Towards Planning with Diffusion for Complex Environments

Tobias Klausen
1009763221

Matin Moezzi
1008701479

Xuanchi Ren
1009173403

Abstract: Model-based reinforcement learning (RL) techniques employ learning just to estimate an approximate dynamics model, leaving the remaining decision-making to traditional trajectory optimizer or policy optimization methods. While theoretically straightforward, this independent two-stage planning has a number of empirical flaws that indicate trained models might not be the best choice for the learned policy or conventional trajectory optimization. Taking advantage of the diffusion probabilistic model, some previous works tried to incorporate a trajectory optimization pipeline into model learning such that sampling from the model and using it for planning become essentially equivalent. In this work, we propose a Latent-based Planning Diffuser, a new framework for planning in high-dimensional environments based on previous works. Also, we modify the diffusion model training by introducing task-specific constraints. We test our approach on the CARLA simulator for autonomous driving and the Adroit domains.

Keywords: Model-based RL, Diffusion Probabilistic Models, Planning for RL

1 Introduction

For model-based reinforcement learning (RL), we first learn a predictive model and use this learned model to evaluate potential sequences of actions and select the best one. Though this process is conceptually simple, it suffers from severe problems. Trajectories generated by this pipeline resemble adversarial examples rather than optimal solutions. Thus, instead of following the above trajectory optimization procedure, model-based RL approaches usually inherit more from model-free algorithms like value functions and policy gradients. To address this issue, Janner et al. [1] proposes to treat trajectory optimization as a generative modeling process with diffusion models.

Though the prior work Diffuser [1] demonstrates a great potential for using diffusion models for planning, it is mainly applied to simple scenarios such as a Maze with a local receptive field and states represented by mainly coordinates. However, when it comes to scenarios with multi-agents and complex environments (such as autonomous driving), such a local receptive will fail to capture the history of trajectories of agents and the interaction between the ego-agent and the environment, as shown in Figure 1. Moreover, the states of such a complicated environment can not be simply represented by coordinates, requiring further consideration of the design of the diffusion model.

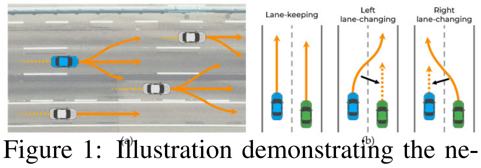

Figure 1: Illustration demonstrating the necessity of global awareness of the whole environment and other agents.

In this project, we aim to plan with a learned diffusion model in challenging dynamic environments containing obstacles and multi-agents, such as a high-DoF robotic hand (Adroit) or autonomous driving simulators (CARLA), to fully leverage the potential of Diffuser [1]. To realize this, we will first replicate the result of Janner et al. [1]. Based on it, we will investigate the possibility of modifying the original diffusion model to a latent diffusion model. Moreover, we would like to perform an empirical analysis on combining the transformer with the current architecture to empower the Diffuser [1] with the global receptive field.

In addition, we modify the reverse diffusion process by adding task-specific constraints to enhance trajectory optimization for continuous control problems.

Main contribution. We summarize our project as follows:

- We will replicate the results of the Diffuser [1] and main baselines based on the official implementation.
- We will extend Diffuser [1] from simple s
- To handle the complex scenarios, we propose to utilize a latent diffusion and a non-local receptive field with the power of Transformers.
- We attach task-specific constraints to the diffusion model training to improve trajectory optimization accuracy for high-dimensional continuous control environments.

2 Related Work

Model-based RL and Planning. A lot of research has been done on model-based reinforcement learning. Their differences stem from the choice of model parameterization, which is connected to various applications of the model for policy learning. The most impressive robotic learning applications yet, it turns out, have been accomplished utilizing the most basic model parameterization, notably, linear models, where the model either operates directly over the raw state or over a feature representation of the state. Such models enable very effective policy optimization using methods from optimal control. However, unless a separate feature learning phase is implemented, they only have limited expressiveness and do not scale well to complex nonlinear dynamics or high-dimensional state spaces.
Nonparametric models like Gaussian Processes (GPs) are an alternative. As long as there is a sufficient amount of data, such models may effectively preserve uncertainty over the predictions and have limitless representation power. However, due to the curse of dimensionality, they are only truly useful in environments with few dimensions.

Diffusion Models. With the success of Diffusion Models in density estimation and sample quality, it is also introduced into the domains of images, video, and 3D with an underlying neural backbone as a UNet. The generative process of Diffusion Models is formulated as an iterative denoising procedure [2] with lots of variants. While Diffusion Models in these domains achieve amazing performance and have been well-investigated, there are few works developing them for decision-making or reinforcement learning tasks. Janner et al. [1] make the first step with a local receptive field on some simple scenarios. In this project, we plan to extend it to complex scenarios, which are challenging and requires a novel architecture.

3 Proposed Method

In this section, we first briefly introduce Diffuser [1], the diffusion model for trajectory optimization, and then talk about our considered improvement to extend it to dynamic and complex scenes.

Given a reward function $r(s_t, a_t)$, our goal of planning is to find the optimal sequence of actions $a_{0:T}$ that satisfies:

$$a^*_{0:T} = \operatorname*{argmin}_{a_{0:T}} \sum_{t=0}^{T} r(s_t, a_t) = \operatorname*{argmin}_{a_{0:T}} \mathcal{J}(s_0, a_{0:T}), \tag{1}$$

where $\mathcal{J}$ is an objective that characterizes the value of trajectories. Moreover, we denote the trajectory τ as:

$$\tau = \begin{bmatrix} s_0 & s_1 & \dots & s_T \\ a_0 & a_1 & \dots & a_T \end{bmatrix} \tag{2}$$

As demonstrated in Janner et al. [1], this planning process can be approximately subsumed to generative modeling framework, and thus it can be decomposed into two modules: (i) a diffusion model $p_\theta(\tau)$ modeling the physically realistic of the trajectories; (ii) a reward constraint $h(\tau)$ characterized by a model $\mathcal{J}_\phi$. The $p_\theta(\tau)$ is a temporal UNet composed of $1D$ convolutions and trained with the denoising diffusion process. It is capable of iteratively generating realistic τ from random sampled noise. For the $\mathcal{J}_\phi$, it is trained separately to predict the cumulative rewards of the trajectories during the diffusion process as a signal indicator.

To apply Diffuser [1] to dynamic and complex scenes, there are two main difficulties: (i) it only considers tasks with simple states that $1D$ arrays can represent, while the states of complex scenarios can not be trivially treated as $1D$ arrays; (ii) it mainly focuses on the local receptive field and thus might fail to deal with the global context.

To address the first difficulty, we plan to adopt a latent diffusion model, which uses an additional auto-encoder to encode the complex states into latent space and then learn a diffusion model to model the trajectories in the latent space. For example, for a state s represented by a 2D image, we can use a pre-trained encoder E to encode it as $e = E(s)$ to the latent space. And the trajectory τ can be written as $(e_0, a_0, e_1, a_1, ..., e_T, a_T)$. Additionally, for the environments with multi-agents, we consider using Graph Convolutional Networks (GCN) for the design of auto-encoder architecture.

Furthermore, to enable the model to have knowledge of the global context, we would like to perform an empirical study of injecting transformer layers into the UNet architecture of the diffusion model. For example, we can add self-attention blocks to the middle of UNet.

4 Proposed Evaluation

The main purposes of our experimental evaluation are (1) to examine the performance of the proposed method in high-dimensional state space environments (e.g., image-based state space) and high-DoF continuous control problems like dexterous hand manipulation tasks, (2) to compare the accuracy of the proposed method and original diffusion planning algorithm in high-dimensional action and state spaces environments (3) to examine whether the proposed diffusion-based method can outperform the model-based and model-free baselines.

Environments. We first consider replicating the results on the **Hopper** environment on the D4RL [3] locomotion benchmark, which requires only $p_\theta(\tau)$. Then we plan to replicate the results on the **Maze2D** environment with the full Diffuser model. Then, we test our method in two environments to examine the latent-based diffusion method performance in high-dimensional space. First is the **CARLA** simulator, in which high-dimensional images represent the state space, and second is the **Adroit** domain, a 24-DoF ShadowHand, to perform a dexterous manipulation task as a high-dimensional continuous action space environment.

Baselines. We compare our method with two prior model-free offline RL algorithms: **BCQ** and **CQL**. We also consider various model-based reinforcement learning algorithms, such as Trajectory Transformer (**TT**), **MBOP** and **MOReL**. Moreover, we use Decision Transformer (**DT**), a return-conditioning approach.

References

[1] M. Janner, Y. Du, J. B. Tenenbaum, and S. Levine. Planning with diffusion for flexible behavior synthesis. In *ICML*, 2022.

[2] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

[3] J. Fu, A. Kumar, O. Nachum, G. Tucker, and S. Levine. D4RL: datasets for deep data-driven reinforcement learning. *CoRR*, abs/2004.07219, 2020.