Hybrid Mode Sample-Based Control

Yilang Liu
Department of Mechanical Engineering
Yale University
yilang.liu@yale.edu

Haoxiang You
Department of Mechanical Engineering
Yale University
haoxiang.you@yale.edu

Ian Abraham
Department of Mechanical Engineering
Yale University
ian.abraham@yale.edu

Abstract: In this work, we propose a novel sample-based approach for the manipulation and whole body control problem. Our method reduces the search space by explicitly accounting for control modes—namely when to apply them, how long to apply them, and which mode to use. We demonstrate that our approach can synthesize complex behaviors, ranging from in-hand manipulation to humanoid locomotion, and achieves significantly better performance in terms of total cost compared to existing methods. Finally, we validate our method on a physical Unitree A1 platform, highlighting its sample efficiency and real-world applicability.

Keywords: Whole-body Control, Locomotion, Manipulation

1 Introduction

Modern robotic systems (e.g., legged robots, dexterous hands) often need to switch between discrete modes, such as making and breaking contacts, to synthesize complex behaviors. Continuous control methods often struggle to solve tasks due to abrupt mode switches and a non-smooth optimization landscape. Hybrid mode control methods, on the other hand, address the non-smooth optimization by coordinating discrete mode switches with continuous control inputs, making them ideal for systems that undergo mode transitions [1, 2, 3]. However, scaling hybrid control to high-dimensional systems is challenging due to the exponential complexity of mode-switch optimization and the non-convex optimization landscape [4, 5]. As such, existing approaches typically rely on simplified models or exhaustive offline enumeration, limiting their applicability to real-world tasks [3, 6, 7].

In this work, we propose a sample-based approach to the whole-body control problem using hybrid mode control. Our method samples two critical decision variables: (1) **when** to switch modes and (2) **how long** to apply control. By focusing on these temporal control modes, we significantly reduce the search space for sampling control. Our approach outperforms existing sample-based continuous control methods [8, 9, 10] in terms of overall performance given limited samples, e.g., we achieve 85 % reduction of total cost in cube manipulation tasks. We validate our approach in real time on a physical robotic platform, showcasing its practical effectiveness for demanding tasks like quadrupedal locomotion. In summary, our contributions are: (1) A novel sample-based control strategy inspired by hybrid control theory; (2) Improved sample complexity and convergence rates against existing sample-based controls in legged locomotion and manipulation tasks; (3) Demonstration of real-time torque-level locomotion control of a Unitree A1 quadruped using our approach.

The rest of this paper is organized as follows: Section 2 provides related work for hybrid control and sample-based control. Section 3 introduces the optimal control problem and the sample-based control for the optimal control problem. Section 4 describes our sample-based hybrid mode control method for the optimal control problem shown in Section 3. Section 5 demonstrates the effectiveness

RSS 2025 Workshop on Whole-Body Control and Bimanual Manipulation (RSS2025WCBM).

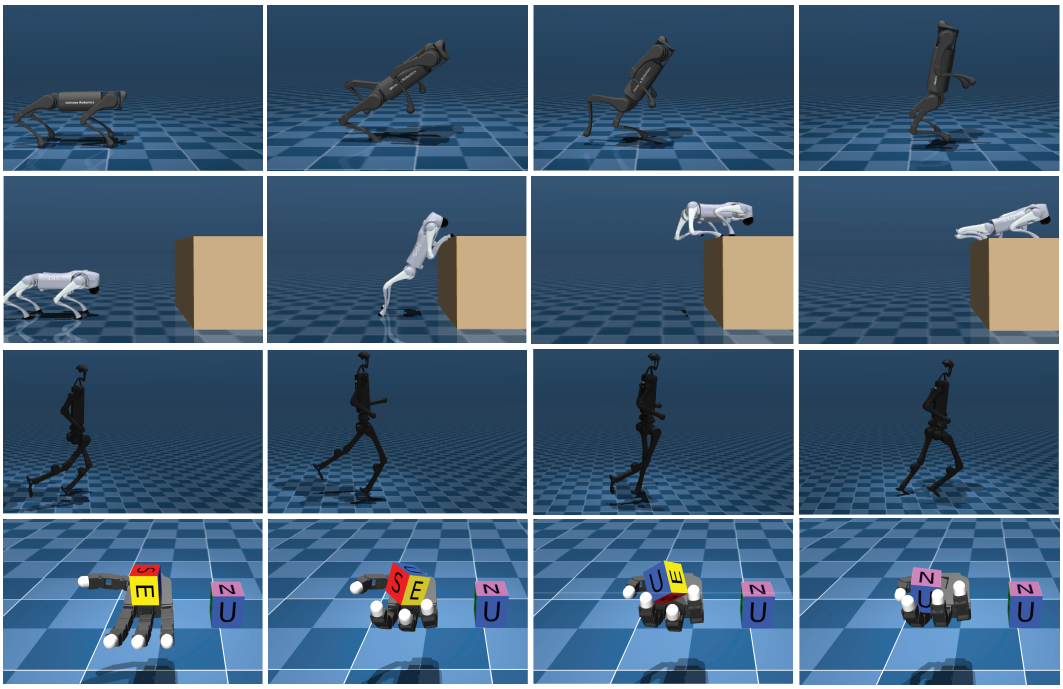

Figure 1: Examples of tasks completed using our method are illustrated through a series of time series images. **First row:** Unitree A1 performing bipedal locomotion. **Second row:** Unitree Go2 quadruped climbing a box. **Third row:** Unitree H1 humanoid locomotion. **Last row:** In-hand manipulation using the Allegro Hand.

of the proposed sample-based hybrid mode control through simulation and real-world experiments. Finally, the conclusion and limitation are presented in Section 6.

2 Related Work

2.1 Hybrid Control

Hybrid control addresses robotic systems that dynamically switch between discrete modes, such as making or breaking contacts in locomotion or transitioning between manipulation phases [3, 2, 1]. Hybrid control problems are typically solved by Linear Complementarity Programming [11, 12, 13] or Mixed-Integer Programming [14]. However, they struggle to scale to high-dimensional systems due to two key challenges: (1) the objective landscape is highly nonconvex [15, 3], and (2) the computational burden becomes intractable due to an increasing number of contact modes [16].

To mitigate these issues, prior works consider simplified models for high-dimensional robot systems[17, 18]. For example, a common, simple model for quadruped locomotion is the single-rigid-body model. This model simplification approximates robot dynamics by focusing on the center of mass and ground reaction force while ignoring joint-level states [19, 20, 21, 22]. However, using a simplified model loses the possibility to fully exploit the whole-body capability. Other methods predefine the order in which contacts are made and broken. In this case, the optimization problem becomes easy to solve [23]. However, predefined contact modes constrain the robot's behavior to a predetermined trajectory.

In this work, we proposed a sampling-based approach for solving the hybrid control problem. Our method does not rely on pre-specified modes or simplified models and can synthesize complex and agile behaviors. Moreover, by actively sampling the hybrid mode, our method synthesizes contact-rich behaviors and conducts computations online.

2.2 Sample-based Control

Sample-based control methods [24, 25, 26] have recently emerged as a simple yet effective approach for solving high-dimensional robotics tasks such as legged locomotion [27, 28, 29, 30] and manipulation [31, 32, 33, 34]. Rather than relying on explicit gradient-based techniques, sample-based controls begin by sampling a control sequence from an initial distribution, executing a forward rollout of that sequence, and then adjusting the sampling distribution based on the resulting costs. This gradient-free nature makes sample-based control particularly well-suited for complex, non-differentiable systems where traditional optimization methods may struggle.

Despite the effectiveness, sample-based control methods exhibit two limitations. First, they often treat control at each timestep as an independent variable, ignoring the inherent hybrid structure of robotic tasks [35]. Second, the number of samples required to adequately explore the control space scales exponentially with the planning horizon. This exponential growth makes sample-based methods computationally infeasible for long-horizon tasks.

Our work unifies the strengths of hybrid control and sample-based optimization. By reparameterizing the key decision variables, i.e., *when* and *how long* to apply control, our approach can discover complex behaviors in contact-rich scenarios, such as in-hand manipulation with the Allegro hand. Additionally, the number of decision variables in our formulation is independent of the time horizon, effectively reducing the search space for long-horizon tasks like in-hand manipulation. As a result, we effectively mitigate the exponential growth in sample requirements, making it feasible to handle both high-dimensional and long-horizon robotic tasks.

3 Preliminaries

This section presents the formulation of the optimal control problem and an overview of sample-based control.

3.1 Problem Formulation

Consider a discrete-time dynamical system:

$$\mathbf{x}_{t+1} = \mathbf{f}(\mathbf{x}_t, \mathbf{u}_t),\ t = 0, 1 \dots T-1, \tag{1}$$

where $\mathbf{x}_t \in \mathcal{X} \subseteq \mathbb{R}^n$ is the state and $\mathbf{u}_t \in \mathcal{U} \subseteq \mathbb{R}^m$ is the control at time step t. A trajectory $\{\mathbf{X}, \mathbf{U}\}$ contains the sequence of states $\mathbf{X} = [\mathbf{x}_0, \cdots, \mathbf{x}_T] \in \mathcal{X}^{0:T}$, and the sequence of controls $\mathbf{U} = [\mathbf{u}_0, \cdots, \mathbf{u}_{T-1}] \in \mathcal{U}^{0:T-1}$ satisfying (1). Here, $\mathcal{X}^{0:T} = \mathcal{X} \times \mathcal{X} \times \cdots (T \text{ times}) \times \mathcal{X}$ and $\mathcal{U}^{0:T-1} = \mathcal{U} \times \mathcal{U} \times \cdots (T-1 \text{ times}) \times \mathcal{U}$ are the cartesian product of domains.

We define the total cost $\mathcal{J}$ to be

$$\mathcal{J}(\mathbf{X}, \mathbf{U}) = \sum_{t=0}^{T-1} c\big(\mathbf{x}_t, \mathbf{u}_t\big) + c_f\big(\mathbf{x}_T\big), \tag{2}$$

where $c : \mathcal{X} \times \mathcal{U} \to \mathbb{R}$ is the cumulative sum of the running cost and $c_f : \mathcal{X} \times \mathcal{U} \to \mathbb{R}$ is the terminal cost. Since we can recover states $\mathbf{X}$ from integration using Eq.(1), we can simplify the total cost $\mathcal{J}(\mathbf{x}_0, \mathbf{U})$ as functions of only initial condition $\mathbf{x}_0$ and sequence of controls $\mathbf{U}$.

The goal of the optimal control problem is to find the control sequence such that the objective $\mathcal{J}(\mathbf{x}_0, U)$ is minimized:

$$\begin{aligned} \mathbf{U}^\star = &\underset{\mathbf{U} \in \mathcal{U}^{0:T-1}}{\arg\min}\ \mathcal{J}(\mathbf{x}_0, \mathbf{U}) \\ \text{s.t.} \quad &\mathbf{x}_{t+1} = \mathbf{f}(\mathbf{x}_t, \mathbf{u}_t),\ t = 0, 1 \dots T-1, \end{aligned} \tag{3}$$

3.2 Sample-based Predictive Control

Sampling-based methods, such as evolutionary algorithms [36] and the cross-entropy methods [37, 10], are widely used to address nonconvex optimization problems. These approaches iteratively

perturb candidate solutions and selectively retain high-performing candidates based on a predefined metric. In the context of optimal control, such a process can be formulated as follows:

$$\Delta \mathbf{U}^\star = \underset{\Delta \mathbf{U}^{(i)} \in \mathcal{D}}{\arg\min}\, \mathcal{J}\left(\mathbf{x}_0, \mathbf{U}^{\text{nom}} + \Delta \mathbf{U}^{(i)}\right) \tag{4}$$

$$\mathbf{U}^{\text{nom}} \leftarrow \mathbf{U}^{\text{nom}} + \Delta \mathbf{U}^\star. \tag{5}$$

Here, $\mathbf{U}^{\text{nom}}$, the nominal control sequence, is the candidate solution to (3). The set

$$\mathcal{D} = \{\Delta \mathbf{U}^{(1)}, \Delta \mathbf{U}^{(2)}, \dots \Delta \mathbf{U}^{(N)}\}, \tag{6}$$

comprises N control perturbations. In this context, each perturbation $\Delta \mathbf{U}^{(i)}$, drawn from the distribution $\rho(\Delta U)$ over $\mathcal{U}^{0:T}$, is resampled at every iteration. A common choice of ρ is a multivariate Gaussian distribution $\mathcal{N}(\mu, \Sigma)$, where $\mu \in \mathbb{R}^{m\times(T-1)}$ is mean perturbation sequence, and $\Sigma \in \mathbb{R}^{m(T-1)\times m(T-1)}$ is the covariance matrix.

More sophisticated algorithms, such as Model Predictive Path Integral (MPPI) control [8], also iteratively adapt μ and Σ during optimization to concentrate sampling in high-performance regions of the control space. A central challenge in sampling-based control is the exponential growth of the search space $\mathcal{U}^{0:T}$ with respect to the horizon T. To remedy this, a sampling-based approach is commonly embedded into a predictive-control framework [9], which iteratively solves shorter-horizon subproblems over a receding window $[t_0 : t_0 + H]$:

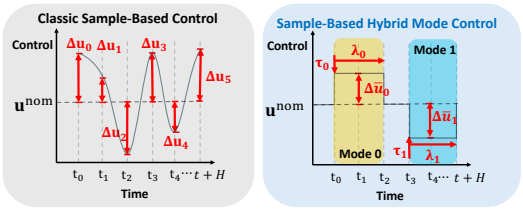

Figure 2: **Summary of differences between our methods and existing sampling method** The red lines denote the decision variables. Existing methods sample the controls at each time step. In contrast, our method considers when, how long, and how much to apply control for each mode.

$$\mathbf{U}^*_{[t_0:t_0+H]} = \underset{\mathbf{U}_{[t_0:t_0+H]}}{\arg\min}\, \mathcal{J}_H\left(\mathbf{x}_0, \mathbf{U}_{[t_0:t_0+H]}\right), \tag{7}$$

where $H \ll T$ and $\mathbf{U}_{[t_0:t_0+H]} = [\mathbf{u}_{t_0}, \dots, \mathbf{u}_{t_0+H}]$ is the short-horizon control sequence. The truncated cost $\mathcal{J}_H$ is defined as:

$$\mathcal{J}_H\left(\mathbf{x}_{t_0}, \mathbf{U}_{[t_0:t_0+H]}\right) = \sum_{t=t_0}^{t_0+H} c(\mathbf{x}_t, \mathbf{u}_t) + c_H(\mathbf{x}_{t_0+H+1}). \tag{8}$$

After solving the short-horizon subproblem (7), the first control input of the optimized sequence, i.e., $\mathbf{u}^*_{t_0}$, is applied. This process repeats as the system dynamics evolve.

The performance of the predictive-control framework is acutely sensitive to the selected short-horizon length. If H is too small, the system may exhibit myopic behavior, becoming trapped in local minima. Conversely, increasing H broadens the search space and introduces significant computational challenges—a critical consideration for online deployments.

Essentially, we require a sampling scheme that remains computationally feasible even for longer horizons—that is, the search space does not expand prohibitively as the horizon increases. Moreover, it should leverage the hybrid nature of many problems to synthesize complex behavior.

4 Sample-based Hybrid Mode Control

Here, we introduce our sample-based hybrid mode control method. A key observation is that in many optimal control problems, especially those involving hybrid dynamics, the control strategy can be segmented. That is, the shift of control is only made when necessary (see [38, 39] or our toy example in Figure 3). Consequently, rather than sampling a control perturbation at every time step t, it may be more effective to sample control segments, i.e., the control modes.

Specifically, we partition the control perturbation $\mathbf{U}_{[t_0:t_0+H]}$ into K distinct modes denoted as $\mathbf{M}_k$, where $k = 1, \ldots, K$. Each mode is characterized by three decision variables: τ_k, the time at which the mode begins; λ_k, the duration of the mode; and $\Delta\bar{\mathbf{u}}_k$, the control perturbation being applied during the mode. We define the collection of these K modes as $\mathcal{M} = \{\mathbf{M}_1, \ldots, \mathbf{M}_K\}$.

Here, each mode $\mathbf{M}_k = \{\tau_k, \lambda_k, \Delta\bar{\mathbf{u}}_k\}$ consists of a piecewise constant control perturbation $\Delta\bar{\mathbf{u}}_k$ applied at discrete time $t_0 \leq \tau_k \leq t_0 + H$ for a duration $\lambda_k \leq H - \tau_k$.

At each iteration of our approach, we uniformly sample a set of K modes that hold these three discrete values. Each mode augments the nominal control sequence $\mathbf{U}^{\text{nom}}$ with $\Delta\mathbf{U}^k_{t_0:t_0+H}$ where

$$\Delta\mathbf{U}^k_t = \begin{cases} \Delta\bar{\mathbf{u}}_k \text{ for } t \in [\tau_k, \tau_k + \lambda_k] \\ \mathbf{0}, \text{ for } t \notin [\tau_k, \tau_k + \lambda_k] \end{cases}, \tag{9}$$

is the synthesized control. The sub-optimization problem for finding kth mode $\mathbf{M}_k$ can be described as

$$\mathbf{M}^*_k = \{\tau^\star_k, \lambda^\star_k, \Delta\bar{\mathbf{u}}^\star_k\} = \underset{\tau^{(i)}_k, \lambda^{(i)}_k, \Delta\bar{\mathbf{u}}^{(i)}_k \in \mathcal{E}_k}{\arg\min} \mathcal{J}(\mathbf{x}_{t_0}, \mathbf{U}^{\text{nom}}_{t_0:t_0+H} + \Delta\mathbf{U}^k_{t_0:t_0+H}),$$

where $\mathcal{E}_k = \{\mathbf{M}^{(1)}_k, \cdots \mathbf{M}^{(N)}_k\}$ is the set contains N samples of $\mathbf{M}_k$. Here, each sample $\mathbf{M}^{(i)}_k = \{\tau^{(i)}_k, \lambda^{(i)}_k, \Delta\bar{\mathbf{u}}^{(i)}\}$ is drawn independently from joint distribution $\rho(\tau_k, \lambda_k, \Delta\bar{\mathbf{u}_k})$. In practice, we sample τ_k uniformly from $[t_0, t_0 + H]$ and λ_k uniformly from $[0, H - \tau_k]$. Additionally, the perturbation $\Delta\bar{\mathbf{u}} \sim \mathcal{N}(\mathbf{0}, \Sigma)$ is sampled independent of both τ_k and λ_k.

The optimal mode sequences $\mathcal{M}^\star \leftarrow \cup\mathbf{M}^*_k$ are computed sequentially by solving Eq. (10), that is, we first solve $\mathbf{M}^\star_k$, update the nominal control via Eq. (9), and then solve for $\mathbf{M}^\star_{k+1}$. We summarize the key differences between our sampling method and previous approaches in Figure 2, while a detailed description of our method within the predictive control framework is provided in Algorithm 1.

Algorithm 1 Hybrid Mode Control

Initialize: nominal control plan $\mathbf{U}^{\text{nom}}_{t:t+H}$, sequence of modes $\mathcal{M}$ with length K, horizon H, current time t, final time T, number of samples N.
while $t \leq T$ **do**
  Observe State $\mathbf{x}_t$
  for $k = 1, \ldots, K$ **do**
    for $i = 1, \cdots, N$ **do**
      Sample $\mathbf{M}^{(i)}_k = \{\tau^{(i)}_k, \lambda^{(i)}_k, \Delta\bar{\mathbf{u}}^{(i)}_k\}$
      Compute $\Delta\mathbf{U}^{k,(i)}_{[t:t+H]}$ via Eq. (9)
      Compute $\mathcal{J}^{(i)}(\mathbf{x}_0, \mathbf{U}^{\text{nom}}_{t:t+H} + \Delta\mathbf{U}^{k,(i)}_{t:t+H})$
    end for
    Compute $\mathbf{M}^*_k$ and $\Delta\mathbf{U}^{k,*}_{t:t+H}$ via Eq. (10)
    Update $\mathbf{U}^{\text{nom}}_{t:t+H} \leftarrow \mathbf{U}^{\text{nom}}_{t:t+H} + \Delta\mathbf{U}^{k,*}_{t:t+H}$
  end for
  Apply first control $\mathbf{U}^{\text{nom}}_t$ to the robot.
  Shift nominal control $\mathbf{U}^{\text{nom}}_{t:H-1} \leftarrow \mathbf{U}^{\text{nom}}_{t+1:H}$
  $t \leftarrow t + 1$
end while

The key advantage of our sampling method is that the search space does not grow with the predictive horizon H, as the time horizon itself is treated as a decision variable. To formalize this, consider discretizing the action space $\mathcal{U}$ into N_u control options, i.e., $\mathcal{U} = \{\bar{\mathbf{u}}^{(0)}, \cdots, \bar{\mathbf{u}}^{(N_u)}\}$ where each $\bar{\mathbf{u}} \in \mathbb{R}^m$. In classical sampling-based methods, the sampling complexity grows exponentially as $\mathcal{O}(N_u^H)$. In contrast, our approach has a sampling complexity of $\mathcal{O}\big((N_u H^2)^K\big) = \mathcal{O}\big(N_u^K H^{2K}\big)$, where K denotes the number of mode switches and the factor H^2 arises from the two additional temporal variables τ and λ. Since $K \ll H \ll N_u$ in practice, our method effectively reduces the exponential growth in sampling complexity.

By eliminating the exponential dependence on H, our method enables planning over significantly longer horizons with limited computational resources. We illustrate this advantage with a toy example below and present more complex behaviors in Section 5.

Toy Example: Cartpole Swing Up

Here, we consider a cartpole swing-up task. We set the total horizon $T = 100$. The state is defined as pole angle θ, cart position p, pole angular velocity $\dot{\theta}$, and cart linear velocity $\dot{p}$. The control is the one-dimensional force applied to the cart. The pole is initialized in up-down $\theta = \frac{1}{2}\pi$, and

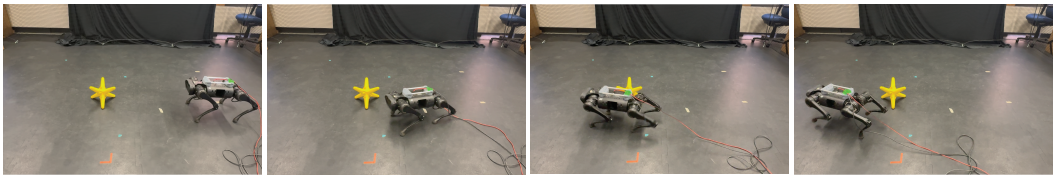

Figure 4: **Hardware Verification** We verify our method in hardware experiments using the Unitree A1 quadruped. From left to right, the Unitree A1 walks autonomously to avoid incoming obstacles. We update the control sequence in real time on a single CPU with onboard sensing, demonstrating the computation efficiency of our method.

the goal is to swing the pole upright and maintain balance. Hence, the stage cost is defined as $c(\mathbf{x}_t, \mathbf{u}_t) = 4.0(\cos(\theta) - 1)^2 + 0.1p^2 + 0.1(\dot{\theta}^2 + \dot{p}^2) + \mathbf{u}^2$ and the terminal cost is defined as $c_f(\mathbf{x}_T) = 4.0(\cos(\theta) - 1)^2$

Figure 5 demonstrates the results for our method and other sampling approaches, both for the receding-horizon sub-optimization problem and the overall performance under a predictive-sampling framework. To ensure scale invariance across tasks, we normalize this gap by the planning horizon length H. Here error bar denotes the standard deviation across 5 random seeds. The performance of classical sampling methods deteriorates as the horizon length increases, whereas our method consistently finds good minima, even with a limited sample size.

Both methods show a decrease in cumulative cost as the horizon increases, as a longer horizon reduces myopic decision-making. However, as the horizon continues to grow, classical methods experience an increase in total cost due to their inability to effectively optimize the sub-problem, whereas our method continues to decrease the overall cost.

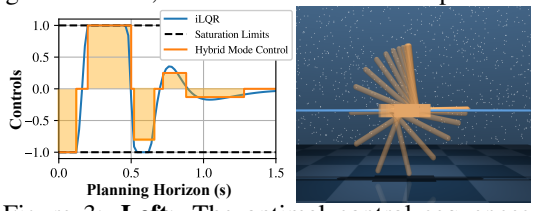

Figure 3: **Left**: The optimal control sequences found for the cartpole system are shown. The blue curve represents the control generated by the iLQR algorithm, a gradient-based approach that guarantees convergence to a minimum. The orange curve represents the control found using our sample-based hybrid control method, where the shaded areas denote distinct modes. Here, we use $K = 4$ distinct modes. **Right**: The visualization of the cartpole trajectory with controls found by our method.

Our method consistently finds a good minimum for the receding-horizon sub-problem by actively considering time as a decision variable. In contrast, the classical sampling approach fails to find a good optimum as the horizon increases due to the expanding search space. As for the overall performance under a predictive-control framework, both methods show an improvement due to reduced myopia. However, the performance of classical methods decreases with a further increased horizon, as they struggle to solve the receding-horizon sub-problem effectively with a limited sample size, whereas our method consistently improves performance. We also compare the control sequence found by our method to the gradient-based iLQR method, which at least guarantees a local optimum, in Figure 3. The control sequence found by our method closely resembles the optimal iLQR sequence, even with a few modes.

5 Experiments and Results

We designed the experiments to answer the following three questions: (1) Does our method scale up to high-dimensional tasks? (2) How does our method compare to other sample-based controls in terms of cost reduction? (3) Can our method be deployed on physical robots with limited computational resources?

5.1 Scalability to High-dimensional Task

Here, we start with simulation experiments to demonstrate scalability to high-dimensional tasks shown in Figure 1. We evaluate our method against a set of robotic platforms and tasks in simulations. The goal of this section is to demonstrate that our method can work in various high-dimensional robotic systems. First, we design quadruped locomotion tasks: bipedal walking and box climbing. Second, we design a humanoid jogging experiment, and the humanoid is tasked to track a desired forward velocity while maintaining an upright orientation. Finally, we showcase in-hand manipulation experiments, and the robot hand is tasked to reorient the cube to the target orientation shown on right side. Each experiment can successfully track the target position and orientation while completing the tasks without any predefined gait. The implementation details can be found in Table 1.

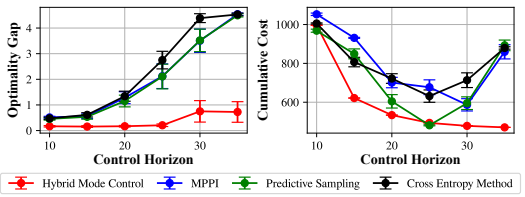

Figure 5: **Top**: Results for the short-horizon subproblem of the cartpole system. The Y-axis quantifies the optimality gap, defined as the difference between the objective value and a locally optimal solution obtained via iterative LQR. **Bottom**: Overall performance under a predictive control framework.

Quadruped Locomotion: The quadruped example uses the Unitree robot. Here, we consider two tasks: bipedal walking and box climbing. The dimension for quadruped state space $x \in \mathbb{R}^{37}$ and positional control inputs $u \in \mathbb{R}^{12}$. Here, state x contains the base position, base orientation, joint angle, linear, angular velocity, and joint velocity. Control inputs u contain robot joint angles. For bipedal walking, given the initial horizon trunk orientation, the goal is to reorient its body to an upright position without falling with $T = 200(4s)$ steps. The stage cost is defined as weighted sum of state deviation and input: $||\mathbf{x} - \mathbf{x}_{ref}||^{\top}_{\mathbf{Q}} + ||\mathbf{u} - \mathbf{u}_{ref}||^{\top}_{\mathbf{R}}$ where $\mathbf{Q}$ and $\mathbf{R}$ are weights. We use the number of samples $N = 30$, and prediction horizon $H = 40\,(0.8s)$. The bipedal locomotion poses a unique challenge since the controller needs to quickly saturate the torque limit to push the body upwards successfully. Additionally, we show that the quadruped can successfully jump on the box given the terminal sitting position. Here, the stage cost is $\mathbf{x} - \mathbf{x}_{ref}||^{\top}_{\mathbf{Q}} + 0.1 \sum f_{\text{contact}}$ where x_{ref} is the fixed terminal target pose, f_{contact} is the contact force at each foot. We use the number of samples $N = 100$, and prediction horizon $H = 50\,(1.0s)$. It shows that our methods can generate a climbing control sequence without a predefined gait.

Humanoid Jogging: In this humanoid example, we need to deal with a higher number of state and control dimensions. Specifically, we use the Unitree H1 humanoid robot as an example where the state space $x \in \mathbb{R}^{51}$ contains position, orientation, and velocity for hips, knees, torso, and shoulders. Control space is $u \in \mathbb{R}^{19}$ We use the number of samples $N = 1000$, and prediction horizon $H = 50\,(1.0s)$. The goal of the humanoid robot is to track the desired torso linear velocity $1.0\,m/s$ without falling. Here, the stage cost is defined as $0.5 \sum ||\theta_{\text{torso}}||^2 + 0.5||h_{\text{torso}} - 1.3||^2 + ||v_{\text{torso}} - 1.0||^2 + 0.01||\mathbf{u}||_2^2$, where θ_{torso} is the angle of torso measured from upright position, h_{torso} is the height of torso, and v_{torso} is the forward velocity of the torso. Without any predefined locomotion gait, this example requires the controller to generate a periodic jogging gait while maintaining whole-body stability. From Figure 1, we see that the hybrid mode control can successfully control the humanoid jogging forward by periodically actuating either leg.

Allegro In-Hand Manipulation: In this example, we present Allegro in-hand manipulation of a colored cube—a *contact-rich*, *long-horizon* task that requires all fingers to anticipate their movements and maintain frequent contact with the cube. The goal is to grasp and reorient the cube to the target orientation and position without falling. The state space $x \in \mathbb{R}^{52}$ includes the finger poses and velocities of the Allegro hand as well as the pose of the cube. Control space $u \in \mathbb{R}^{16}$ includes the positional angle of the fingers. Here, the stage cost is defined as $2.5||\mathbf{R}^T_{\text{target}}\mathbf{R}_{\text{cube}}||_2^2 + 5||\mathbf{v}_{\text{cube}}||_2^2 + 0.05||\mathbf{u}||_2^2 + 1.25||\mathbf{q}_{\text{hand}} - \mathbf{q}_{\text{target}}||_2^2 + 0.0005||\mathbf{v}_{\text{hand}}||_2^2$, where $||\mathbf{R}^T_{target}\mathbf{R}_{cube}||_2^2$ represents rotation error between cube to target pose, $\mathbf{v}_{cube}$ is the velocity of the cube, $||\mathbf{q}_{\text{hand}} - \mathbf{q}_{\text{target}}||$ measures the joint deviation between hands postured to target and $\mathbf{v}_{\text{hand}}$ is the joint velocity of hands. Hybrid mode control samples finger positional angles that best minimize the tracking loss. For the purpose of illustration, we add the floating cube, demonstrating the target cube

orientation. From Figure 1, we see that the hybrid mode control can successfully control the hand toward the target orientation of the cube, planning contact-rich interactions with the cube's surface.

5.2 Comparison to Existing sample-based controls.

To demonstrate the advantage of our method, we compare the total performance in terms of cumulative cost against existing sample-based methods. The baselines are MPPI [8], cross entropy method [10], and predictive sampling [9]. We use the same number of samples, sampling covariance, and control horizon across the sampling-based methods to ensure fair comparisons. All sample-based control methods share the same objectives and random seeds. We fix the temperature coefficient of MPPI to be 0.1 across all three tasks. The detailed control parameters are shown in Table 1. We report the simulation results in Table 2.

Table 1: Task Implementation Details for Examples Presented in This Work.

Task Name	Samples Size N	Horizon H	Modes K
Bipedal Walking	30	40 (0.8s)	2
Crate Climbing	100	50 (1.0s)	2
Humanoid Jogging	1000	50 (1.0s)	10
In-hand Manipulation	1000	125 (5s)	5

For all three locomotion and manipulation tasks, our methods outperform existing sample-based methods by 35 $\%$ in bipedal walking tasks, 67 $\%$ in box climbing, 41 $\%$ in humanoid jogging tasks, and 85 $\%$ in hand manipulation tasks. Notice that only our methods can successfully balance the quadruped in the bipedal walking task.

Table 2: Performance Comparison over three tasks.

	Bipedal Walking	Box Climbing	Humanoid Jogging	In-Hand Manipulation
PS [9]	279	208	338	272
MPPI [8]	282	231	480	292
CEM [10]	265	216	456	280
Ours	**173**	**70**	**198**	**40**

5.3 Hardware Experiments

We further validate our method in real-world experiments using the quadrupedal robot Unitree A1. As shown in Fig. 4, we successfully deploy our sample-based control for autonomous object avoidance.

Specifically, we model a quadruped's state $\mathbf{x} \in \mathbb{R}^{37}$ as base position, orientation, joint angle, base linear velocity, base angular velocity, and joint velocity. The control is the target joint angle, which is tracked by a PD controller running at 1 kHz. We run the proposed method at 100 Hz on a single Intel i7-12700H CPU with 32 GB of memory. During each loop, we sample $N = 30$ control perturbations, where each sample is $H = 40$ control horizon and is synthesized by $K = 2$ control modes. Additionally, we implement state estimation based on an extended Kalman filter using onboard sensing [40]. In comparison with those existing sample-based control frameworks [27, 28] relying on a highly accurate mocap system, our method only uses onboard sensing, demonstrating robustness under noisy state measurements.

6 Conclusions

This work presents a sample-based hybrid mode control method inspired by hybrid control theory. Our approach offers an alternative modeling scheme for sampling within a predictive-control framework. By introducing time as a decision variable, our method can effectively be applied to longer-horizon tasks. Moreover, by actively considering the hybrid nature of control, our method can synthesize complex behavior for high-dimensional systems. There are several limitations and

future directions to consider. Sample-based control typically requires an accurate model—one can only simulate what is well-represented in that model. This reliance on a precise model constrains real-world applications, particularly in unstructured environments or scenarios where obtaining a reliable model is challenging, such as turbulence control in fluid dynamics. Future research directions include integrating our methods with data-driven approaches that do not require explicit modeling. This integration can be achieved by either learning a residual dynamical model [41] or implementing direct control in the learned latent space [42, 43].

References

[1] J. H. Park and H. Chung. Hybrid control for biped robots using impedance control and computed-torque control. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, volume 2, pages 1365–1370 vol.2, 1999. doi:10.1109/ROBOT.1999.772551.

[2] A. D. Ames. Human-inspired control of bipedal walking robots. *IEEE Transactions on Automatic Control*, 59(5):1115–1130, 2014.

[3] A. R. Ansari and T. D. Murphey. Sequential action control: Closed-form optimal control for nonlinear and nonsmooth systems. *IEEE Transactions on Robotics*, 32(5):1196–1214, Oct. 2016. ISSN 1941-0468. doi:10.1109/tro.2016.2596768. URL http://dx.doi.org/10.1109/TRO.2016.2596768.

[4] J. Carpentier and N. Mansard. Multicontact locomotion of legged robots. *IEEE Transactions on Robotics*, 34(6):1441–1460, 2018. doi:10.1109/TRO.2018.2862902.

[5] J. Carpentier, R. Budhiraja, and N. Mansard. Learning feasibility constraints for multicontact locomotion of legged robots. In *Robotics: Science and Systems*, page 9p, 2017.

[6] H. Geyer, A. Seyfarth, and R. Blickhan. Compliant leg behavior explains basic dynamics of walking and running. *Proceedings. Biological sciences / The Royal Society*, 273:2861–7, 08 2006. doi:10.1098/rspb.2006.3637.

[7] M. Srinivasan and A. Ruina. Computer optimization of a minimal biped model discovers walking and running. *Nature*, 439:72–5, 02 2006. doi:10.1038/nature04113.

[8] G. Williams, P. Drews, B. Goldfain, J. M. Rehg, and E. A. Theodorou. Information theoretic model predictive control: Theory and applications to autonomous driving, 2017. URL https://arxiv.org/abs/1707.02342.

[9] T. Howell, N. Gileadi, S. Tunyasuvunakool, K. Zakka, T. Erez, and Y. Tassa. Predictive sampling: Real-time behaviour synthesis with mujoco, 2022. URL https://arxiv.org/abs/2212.00541.

[10] S. Mannor, R. Rubinstein, and Y. Gat. The cross entropy method for fast policy search. *Proceedings, Twentieth International Conference on Machine Learning*, 2, 07 2003.

[11] M. Posa, C. Cantu, and R. Tedrake. A direct method for trajectory optimization of rigid bodies through contact. *The International Journal of Robotics Research*, 33(1):69–81, 2014. doi:10.1177/0278364913506757. URL https://doi.org/10.1177/0278364913506757.

[12] A. Patel, S. L. Shield, S. Kazi, A. M. Johnson, and L. T. Biegler. Contact-implicit trajectory optimization using orthogonal collocation. *IEEE Robotics and Automation Letters*, 4 (2):2242–2249, Apr. 2019. ISSN 2377-3774. doi:10.1109/lra.2019.2900840. URL http://dx.doi.org/10.1109/LRA.2019.2900840.

[13] G. Kim, D. Kang, J.-H. Kim, S. Hong, and H.-W. Park. Contact-implicit model predictive control: Controlling diverse quadruped motions without pre-planned contact modes or trajectories. *The International Journal of Robotics Research*, Oct. 2024. ISSN 1741-3176. doi:10.1177/02783649241273645. URL http://dx.doi.org/10.1177/02783649241273645.

[14] P. Belotti, C. Kirches, S. Leyffer, J. Linderoth, J. Luedtke, and A. Mahajan. Mixed-integer nonlinear optimization. *Acta Numerica*, 22:1–131, 2013. doi:10.1017/S0962492913000032.

[15] A. W. Winkler, C. D. Bellicoso, M. Hutter, and J. Buchli. Gait and trajectory optimization for legged systems through phase-based end-effector parameterization. *IEEE Robotics and Automation Letters*, 3(3):1560–1567, 2018. doi:10.1109/LRA.2018.2798285.

[16] B. Aceituno-Cabezas, C. Mastalli, H. Dai, M. Focchi, A. Radulescu, D. G. Caldwell, J. Cappelletto, J. C. Grieco, G. Fernandez-Lopez, and C. Semini. Simultaneous contact, gait and motion planning for robust multi-legged locomotion via mixed-integer convex optimization. *IEEE Robotics and Automation Letters*, page 1–1, 2017. ISSN 2377-3774. doi:10.1109/lra.2017.2779821. URL http://dx.doi.org/10.1109/LRA.2017.2779821.

[17] P.-B. Wieber. *Holonomy and Nonholonomy in the Dynamics of Articulated Motion*, volume 340, pages 411–425. Lecture Notes in Control and Information Sciences, 07 2007. ISBN 978-3-540-36118-3. doi:10.1007/978-3-540-36119-0_20.

[18] P. Wensing and D. Orin. Improved computation of the humanoid centroidal dynamics and application for whole-body control. *International Journal of Humanoid Robotics*, 13:1550039, 09 2015. doi:10.1142/S0219843615500395.

[19] J.-P. Sleiman, F. Farshidian, M. V. Minniti, and M. Hutter. A unified mpc framework for whole-body dynamic locomotion and manipulation, 2021. URL https://arxiv.org/abs/2103.00946.

[20] J. Di Carlo, P. M. Wensing, B. Katz, G. Bledt, and S. Kim. Dynamic locomotion in the mit cheetah 3 through convex model-predictive control. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–9, 2018. doi:10.1109/IROS.2018.8594448.

[21] Y. Ding, A. Pandala, C. Li, Y.-H. Shin, and H.-W. Park. Representation-free model predictive control for dynamic motions in quadrupeds. *IEEE Transactions on Robotics*, 37(4):1154–1171, Aug. 2021. ISSN 1941-0468. doi:10.1109/tro.2020.3046415. URL http://dx.doi.org/10.1109/TRO.2020.3046415.

[22] Z. Zhou, B. Wingo, N. Boyd, S. Hutchinson, and Y. Zhao. Momentum-aware trajectory optimization and control for agile quadrupedal locomotion, 2022. URL https://arxiv.org/abs/2203.01548.

[23] P. Fernbach, S. Tonneau, O. Stasse, J. Carpentier, and M. Taïx. C-croc: Continuous and convex resolution of centroidal dynamic trajectories for legged robots in multicontact scenarios. *IEEE Transactions on Robotics*, 36(3):676–691, 2020. doi:10.1109/TRO.2020.2964787.

[24] G. Williams, A. Aldrich, and E. Theodorou. Model predictive path integral control using covariance variable importance sampling, 2015. URL https://arxiv.org/abs/1509.01149.

[25] D. Wierstra, T. Schaul, J. Peters, and J. Schmidhuber. Natural evolution strategies. In *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*, pages 3381–3387, 2008. doi:10.1109/CEC.2008.4631255.

[26] P. I. Frazier. A tutorial on bayesian optimization, 2018. URL https://arxiv.org/abs/1807.02811.

[27] H. Xue, C. Pan, Z. Yi, G. Qu, and G. Shi. Full-order sampling-based mpc for torque-level locomotion control via diffusion-style annealing, 2024. URL https://arxiv.org/abs/2409.15610.

[28] J. Alvarez-Padilla, J. Z. Zhang, S. Kwok, J. M. Dolan, and Z. Manchester. Real-time whole-body control of legged robots with model-predictive path integral control, 2024. URL https://arxiv.org/abs/2409.10469.

[29] G. Turrisi, V. Modugno, L. Amatucci, D. Kanoulas, and C. Semini. On the benefits of gpu sample-based stochastic predictive controllers for legged locomotion, 2024. URL https://arxiv.org/abs/2403.11383.

[30] Z. Yi, C. Pan, G. He, G. Qu, and G. Shi. Covo-mpc: Theoretical analysis of sampling-based mpc and optimal covariance design, 2024. URL https://arxiv.org/abs/2401.07369.

[31] T. Pang, H. J. T. Suh, L. Yang, and R. Tedrake. Global planning for contact-rich manipulation via local smoothing of quasi-dynamic contact models. *IEEE Transactions on Robotics*, 39(6): 4691–4711, 2023. doi:10.1109/TRO.2023.3300230.

[32] G. Rizzi, J. J. Chung, A. Gawel, L. Ott, M. Tognon, and R. Siegwart. Robust sampling-based control of mobile manipulators for interaction with articulated objects. *IEEE Transactions on Robotics*, 39(3):1929–1946, 2023. doi:10.1109/TRO.2022.3233343.

[33] I. Abraham, A. Handa, N. D. Ratliff, K. Lowrey, T. D. Murphey, and D. Fox. Model-based generalization under parameter uncertainty using path integral control. *IEEE Robotics and Automation Letters*, 5:2864–2871, 2020. URL https://api.semanticscholar.org/CorpusID:212635299.

[34] M. Bhardwaj, B. Sundaralingam, A. Mousavian, N. Ratliff, D. Fox, F. Ramos, and B. Boots. Storm: An integrated framework for fast joint-space model-predictive control for reactive manipulation, 2021. URL https://arxiv.org/abs/2104.13542.

[35] L. Liu, K. Yin, M. Van de Panne, T. Shao, and W. Xu. Sampling-based contact-rich motion control. In *ACM SIGGRAPH 2010 papers*, pages 1–10. Association for Computing Machinery, 2010.

[36] T. Salimans, J. Ho, X. Chen, S. Sidor, and I. Sutskever. Evolution strategies as a scalable alternative to reinforcement learning, 2017. URL https://arxiv.org/abs/1703.03864.

[37] P.-T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein. A tutorial on the cross-entropy method. *Annals of operations research*, 134:19–67, 2005.

[38] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, 1999.

[39] M. S. Branicky. Multiple lyapunov functions and other analysis tools for switched and hybrid systems. *IEEE Transactions on automatic control*, 43(4):475–482, 1998.

[40] M. Blösch, M. Hutter, M. A. Höpflinger, S. Leutenegger, C. Gehring, C. D. Remy, and R. Y. Siegwart. State estimation for legged robots - consistent fusion of leg kinematics and imu. In *Robotics: Science and Systems*, 2012. URL https://api.semanticscholar.org/CorpusID:1367000.

[41] J. Y. Luo, Y. Song, V. Klemm, F. Shi, D. Scaramuzza, and M. Hutter. Residual policy learning for perceptive quadruped control using differentiable simulation, 2024. URL https://arxiv.org/abs/2410.03076.

[42] N. Hansen, X. Wang, and H. Su. Temporal difference learning for model predictive control. *arXiv preprint arXiv:2203.04955*, 2022.

[43] N. Hansen, H. Su, and X. Wang. Td-mpc2: Scalable, robust world models for continuous control. *arXiv preprint arXiv:2310.16828*, 2023.