updated using the policy gradient method as in [32]:

$$
\begin{aligned}
\nabla_{\theta^{\mu_i}} \mu|_{x_t} = \frac{1}{N} \sum_t \nabla_u Q(x, u|\theta^{Q_i})|_{x=x_t, u=\mu(x_t)} \\
\nabla_{\theta^{\mu_i}} \mu(x|\theta^{\mu_i})|_{x_t}. \qquad (9)
\end{aligned}
$$

By Theorem III.1, the optimal controller for systems with regional dynamics is generally not continuous and differentiable. Thus, the policy gradient estimation in Eqn. (9) is highly biased [33].

To address this challenge, we introduce a piecewise reinforcement learning framework that learns an optimal switched controller for hybrid systems over a fixed time horizon. Fig. 2 shows the proposed piecewise actor framework. In this framework, n_d policies are generated that map the state directly to the policy μ. Then, a multilayer perceptron (MLP) with a Gumbel softmax activation function is used to learn the choice of each policy by generating a one-hot encoding. This mechanism can map states to different policy networks and thus solves the non-differentiability problem of the original optimal control problem. The critic follows the standard DPG structure.

V. Illustrative examples

We evaluate the proposed algorithm on three customized environments with state-dependent switches, i.e., a first-order switched system (FO), a multiple region switched system (MR), and an actuated Spring-mass model of hopping (Spring). Each environment is discretized using a forward Euler scheme, $x_{t+1} = x_t + f(x_t, u_t)dt$. We let $dt = 0.01$.

The MLP has a_l hidden layers and the critic network has c_l hidden layers with ReLu activation functions and ReLu output activation function. Each example is run with 5 different randomly generated seeds under N episodes, which leads to a $N \times \frac{tf}{dt}$ total interactions with the environments. Results are reported as the mean ±1 standard deviation across trials. All the simulations are performed in PyTorch on a desktop PC equipped with an NVIDIA GeForce RTX 4060 GPU.

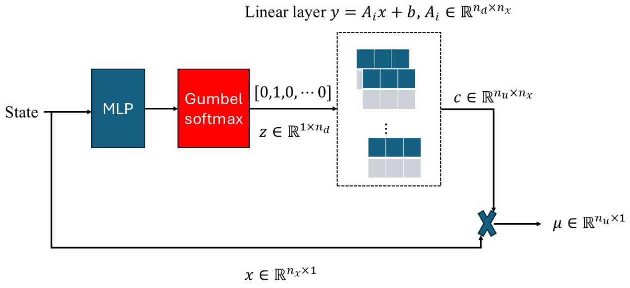

Linear layer $y = A_i x + b$, $A_i \in \mathbb{R}^{n_d \times n_x}$
State
MLP
Gumbel softmax
[0,1,0, ⋯ 0]
$z \in \mathbb{R}^{1 \times n_d}$
$c \in \mathbb{R}^{n_u \times n_x}$
$\mu \in \mathbb{R}^{n_u \times 1}$
$x \in \mathbb{R}^{n_x \times 1}$

Fig. 2. The actor of the piecewise reinforcement learning algorithm using Gumbel softmax.

A. Tested Environments

1) A first-order system: FO: The first example is a first-order state-dependent switched system.

$$
\begin{aligned}
\max J = \frac{1}{2}\int_0^1 -(x^2 + u^2)\mathrm{d}t \\
s.t.,\ \dot{x} = \begin{cases} 2x + u, & x > 1 \\ -x + u, & x < 1 \end{cases} \\
x(0) = 2,
\end{aligned}
$$

where the fixed terminal time $t_f = 1$. The states start at $x(0) = 2$. The system's dynamic switches at the state $x = 1$. The reward function is designed as $r = -0.5(x_t^2 + u_t^2)$.

2) A multi-region system: MR: Next, we consider the example adapted from [34].

$$
\begin{aligned}
\max J = \int_0^{t_f} -\frac{1}{2}(x^\top x + u^2)dt \\
\text{s.t.}, \dot{x} = A_q x + B_q u
\end{aligned}
$$

with

$$
A_1 = \begin{bmatrix} -1 & 2 \\ -2 & -1 \end{bmatrix}, A_2 = \begin{bmatrix} -1 & -2 \\ 1 & -0.5 \end{bmatrix}, A_3 = \begin{bmatrix} -0.5 & -5 \\ 1 & -0.5 \end{bmatrix},
$$

$$
A_4 = \begin{bmatrix} -1 & 0 \\ 2 & -1 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 1 \end{bmatrix}.
$$

The terminal time is $t_f = 2$. Separating regions are $\mathcal{R}_i, i = 1,2,3,4$, the switching interfaces are $m_{12} = x_2 + 5 = 0$, $m_{13} = x_1 + 5 = 0$, $m_{23} = -m_{32} = x_1 - x_2 = 0$, $m_{24} = -m_{42} = x_1 + 2 = 0$, $m_{34} = -m_{43} = x_2 + 2 = 0$. The controller is constrained in all locations to $-10 \leq u_q \leq 10$. The observation space is the position (x_1, x_2, t) where $x_1 \in [-10, 10]$, $x_2 \in [-10, 10]$, $t \in [0, 2]$. The action space is the control input $u \in [-10, 10]$. The reward function is defined as $r_t = -0.5(x_t^\top x_t + u_t^2)$.

[5] To simplify the notation, the x also includes the time variable t. The $\mu(x|\theta^\mu)$ is the parameterized optimal control policy and $Q(x_t, u_t)$ is parameterized with parameters θ^Q and learned using the HJB equation (5).

3) Actuated Spring-mass model of hopping: Spring: The actuated Spring-mass hopper has four states: the height of the mass (z) and its time derivative ($\dot{z}$), and the natural length of the leg spring (L) and its time derivative ($\dot{L}$). There are two phases of the hopper: when the hopper is on the ground, the dynamics of the mass are

$$\ddot{z} = \frac{K(L)(L-z) + D(L)(\dot{L}-z)}{m} - g,$$

where m is the value of mass, $K(L)$ and $D(L)$ are the stiffness and damping coefficient respectively. When the hopper is in the flight phase, the dynamics is $\ddot{z} = -g$. The robot lifts off when $L = z$. A more illustrative example of the model can be found in [35]. We write the states of the robot as $x = [z, \dot{z}, L, \dot{L}]^\top$. This leads to a system with piecewise-smooth dynamics:

$$\dot{x} = f(x,u) = \begin{cases} [\dot{z}, -g, \dot{L}, u]^\top, \ z - L > 0 \\ [\dot{z}, F(x)/m - g, \dot{L}, u]^\top, \ z - L < 0 \end{cases}$$

where $F(x) = K(L)(L-z) + D(L)(\dot{L}-z)$ and u is the control input. We let $K(L) = 10$ and $D(L) = 0.1$. The initial state is set to $x_0 = [z(0), \dot{z}(0), L(0), \dot{L}(0)]^\top = [1, 0, 0.75, 0]^\top$ and the input is bounded as $u \in [-10, 10]$. The objective is to maximize the following objective function

$$J = -(((z(t_f) - z(0))^2 + \dot{z}^2(t_f) + \int_0^{t_f} u^2 \mathrm{d}t),$$

where $t_f = 1.8$.

B. Performance Comparison

Fig. 3 presents the performance of RL and PRL under various hyperparameter settings. We fix the target network update rate to $\tau = 0.005$ and the variance of added Gaussian noise to $\sigma = 0.05$. The temperature of Gumbel softmax is denoted by T, and we adopt the "hard" setting here to induce a switched control policy. In the figure, each legend entry follows the format (algorithm, number of layers for actor a_l, number of nodes of each layer for actor-network a_{hid}, batch size (bs)).

As shown in Fig. 3(a), in the FO environment, PRL consistently converges to a stable reward after approximately 400 episodes across different hyperparameter settings, whereas RL exhibits convergence issues. In the MR environment (Fig. 3(b)), both algorithms reach a stable reward after about 300 episodes; however, PRL demonstrates greater robustness to hyperparameter variations compared to RL. In the Spring environment (Fig. 3(c)), PRL's learning curve exhibits lower variance than RL's, despite using only a single-layer actor network. Moreover, PRL's performance remains robust across different Gumbel temperatures, as illustrated in Fig. 3(d-f) for all environment settings.

VI. Conclusion

In this article, we proposed a piecewise reinforcement learning approach to address the optimal control problem for state-dependent switched systems. We theoretically analyzed the feasibility of the proposed idea and validated it across multiple examples. Our results show an improved performance of the proposed method compared to standard reinforcement learning. We also offered a theoretical explanation for why learning a piecewise policy is better for switched systems. Future work will extend to higher-dimensional systems involving contact dynamics.

References

[1] C. Liu and Z. Gong, *Optimal control of switched systems arising in fermentation processes (to appear)*. Springer Berlin, Heidelberg, 01 2014, vol. 97.

[2] S. Yuan, L. Zhang, O. Holub, and S. Baldi, "Switched adaptive control of air handling units with discrete and saturated actuators," *IEEE Control Systems Letters*, vol. 2, no. 3, pp. 417–422, 2018.

[3] J. T. Betts, *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming, Second Edition*, 2nd ed. Society for Industrial and Applied Mathematics, 2010. [Online]. Available: https://epubs.siam.org/doi/abs/10.1137/1.9780898718577

[4] H.-W. Park, P. Wensing, and S. Kim, "High-speed bounding with the mit cheetah 2: Control design and experiments," *The International Journal of Robotics Research*, vol. 36, p. 027836491769424, 03 2017.

[5] M. Egerstedt, "Behavior based robotics using hybrid automata," in *Hybrid Systems: Computation and Control*, N. Lynch and B. H. Krogh, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 103–116.

[6] M. S. Shaikh and P. E. Caines, "On relationships between weierstrass-erdmannn corner condition, snell's law and the hybrid minimum principle," in *2007 International Bhurban Conference on Applied Sciences Technology*, 2007, pp. 117–122.

[7] D. Stewart and M. Anitescu, "Optimal control of systems with discontinuous differential equations," *Numerische Mathematik*, vol. 114, pp. 653–695, 04 2012.

[8] U. Rosolia and A. Ames, "Iterative model predictive control for piecewise systems," *IEEE Control Systems Letters*, vol. PP, 04 2021.

[9] L. Arturo, *Optimal Control: An Introduction*. Birkhäuser, 2001.

[10] V. Azhmyakov, S. Attia, and J. Raisch, "On the maximum principle for impulsive hybrid systems," *Lect. Notes Comput. Sci.*, vol. 4981, pp. 30–42, 04 2008.

[11] A. Pakniyat and P. E. Caines, "On the hybrid minimum principle: The hamiltonian and adjoint boundary conditions," *IEEE Transactions on Automatic Control*, vol. 66, no. 3, pp. 1246–1253, 2021.

[12] M. S. Shaikh and P. E. Caines, "On the hybrid optimal control problem: Theory and algorithms," *IEEE Transactions on Automatic Control*, vol. 52, no. 9, pp. 1587–1603, 2007.

[13] M. Zhou and E. I. Verriest, "Generalized Euler-Lagrange equation: A challenge to Schwartz's distribution theory," *Proceedings of American Control Conference*, 2022.

[14] B. J. Driessen and N. Sadegh, "On the discontinuity of the costates for optimal control problems with coulomb friction," *Optimal Control Applications & Methods*, vol. 22, pp. 197–200, 2000.

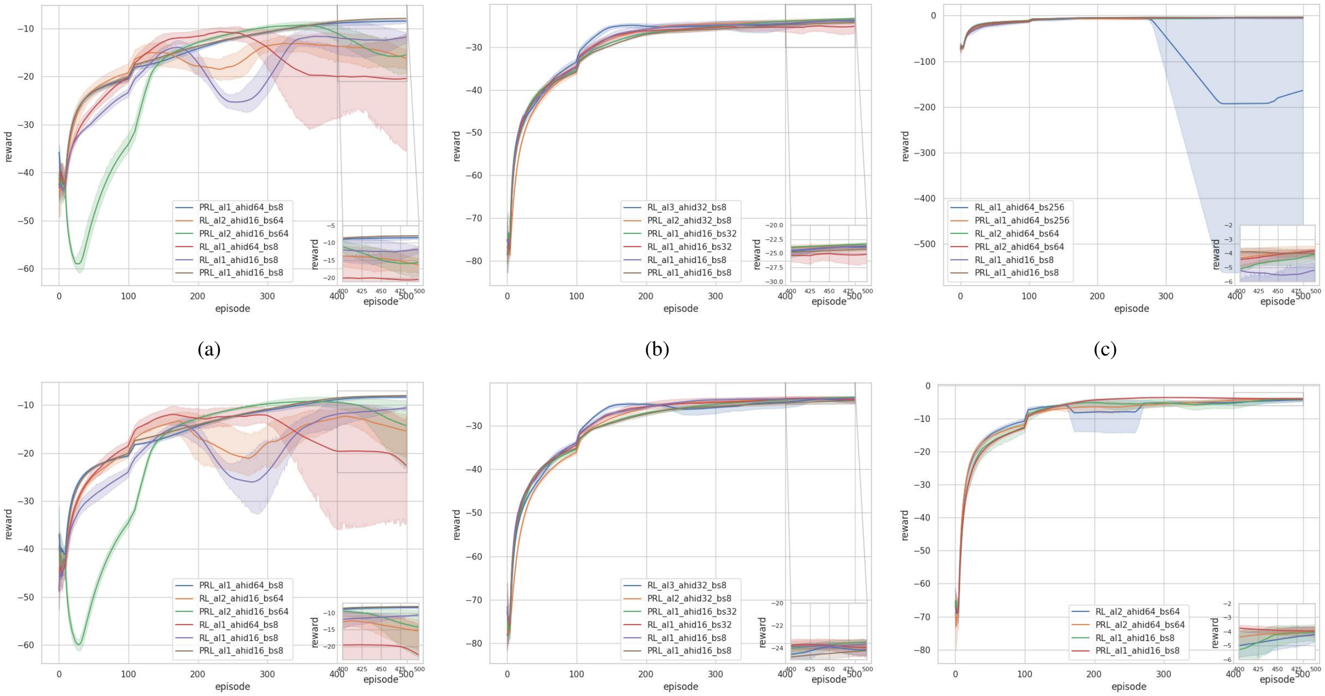
[15] P. Verheyen, *The economic explanation of the jump of the co-state variable*, ser. Research memorandum / Tilburg University, Department of Economics. Unknown Publisher, 1991, vol. FEW 491, pagination: 15, v.

[16] M. Zhou, E. I. Verriest, Y. Guan, and C. Abdallah, "Jump law of co-state in optimal control for state-dependent switched systems and applications," in *2023 American Control Conference (ACC)*, 2023, pp. 3566–3571.

[17] M. Zhou, E. Verriest, and C. Abdallah, "A model-free optimal control method," in *SoutheastCon 2024*, 2024, pp. 948–954.

[18] M. Peña, E. F. Camacho, S. Piñón, and R. Carelli, "Model predictive controller for piecewise affine system," *IFAC Proceedings Volumes*, vol. 38, no. 1, pp. 141–146, 2005, 16th IFAC World Congress. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1474667016368963

[19] M. Rungger and O. Stursberg, "A numerical method for hybrid optimal control based on dynamic programming," *Nonlinear Analysis: Hybrid Systems*, vol. 5, no. 2, pp. 254–274, 2011, special Issue related to IFAC Conference on Analysis and Design of Hybrid Systems (ADHS'09). [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1751570X10000683



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 3. J under 5 runs with randomly generated seed. (a) FO: $t_f = 1$, $T = 0.01$; (b) MR: $t_f = 2$, $T = 0.01$; (c) Spring: $t_f = 1.8$, $T = 0.01$; (d) FO: $t_f = 1$, $T = 0.001$; (e) MR: $t_f = 2$, $T = 0.001$; (f) Spring: $t_f = 1.8$, $T = 0.001$.

[20] ——, "A numerical method for hybrid optimal control based on dynamic programming," *Nonlinear Analysis: Hybrid Systems*, vol. 5, no. 2, pp. 254–274, 2011, special Issue related to IFAC Conference on Analysis and Design of Hybrid Systems (ADHS'09). [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1751570X10000683

[21] B. Passenberg, P. E. Caines, M. Sobotka, O. Stursberg, and M. Buss, "The minimum principle for hybrid systems with partitioned state space and unspecified discrete state sequence," in *49th IEEE Conference on Decision and Control (CDC)*, 2010, pp. 6666–6673.

[22] A. Schollig, P. E. Caines, M. Egerstedt, and R. Malhame, "A hybrid bellman equation for systems with regional dynamics," in *2007 46th IEEE Conference on Decision and Control*, 2007, pp. 3393–3398.

[23] M. L. Greene, M. Abudia, R. Kamalapurkar, and W. E. Dixon, "Model-based reinforcement learning for optimal feedback control of switched systems," in *2020 59th IEEE Conference on Decision and Control (CDC)*, 2020, pp. 162–167.

[24] X. Li, L. Dong, L. Xue, and C. Sun, "Hybrid reinforcement learning for optimal control of non-linear switching system," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 11, pp. 9161–9170, 2023.

[25] M. Wabartha and J. Pineau, "Piecewise linear parametrization of policies: Towards interpretable deep reinforcement learning," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: https://openreview.net/forum?id=hOMVq57Ce0

[26] J. Zhao and M. Gan, "Finite-horizon optimal control for continuous-time uncertain nonlinear systems using reinforcement learning," *International Journal of Systems Science*, vol. 51, no. 13, pp. 2429–2440, 2020. [Online]. Available: https://doi.org/10.1080/00207721.2020.1797223

[27] I. C.-D. Martino Bardi, *Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations*. Birkhäuser Boston, MA, 11 January 2008.

[28] M. Zhou, E. Verriest, and C. Abdallah, "A model-free optimal control method with fixed terminal states," in *SoutheastCon 2025*, 2025, pp. 649–655.

[29] C. T. H. Baker and P. A. Radcliffe, "Error bounds for some chebyshev methods of approximation and integration," *SIAM Journal on Numerical Analysis*, vol. 7, no. 2, pp. 317–327, 1970. [Online]. Available: http://www.jstor.org/stable/2949465

[30] C. Niu, H. Liao, H. Ma, and H. Wu, "Approximation properties of chebyshev polynomials in the legendre norm," *Mathematics*, vol. 9, no. 24, 2021. [Online]. Available: https://www.mdpi.com/2227-7390/9/24/3271

[31] P. Kidger and T. Lyons, "Universal approximation with deep narrow networks," 2020.

[32] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," 2019.

[33] H. J. T. Suh, M. Simchowitz, K. Zhang, and R. Tedrake, "Do differentiable simulators give better policy gradients?" in *International Conference on Machine Learning*, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:246472918

[34] M. Zhou, E. Verriest, and C. Abdallah, "A model-free optimal control method," in *SoutheastCon 2024*, 2024, pp. 948–954.

[35] T. Westenbroek, X. Xiong, A. D. Ames, and S. Shankar Sastry, "Optimal control of piecewise-smooth control systems via singular perturbations," in *2019 IEEE 58th Conference on Decision and Control (CDC)*, 2019, pp. 3046–3053.