Time-Correlated Action Sampling in Model Predictive Path Integral

Minhyeong Lee and Dongjun Lee

***Abstract*— In this paper, we introduce time-correlated model predictive path integral (TC-MPPI), a novel method designed to mitigate action noise in sampling-based control algorithms. Unlike conventional smoothing techniques that rely on post-processing or additional state variables, TC-MPPI incorporates temporal correlation of actions into stochastic optimal control, effectively enforcing quadratic costs on action derivatives. This reformulation enables us to generate smooth action sequences without requiring further adjustments, using a time-correlated and conditional Gaussian sampling distribution. We demonstrate the effectiveness of our approach through simulations on various robotic platforms, including a pendulum, cart-pole, 2D bicopter, 3D quadcopter, and autonomous vehicle. Simulation videos are available at https://youtu.be/nWfJ2MAV2JI.**

I. Introduction

Model predictive control (MPC) is a powerful framework for managing robotic systems. It optimizes a future trajectory while adhering to state and action feasibility, dynamics, and safety constraints. Traditionally, gradient-based optimization techniques [1], [2] have been widely employed due to their flexibility and robustness. Despite their success, these methods often rely on differentiable objectives, dynamics, and constraints, limiting their applicability to non-differentiable problems. Moreover, their computational complexity often necessitates simplified or linearized dynamics, potentially compromising the control performance.

In contrast, sampling-based methods do not require the control problem to be differentiable and can effectively accommodate nonlinear dynamics. They also tend to explore the solution space more broadly, which helps to avoid poor local optima. Nonetheless, the sampling-based methods face considerable challenges, such as low sample efficiency and inherent noise, which can result in jittery actions. In this work, our primary focus is mitigating such noisy actions while ensuring optimality and sample efficiency.

The proposed method, *time-correlated model predictive path integral* (TC-MPPI), is inspired by stochastic optimal control with dynamic extension, where action derivatives are treated as extended state variables to ensure action smoothness. However, dynamic extension also increases the number of state variables, reducing sample efficiency. To address this, we reformulate the control problem by incorporating the temporal correlation of actions, as illustrated in Fig. 1. This reformulation adjusts the sampling mean and covariance by introducing quadratic costs on extended variables and a conditional distribution to maintain causality in time-correlated sampling. With this adjustment, the sampling distribution generates smooth action sequences without requiring post-processing or additional state variables. Notably, TC-MPPI integrates action smoothing within the stochastic optimal control rather than merely penalizing action derivatives.

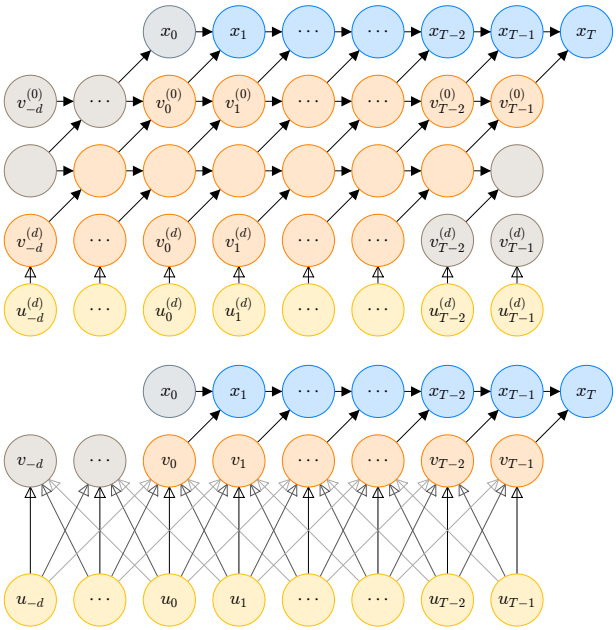

Fig. 1. Markov chains of the (top) extended and (bottom) time-correlated system. Blue nodes represent original system states, orange nodes denote original system actions and their derivatives, and yellow nodes indicate the mean of action nodes. Gray nodes represent the initial state, previous actions, and extended states that do not influence the future states.

Model predictive path integral (MPPI) [3], [4] is a leading approach for sampling-based real-time trajectory generation. Multiple MPPI variants have been developed to enhance performance. For example, linearized dynamics and control laws improve robustness and sample efficiency [5]–[7]. In safety-critical applications, conditional value-at-risk [8], [9] and control barrier functions [10] have been incorporated. To improve sampling distributions, techniques such as conditional variational autoencoders [11], normal log-normal distributions [12], adaptive importance sampling [13], and Stein variational gradient descent [14] have been employed. Yet, these advancements have not fully addressed the action noise. A notable exception is smooth-MPPI (S-MPPI) [15], which introduces a lifting strategy to separate the action from the sampling space, producing smooth action sequences.

This work is supported by the Technology Innovation Program (20024355 and 1415187329, Development of autonomous driving connectivity technology based on sensor-infrastructure cooperation) funded by the Ministry of Trade, Industry & Energy (MOTIE, Korea). Corresponding author: Dongjun Lee.

The authors are with the Department of Mechanical Engineering, Institute of Advanced Machines and Design (IAMD) and Institute of Engineering Research (IOER), Seoul National University, Seoul 08826, South Korea (e-mail: minhyeong@snu.ac.kr; djlee@snu.ac.kr).

While effective, the performance of S-MPPI can degrade with fewer samples or longer prediction horizons due to the increased complexity. In contrast, our work achieves action smoothing without relying on post-processing or additional state variables, ensuring smooth action sequences while preserving both optimality and sample efficiency.

The rest of the paper is organized as follows: Section II introduces key components of the stochastic optimal control problem and the MPPI algorithm; Section III details the proposed TC-MPPI algorithm; Section IV demonstrates the algorithm across various robotic tasks; and Section V concludes the paper.

II. Preliminary

This section outlines the theoretical background of model predictive path integral (MPPI) and offers intuitive insights into the MPPI algorithm to enhance understanding. To explore the details, let us consider a discrete-time system:

$$x_{t+1} = f_t(x_t, v_t) \tag{1}$$

where $x_t \in \mathcal{X}_t$ is the state, $v_t \in \mathcal{U}_t \subseteq \mathbb{R}^{n_u}$ is the action, and $f_t : \mathcal{X}_t \times \mathcal{U}_t \to \mathcal{X}_{t+1}$ is the transition model at time step $t \in \mathbb{Z}$. The system exhibits stochastic behavior, with the action being normally distributed as

$$V \sim \mathcal{Q}_{U,\Sigma} := \mathcal{N}(U, \Sigma) \tag{2}$$

where $V := v_{0:T-1}$ is the action sequence, $U := u_{0:T-1}$ is the mean input sequence, $u_t \in \mathcal{U}_t$ is the mean input, $\Sigma := \Sigma_{0:T-1}$ is the block diagonal covariance matrix, $\Sigma_t \in \mathbf{S}^{n_u}_{++}$ is the positive definite covariance matrix, and $T \in \mathbb{Z}_{++}$ is the horizon length. For simplicity, a sequence of vectors $\star_t$ is denoted as $\star_{a:b} := (\star_a, \star_{a+1}, \dots, \star_b)$, and a block diagonal matrix of matrices $\star_t$ is also denoted as $\star_{a:b} := \mathrm{diag}(\star_a, \star_{a+1}, \dots, \star_b)$. To maintain consistency with subsequent notations, we denote the distribution of V as $\mathcal{Q}_{U,\Sigma}$ and its probability density function as $q(V \mid U, \Sigma)$.

Now, an optimal control problem is formulated as

$$U^* = \arg\min_U \mathbb{E}_{V \sim \mathcal{Q}_{U,\Sigma}} \left[c_T(x_T) + \sum_{t=0}^{T-1} \ell_t(x_t, u_t) \right] \tag{3}$$

where $\ell_t : \mathcal{X}_t \times \mathcal{U}_t \to \mathbb{R}$ is the running cost function, and $c_T : \mathcal{X}_T \to \mathbb{R}$ is the terminal state-dependent cost function. Assuming that the running cost can be decoupled into a state-dependent cost $c_t : \mathcal{X}_t \to \mathbb{R}$ and a quadratic action-dependent cost, the control objective can be rewritten as

$$c_T(x_T) + \sum_{t=0}^{T-1} \ell_t(x_t, u_t) = S(V) + \frac{\lambda}{2} \|U - U_{\mathrm{ref}}\|^2_{\Sigma^{-1}} \tag{4}$$

where $U_{\mathrm{ref}} := u_{\mathrm{ref},0:T-1}$ is the reference action sequence, $u_{\mathrm{ref},t} \in \mathcal{U}_t$ is the reference action, $\lambda \in \mathbb{R}_{++}$ is the temperature, and $S(V) := c_T(x_T) + \sum_{t=0}^{T-1} c_t(x_t)$ is the state-dependent cost associated with the state trajectory $X := x_{0:T}$ following the transition model and the action sequence V.

As noted in [4], the optimal distribution $\mathcal{Q}^*$ that provides a lower bound for the control objective (4) is observed as

$$q^*(V) \propto \exp\left(-\frac{1}{\lambda} S(V) \right) p(V) \tag{5}$$

where $p(V) := q(V \mid U_{\mathrm{ref}}, \Sigma)$ is the base probability density function. It implies that the optimal control problem (3) can be solved by minimizing the Kullback-Leibler (KL) divergence between the controlled distribution $\mathcal{Q}_{U,\Sigma}$ and the optimal distribution $\mathcal{Q}^*$ so that

$$U^* = \arg\min_U D_{\mathrm{KL}} \left(\mathcal{Q}^* \,\|\, \mathcal{Q}_{U,\Sigma} \right) \tag{6}$$

which leads to a sampling-based optimization:

$$U^* = \mathbb{E}_{V \sim \mathcal{Q}^*}[V] = \mathbb{E}_{V \sim \hat{\mathcal{Q}}}[w(V)V] \tag{7}$$

where $\hat{\mathcal{Q}}$ is the sampling distribution, $\hat{q}(V) := q(V \mid \hat{U}, \Sigma)$ is the sampling probability density function, and $w(V) := \frac{q^*(V)}{\hat{q}(V)}$ is the importance sampling weight. Importance sampling is employed because directly sampling V from the optimal distribution $\mathcal{Q}^*$ is typically intractable.

The optimal distribution $\mathcal{Q}^*$ derived from MPPI can also be characterized using Bayesian inference:

$$q^*(V) := p(V \mid o_\tau) = \frac{p(o_\tau \mid V)\, p(V)}{p(o_\tau)} \tag{8}$$

where $o_\tau \in \{0, 1\}$ is the optimality indicator of the trajectory $\tau := (X, V)$. In the context of the optimal distribution (5), the base probability serves as the prior probability $p(V)$, while the negative exponential of the state-dependent cost $S(V)$ represents the likelihood $p(o_\tau \mid V)$.

III. Time-Correlated MPPI

A notable challenge in sampling-based optimal control is managing noisy actions. While filtering can reduce the noise, weak filters might be ineffective, and strong filters can compromise performance or optimality. Adjusting the sampling distribution is another option, but incorrect choices can lead to optimization failures. To address this, we incorporate temporal correlation of actions into stochastic optimal control, ensuring smooth action sequences and sample efficiency.

We first formulate an optimal control with dynamic extension to obtain smooth action sequences. In this context, an extended system is defined as

$$x_{t+1} = f_t(x_t, v_t^{(0)}) \quad \text{and} \quad v_{t+1}^{(i)} = v_t^{(i)} + h_t v_t^{(i+1)} \tag{9}$$

where $v_t^{(i)} \in \mathbb{R}^{n_u}$ is the i-th derivative of the action, and h_t is the step size. For a dynamic extension depth $d \in \mathbb{Z}_+$, the new action $v_t^{(d)}$ follows a normal distribution:

$$v^{(d)}_{-d:T-1} \sim \mathcal{N}(u^{(d)}_{-d:T-1}, \mathrm{inv}(R^{(d)}_{-d:T-1})) \tag{10}$$

where $u_t^{(d)} \in \mathbb{R}^{n_u}$ is the mean input, and $R_t^{(d)} \in \mathbf{S}^{n_u}_{++}$ is the inverse covariance matrix. It is important to note that the prediction horizon begins from $-d$ rather than zero, since $v_{-d}^{(d)}$ influences the initial action $v_0^{(0)}$, as shown in Fig. 1.

While the extended system enables smooth action sequences, it increases the number of state variables, often

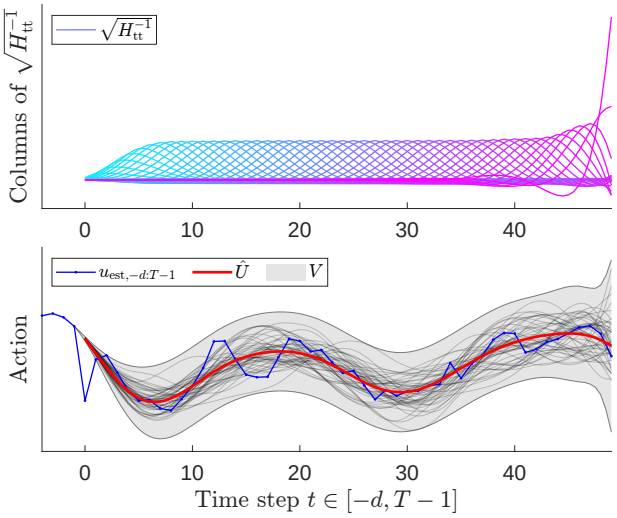

Fig. 2. Example of the sampling distribution. Top: Columns of the covariance square root matrix, forming the action sequence basis. Bottom: Mean (red) and sampled (gray) action sequences. An artificial noise is added to the estimated optimal sequence (blue) to show the smoothing effect of the gradient operator.

requiring more samples or iterations in the sampling-based control. Therefore, we revert the sampling space to the original action space by designing the cost of extended states $v_t^{(i)}$ to be quadratic. From the fact that action derivatives $v_t^{(i)}$ can be represented as finite differences of actions v_t, the cost of extended states can be expressed in a quadratic form of $v_{-d:T-1}$. Since the states x_t also can be obtained using actions v_t, an optimal distribution of $v_{-d:T-1}^{(d)}$ for the extended system (9), similar to (5), can be written as a function of the original action sequence $v_{-d:T-1}$. In the context of Bayesian inference (8), this reformulation transfers the cost associated with extended variables to the prior distribution, thereby improving sample efficiency.

However, the transcription causes a causality issue because the action sequence includes previous actions $v_{-d:-1}$, which are beyond our control at the current time step $t = 0$. To address this, we introduce a conditional distribution given the previous action sequence $u_{-d:-1}$. Then, we obtain a conditional optimal probability density:

$$q_{\rm tc}^*(V) \propto \exp\left(-\frac{1}{\lambda}S(V)\right)\exp\left(-\frac{1}{2}\|V-\bar{U}\|_{H_{\rm tt}}^2\right) \quad (11)$$

where $\bar{U} \in \mathbb{R}^{n_u T}$ and $H_{\rm tt} \in \mathbb{R}^{n_u T \times n_u T}$ are the mean and inverse covariance of the conditional prior distribution, and $\mathcal{Q}_{\rm tc}^*$ is the distribution of the optimal probability $q_{\rm tc}^*(V)$. The optimal U, which aligns $\mathcal{N}(U, H_{\rm tt}^{-1})$ with the optimal distribution $\mathcal{Q}_{\rm tc}^*$, can be obtained by

$$U^* = \mathbb{E}_{V\sim\mathcal{Q}_{\rm tc}^*}[V] = \mathbb{E}_{V\sim\hat{\mathcal{Q}}_{\rm tc}}[w_{\rm tc}(V)V] \quad (12)$$

where $\hat{\mathcal{Q}}_{\rm tc} := \mathcal{N}(\hat{U}, H_{\rm tt}^{-1})$ is the sampling distribution, $\hat{q}_{\rm tc}(V)$ is the corresponding probability density function, and

$$w_{\rm tc}(V) := \frac{q_{\rm tc}^*(V)}{\hat{q}_{\rm tc}(V)} \propto \exp\left(-\frac{1}{\lambda}S(V) + (\bar{U}-\hat{U})^{\rm T} H_{\rm tt} V\right)$$

TABLE I

SUCCESS RATE AND TERMINAL ERROR OF SWING-UP TASKS

Method	Pendulum swing-up		Cart-pole swing-up	
	Success	Error (deg)	Success	Error (deg)
MPPI	50/50	0.46 ± 0.32	50/50	0.36 ± 0.28
MPPI w/ SGF	50/50	0.33 ± 0.25	50/50	0.27 ± 0.21
S-MPPI	30/50	5.62 ± 3.96	50/50	0.22 ± 0.18
MPPI w/ DE	20/50	6.33 ± 4.47	0/50	73.88 ± 52.62
TC-MPPI	50/50	0.31 ± 0.22	50/50	0.18 ± 0.13

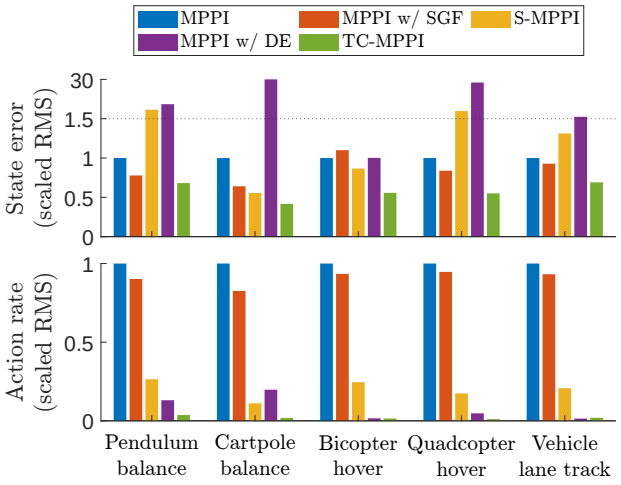

Fig. 3. State errors and action change rates for stabilizing tasks. High action rates indicate high action noise. To account for varying scales across tasks, the values are normalized so that MPPI values are set to 1. Scaled errors exceeding 30 are not shown.

is the importance sampling weight. Since $V \sim \mathcal{N}(\hat{U}, H_{\rm tt}^{-1})$ is equivalent to $V = \hat{U} + H_{\rm tt}^{-1/2}\mathcal{E}$ with $\mathcal{E} \sim \mathcal{N}(0, I)$, the columns of the covariance square root matrix serves as a basis for the random action sequence, capturing the temporal correlation. Fig. 2 illustrates an example of the time-correlated and conditional Gaussian sampling distribution.

The proposed method is called time-correlated model predictive path integral (TC-MPPI). It is important to note that advanced techniques commonly used in MPPI-like algorithms (e.g., incorporating exploration samples or decoupling the action cost from the temperature [4]), though not addressed in this paper, can also be seamlessly integrated into our framework.

IV. RESULT

We evaluate the proposed method across various control tasks to demonstrate its ability to generate smooth actions while maintaining high performance. TC-MPPI is compared with MPPI variants: the original MPPI [3], MPPI with a Savitzky-Golay filter (SGF) [4], and smooth-MPPI (S-MPPI) [15]. MPPI with dynamic extension (DE) is also tested to validate the influence of extended dynamics (9). All algorithms are implemented in MATLAB and tested on a Windows 11 machine with AMD Ryzen 5 3600X CPU and 16 GB RAM. Refer to [16] for the details of control and system parameters. Simulation videos are available at https://youtu.be/nWfJ2MAV2JI.

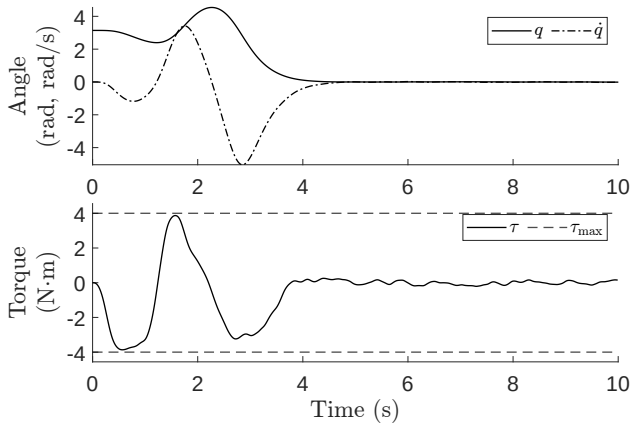

Fig. 4. State and action for the pendulum swing-up task using TC-MPPI. The resulting actions are smooth and effectively accomplish the task.

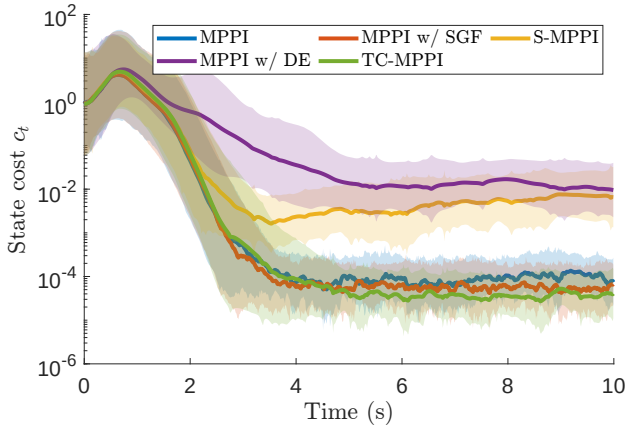

Fig. 5. Average state cost $c_t(x_t)$ for the pendulum swing-up task. TC-MPPI achieves action smoothing without compromising performance.

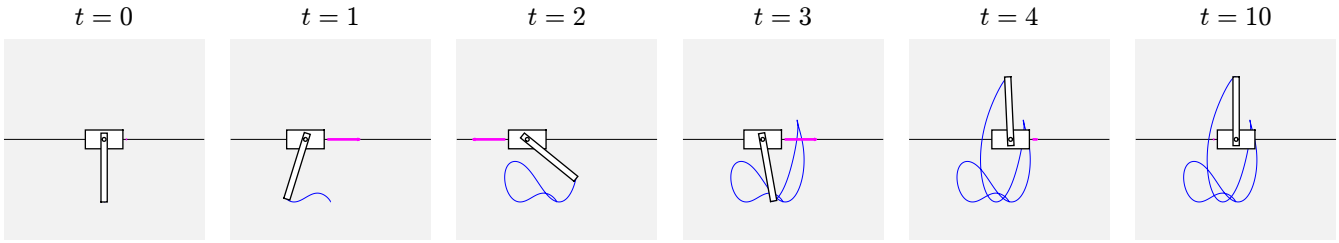

Fig. 6. Snapshots of the cart-pole swing-up task using TC-MPPI. The cart force (magenta) and the pole tip trajectory (blue) are shown. As the cart force is constrained, it adjusts the force direction to swing-up the pole, ultimately balancing at the origin.

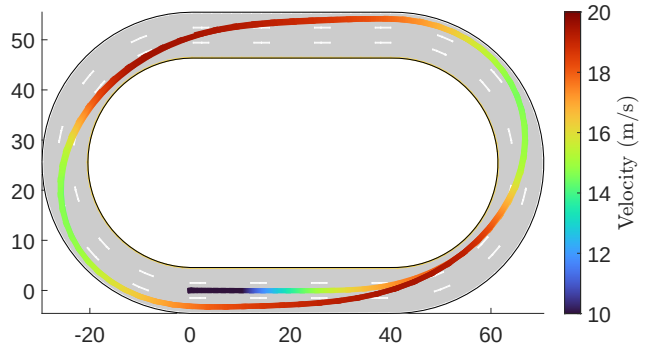

Fig. 7. Trajectory of autonomous vehicle circuit tracking using TC-MPPI. Vehicle speed is shown with a color map. The vehicle starts at the origin with zero velocity, accelerates to the desired velocity, and slows down at corners to prevent slip and collision.

Table I summarizes success rates and errors for the pendulum and cart-pole swing-up tasks. Fig. 3 shows RMS state errors and action change rates for stabilizing tasks. With small sample sizes, MPPI with SGF often outperforms MPPI without filters, as the filter can mitigate action variance. However, SGF can also degrade performance by violating the solution optimality. S-MPPI effectively reduces action noise and improves performance, but it may result in higher errors if the sample size is insufficient relative to problem complexity. MPPI with DE produces smooth actions yet sacrifices control performance, often leading to failures. In contrast, TC-MPPI remarkably reduces action noise and achieves lower errors, demonstrating its ability to smooth actions without compromising performance. While performance is not our primary objective, TC-MPPI yields better results by improving action variance and sample efficiency. In practice, action smoothness would also be beneficial with unmodeled high-frequency dynamics, though it is not considered in simulations.

V. Conclusion

In conclusion, we propose a sampling-based control algorithm that generates smooth action sequences. Our approach incorporates the temporal correlation of actions in stochastic optimal control, which is equivalent to applying a quadratic cost on action derivatives. The sampling distribution is then derived as a conditional Gaussian distribution, enabling smooth action generation without requiring post-processing or additional state variables. We validate the proposed method through simulations on various platforms, including a pendulum, cart-pole, 2D bicopter, 3D quadcopter, and autonomous vehicle. The results demonstrate a significant reduction in action noise while improving overall performance. Future research directions include conducting hardware experiments, analyzing the effect of correlation parameters, quantifying robustness against unexpected disturbances and communication delays, integrating fast simulators [17], and extending our time correlation approach to other sampling-based methods [18], [19].

REFERENCES

[1] O. Von Stryk and R. Bulirsch, "Direct and indirect methods for trajectory optimization," *Annals of Operations Research*, vol. 37, pp. 357–373, 1992.

[2] A. V. Rao, "Trajectory optimization: A survey," *Optimization and Optimal Control in Automotive Systems*, pp. 3–21, 2014.

[3] G. Williams, N. Wagener, B. Goldfain, P. Drews, J. M. Rehg, B. Boots, and E. A. Theodorou, "Information theoretic MPC for model-based reinforcement learning," in *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1714–1721, IEEE, 2017.

[4] G. Williams, P. Drews, B. Goldfain, J. M. Rehg, and E. A. Theodorou, "Information-theoretic model predictive control: Theory and applications to autonomous driving," *IEEE Transactions on Robotics*, vol. 34, no. 6, pp. 1603–1622, 2018.

[5] G. Williams, B. Goldfain, P. Drews, K. Saigol, J. M. Rehg, and E. A. Theodorou, "Robust sampling based model predictive control with sparse objective information," in *Robotics: Science and Systems*, vol. 14, 2018.

[6] M. S. Gandhi, B. Vlahov, J. Gibson, G. Williams, and E. A. Theodorou, "Robust model predictive path integral control: Analysis and performance guarantees," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1423–1430, 2021.

[7] J. Yin, Z. Zhang, E. Theodorou, and P. Tsiotras, "Trajectory distribution control for model predictive path integral control using covariance steering," in *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1478–1484, IEEE, 2022.

[8] Z. Wang, O. So, K. Lee, and E. A. Theodorou, "Adaptive risk sensitive model predictive control with stochastic search," in *Proceedings of Learning for Dynamics and Control*, pp. 510–522, PMLR, 2021.

[9] J. Yin, Z. Zhang, and P. Tsiotras, "Risk-aware model predictive path integral control using conditional value-at-risk," in *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 7937–7943, IEEE, 2023.

[10] J. Yin, C. Dawson, C. Fan, and P. Tsiotras, "Shield model predictive path integral: A computationally efficient robust MPC method using control barrier functions," *IEEE Robotics and Automation Letters*, vol. 8, no. 11, pp. 7106–7113, 2023.

[11] R. Kusumoto, L. Palmieri, M. Spies, A. Csiszar, and K. O. Arras, "Informed information theoretic model predictive control," in *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 2047–2053, IEEE, 2019.

[12] I. S. Mohamed, K. Yin, and L. Liu, "Autonomous navigation of AGVs in unknown cluttered environments: log-MPPI control strategy," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10240–10247, 2022.

[13] D. M. Asmar, R. Senanayake, S. Manuel, and M. J. Kochenderfer, "Model predictive optimized path integral strategies," in *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 3182–3188, IEEE, 2023.

[14] K. Honda, N. Akai, K. Suzuki, M. Aoki, H. Hosogaya, H. Okuda, and T. Suzuki, "Stein variational guided model predictive path integral control: Proposal and experiments with fast maneuvering vehicles," in *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 7020–7026, IEEE, 2024.

[15] T. Kim, G. Park, K. Kwak, J. Bae, and W. Lee, "Smooth model predictive path integral control without smoothing," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10406–10413, 2022.

[16] M. Lee and D. Lee, "Time-correlated model predictive path integral: Smooth action generation for sampling-based control," in *Proceedings of IEEE International Conference on Robotics and Automation*, IEEE, 2025, to appear.

[17] J. Lee, M. Lee, and D. Lee, "Large-dimensional multibody dynamics simulation using contact nodalization and diagonalization," *IEEE Transactions on Robotics*, vol. 39, no. 2, pp. 1419–1438, 2022.

[18] R. Y. Rubinstein and D. P. Kroese, *The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation, and Machine Learning*, vol. 133. Springer, 2004.

[19] F. Stulp and O. Sigaud, "Path integral policy improvement with covariance matrix adaptation," in *Proceedings of International Conference on Machine Learning*, pp. 281–288, 2012.