MoManTu based Robotics Control Policy

Shunkang Jia
jiashk@shanghaitech.edu.cn

Tianhang Liu
liuth@shanghaitech.edu.cn

Zichen Jin
jinzch12023@shanghaitech.edu.cn

Abstract

In general, most of the tasks performed by robots are related to picking up and placing objects, usually combined with trajectory planning and manipulation. For the objects to be picked up, they are given a generous collision detection volume until they finally approaches the target position. Occasionally, optimal trajectory may be too close to the obstacles to make the kinematics solver regard it safe and plausible. This situation gives us the intuition that the solver is much too conservative under certain collision avoidance constraints. Thus, our group (intended to) implements a control algorithm to eliminate the conservatism. Conventional method like control barrier function(CBF) [3] takes the network-predicted trajectory plan as input, and imposes collision constraints using an optimization layer. The effectiveness is well proved while conservatism still exists. Our final target is to make improvement on learning-based CBFs. The initialization modality has been considered the major factor which influence the final output of an optimization problem. Delightfully, diffusion model[1] gains the ability to generate motion trajectory and increase the modality. Based on the properties of diffusion model and control barrier function, we decide to combine both of the architectures to erase the conservatism of trajectory planning.

1. Introduction

Trajectory planning with collision avoidance has been a long-term excavated and difficult mission depending on the target and environment complexity. Object picking and placement, for instance, only consider the grabbed objects as rigid and static objects. The very few control algorithms that directly intend to interact with deformable surface objects in motion planning only reduce the threat of obstacles by other methods such as image or semantic point cloud recognition [11]. For some other cases, small object can be placed to the target position by underactuated movements like pushing and ticking, not simply grabbing. Diversity of actions to complete the placement mission is also called modality. Diffusion model is currently the state-of-the-art method ensuring a high level of modality to motion generation. On the other hand, increased modality also causes the instability of the system which can be decreased using control barrier function layer.

So we propose an external damping(or constraints relaxation) system, attempt to increase the interaction, or reduce the safe distances between given solutions and the obstacles. For a further improvement, we also want to estimate the contact force that may be exerted onto the obstacle, and calculate their contact trajectory on the obstacle surface, this trajectory includes the pose, velocity and acceleration of the joint end in contact with the obstacle surface. Our system also requires a ground truth handwriting trajectory of a large range used as an optimization target for inverse kinematics problem. This training process is based on conditional diffusion model.

For a traditional control barrier function in [3],

$$\begin{aligned} u(x) = \underset{(u,\delta)\in\mathbb{R}^{m+1}}{\operatorname{argmin}} \quad & \frac{1}{2}u^T H(x)u + p\delta^2 \quad \text{(CLF-CBF QP)} \\ \text{s.t.} \quad & L_f V(x) + L_g V(x)u \leq -\gamma(V(x)) + \delta \\ & L_f h(x) + L_g h(x)u \geq -\alpha(h(x)) \end{aligned} \tag{1}$$

where here $H(x)$ is any positive definite matrix (pointwise in x), and δ is a relaxation variable that ensures solvability of the QP as penalized by $p > 0$ (i.e., to ensure the QP has a solution one must relax the condition on stability to guarantee safety). In [2] it was established that this controller is Lipschitz continuous.

As for the state dynamics constraint which are denoted in the form of Lie derivatives, we can illustrate it in the original form of Newton-Euler equation as follows:

$$M(q)\ddot{q} + B(\dot{q}) + C(q,\dot{q})\dot{q} = \tau_g(q) + \tau_u(q)$$

Where $M(q)$ stands for the inertia matrix, $B(\dot{q})$ is the damping torque related to the system, C contains the cori-

olis forces and centripetal forces, on the right side are the gravity torque and robot's motor torque.

Since we're modeling a dissipation system, the energy loss comes from the internal and external damping and contact forces, which leads to the dynamics constraints changes, because action can be regarded as the descent gradient of system energy function. Conventional method use Bayesian filter or other kinds of observation and prediction fusion methods to estimate and track the robot pose. While, as we will prove later in chapter 3, conditioning score-based diffusion model has the ability to fuse different sources of observations.

2. Related Work

The work from [13] encodes the environment and embeds it with the robot's joint states and target poses as inputs to train a reinforcement neural network, the output is action sequece in the robot's configuration space. After the output layer, they use an optimization layer which enforces the output to follow the initial trajectory in the configuration space. Basically, this is an RL-based inverse kinematics solver leveraging multiple rewards to shrink the configuration searching space and implicitly impose the dynamics constraints. To accelerate both the training and predicting processes, they also sample some initial IK poses for the first frame of the target trajectory, as a warm up for the training. Another group proposed an architecture to simultaneously solve the upper-limb and lower-limb control policy. An RL algorithm for one stage manipulation and locomotion is provided and adopted to a four-legged robot dog with an robotics arm on its back. The control of hands and legs or wheels is so tied together that they form low-dimension synergies. For instance, the robot bends or stretches its legs with the movement of the arm to extend the reach of the end-effector. This system perfectly solve the conflicting objectives and local minima problems. When the arm tilts to the right, the robot needs to change the walking gait to account for the weight balance. They use both manipulation and locomotion rewards including command following, lower energy dissipation and keeping alive, which encourages command following while penalizes positive mechanical energy consumption to enable smooth motion and guarantee the robot keeps in balance.

[5] also leveraged multi-agents system which make cooperation between two agents' arms. However, such a seemingly common skill introduces a lot of challenges for robots to achieve: The robots need to operate such dynamic actions at high-speed, collaborate precisely, and interact with diverse objects. This paper proposes a system with two multi-finger hands attached to robot arms to solve this problem. They train the system using Multi-Agent Reinforcement Learning in simulation and perform Sim2Real transfer to deploy on the real robots. In conclusion, they are training robots to predict environment changes like flying objects which give us the intuition on environmental energy loss constraints for control barrier functions. Besides environment or external variation prediction, perception for the above changes is equivalently significant. According to [12], detecting objects, and estimating their 3D position, orientation and size is an important requirement in virtual and augmented reality, robotics, and 3D scene understanding. These applications require operation in new environments that may contain previously unseen object instances. In [7], they proposed a novel differentiable framework for the uncertain pose estimation during contact, so that it can be solved in an efficient and accurate manner with gradient-based solver. To achieve this, they introduce a new geometric definition that is highly adaptable and capable of providing differentiable contact features. They approach the problem from a bi-level perspective and utilize the gradient of these contact features along with differentiable optimization to efficiently solve for the uncertain pose.

A novel structure dealing with safety constraints was also proposed by [8]. This paper provides deterministic methods for motion planning guarantee safety amidst uncertainty in obstacle locations by trying to restrict the robot from operating in any possible location that an obstacle could be in. We may optimize the opposite loss function to maximize the probability for end joint meeting an obstacle, at the meanwhile, still keep the other parts of the robot safe. The objective-function-based method actually introduce soft constraints to the robotics system, while control barrier functions introduce hard constraints. According to [10], grasping in cluttered environments is a fundamental but challenging robotic skill. It requires both reasoning about unseen object parts and potential collisions with the manipulator. Most existing data-driven approaches avoid this problem by limiting themselves to top-down planar grasps which is insufficient for many real-world scenarios and greatly limits possible grasps.

As for a complete pipeline which includes all of the learning, optimizing, pose updating processes, the paper[6] coming from 2023, Nature, proposed a engineering feasible architecture for reinforcement learning with tremendous reduced sim-to-real gap. The drone system consists of two key modules: a perception system that translates visual and inertial information into a low-dimensional state observation and a control policy that maps this state observation to control commands. Control commands specify desired collective thrust and body rates, the same control modality that the human pilots use, The perception system consists of a VIO module that computes a metric estimate of the drone state from camera images and high-frequency measurements obtained by an inertial measurement unit (IMU). The VIO estimate is coupled with a neural network that detects the corners of racing gates in the image stream. The

corner detections are mapped to a 3D pose and fused with the VIO estimate using a Kalman filter, We use model-free on-policy deep RL to train the control policy in simulation. During training, the policy maximizes a reward that combines progress towards the centre of the next racing gate with a perception objective to keep the next gate in the field of view of the camera. To transfer the racing policy from simulation to the physical world, we augment the simulation with data-driven residual models of the vehicle's perception and dynamics. This system is similar to a visual servoing or any other close-loop structure which adjust corresponding actions based on the target and current estimated states. This gives us the motivation to improve servoing system or state filter system by subsituting with conditional diffusion model. In next chapter, we will prove the equivalence of this substitution.

3. Method

In this section, we first give the fundamental frameworks we used for robotics development which includes ROS message delivery and state machine structure. Based on the low level integration works, we then connect the built blocks to the diffusion model layer and control barrier function layer. Detail proof about

3.1. FlexBe state machine

For the basic code architecture, we follow the demand of FlexBe framework which leverages the ROS subscriber and publisher mode. FlexBe uses behaviour and state to assemble and manage robot skills and actions. The robot system first initialize in a starter state like stably standing or looking-down. Then every state provides feasible actions triggered by the message topic subscribed by itself. After actions execution, robot configuration updates and the contents in corresponding topic messages related to cameras and motor sensors also change. When FlexBe state receives the changing messages, state transformation occurs and the new current state provides another set of actions and skills. All the states are integrated into a behaviour model with transformation linkage. The behaviour maintains the topology between various states.

Thus, our neural network can finish the high-level planning mission, and leave the detail control problems to the FlexBe state machine.

3.2. Langevinized Ensemble Kalman Filter (not implemented)

In this sub-section, we try to look for the connection between diffusion and denoising Gaussian filtering system from another aspect. The ensemble Kalman filter under Langevan motion actually reveals the same intrinsic as a score-based diffusion model. First of all, let's look at the following simple Kalman filter process, x is the random variable of the system state, y represents the system observation measurement,

$$x_t = g(x_{t-1}) + u_t, \quad u_t \sim N(0, U_t), \tag{2}$$

$$y_t = H_t x_t + \eta_t, \quad \eta_t \sim N(0, \Gamma_t), \tag{3}$$

where the function g is the state transition equation from time $t-1$ to time t, and there is Gaussian noise in both the observation process and the evolution process of the system. In the past, the filtering system was based on the prior probabilities of x and y, and the updated state and the observed measurements were fused, during which the two distributions would present an adversarial situation, and we would judge the credibility of each distribution, then adopt an interpolation operation between the two distributions, so as to obtain the maximum posterior probability. However, in many cases, we are not able to retrieve the distribution of the accurate prior probability of a system state, or this prior probability needs to be continuously adapted in the process of system iteration. Therefore, the ensemble Kalman filter achieves this adaptive effect by forecasting and analyzing the real-world data in batches to gradually approximate such a distribution. There are many ways to approximate it, one of which is based on Langevinized dynamics. Through the gradient ascend of the hypothetical distribution at the previous time, the mean of the distribution at the next moment is moving closer to the mode of the real system. Looking at the process of forecasting, we would find that the iteration process of it is basically the same as that of score base diffusion model.

$$\begin{aligned}\nabla \log p(\boldsymbol{x}_t|y) &= \nabla \log \left(\frac{p(\boldsymbol{x}_t)p(y|\boldsymbol{x}_t)}{p(y)} \right) \\ &= \nabla \log p(\boldsymbol{x}_t) + \nabla \log p(y|\boldsymbol{x}_t) - \nabla \log p(y) \\ &= \underbrace{\nabla \log p(\boldsymbol{x}_t)}_{\text{unconditional score}} + \underbrace{\nabla \log p(y|\boldsymbol{x}_t)}_{\text{adversarial gradient}}\end{aligned} \tag{4}$$

The filtering process described in the popular understanding is actually a process of fusion of multiple signal sources, and the adversarial term is actually the same as the objective function of the Kalman filter, not only as an addition to fitting the predicted prior distribution, but also trying to maximize the likelihood estimation of the observations. The gradient direction of the score function is actually the direction in which the posterior probability of the function rises.

We now give some prove and intuitions about improving a Bayesian filter system. Consider a Bayesian inverse problem for the linear regression

$$y = Hx + \eta, \tag{5}$$

where $\eta \sim N(0, \Gamma)$ for some covariance matrix $\Gamma, y \in \mathbb{R}^N$,and$x \in \mathbb{R}^p$ is an unknown continuous parameter vec-

tor. To accommodate the case that N is extremely large, we assume that y can be partitioned into $B = N/n$ independent and identically distributed blocks $\{y_1, \ldots, y_B\}$, where each block is of size n and has the covariance matrix V such that Γ =diag$[V, \cdots, V]$.

Let $\pi(x)$ denote the prior density function of x, which is assumed to be differentiable with respect to x. Let $\pi(x|y)$ denote the posterior distribution. To develop an efficient algorithm for simulating from $\pi(x|y)$, which is scalable with respect to both the sample size N and the dimension p, we reformulate the model as a state-space model through subsampling and Langevin diffusion:

$$x_t = x_{t-1} + \epsilon_t \frac{n}{2N} \nabla \log \pi(x_{t-1}) + w_t, y_t = H_t x_t + v_t, \tag{6}$$

where $w_t \sim N(0, \frac{n}{N}\epsilon_t I_p) = N(0, \frac{n}{N} Q_t)$, i.e., $Q_t = \epsilon_t I_p$, y_t denotes a data block randomly drawn from $\{y_1, \ldots, y_B\}, v_t \sim N(0, V_t)$ with $V_t = V$,and H_t is a submatrix of H extracted with the corresponding y_t In the state-space model, at each stage t, the state evolves according to an Euler-discretized Langevin equation of the prior distribution, and the measurement varies with subsampling. As shown in [14] of the Supplementary Material, the filtering distribution of the state-space model converges to the target posterior $\pi(x|y) \to \infty$, provided that ϵ_t decays to zero in an appropriate rate and the matrix V satisfies some regularity conditions.

3.3. classifier-free diffusion(not implemented)

We use classifier-free guidance with low-temperature sampling, which we hypothesize to implicitly perform dynamics programming to capture the best behaviors in the dataset and glean return maximizing trajectories. Our straightforward conditional generative modeling formulation outperforms existing approaches on standard D4RL tasks[4]

As conventional reinforcement learning groups always propose, action sequence prediction without future reward estimation becomes meaningless. In another word, it may be simpler to think of the problem differently, we are trying to solve a non-episodic problem here, in that there is no natural separation of the process into separate meaningful episodes. While no physical process is actually infinite, action prediction without any computed loss is just a theoretical nicety. As a consequence, when we're running a environment in simulation, or multiple versions of it for training purposes, we don't treat them as episodes mathematically(abandon the pseudo episode), there is no terminal state, then we can never obtain a simple episodic return value and can't get the action distribution transfer model conditioned on the sensor inputs.

Because the above properties, we must train our planning generation model with specific targets labeled for each input sequence. The conditional information includes the robot's pose prediction generated from last episode, and the current timestamp's sensor signals as observations. According to [9], score-based diffusion model intrinsically represents a form of Langevin dynamics, which can be embedded into a probabilistic filter system, for example, a Kalman filter system. This gives us the insight, that our pipeline actually substitutes the Langevin dynamics with diffusion model in a embedding Kalman filter.

References

[1] Anurag Ajay, Yilun Du, Abhi Gupta, Joshua Tenenbaum, Tommi Jaakkola, and Pulkit Agrawal. Is conditional generative modeling all you need for decision-making?, 2023. 1

[2] Aaron D. Ames, Xiangru Xu, Jessy W. Grizzle, and Paulo Tabuada. Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control*, 62(8):3861–3876, 2017. 1

[3] Aaron D. Ames, Samuel Coogan, Magnus Egerstedt, Gennaro Notomista, Koushil Sreenath, and Paulo Tabuada. Control barrier functions: Theory and applications, 2019. 1

[4] Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4rl: Datasets for deep data-driven reinforcement learning, 2021. 4

[5] Binghao Huang, Yuanpei Chen, Tianyu Wang, Yuzhe Qin, Yaodong Yang, Nikolay Atanasov, and Xiaolong Wang. Dynamic handover: Throw and catch with bimanual hands, 2023. 2

[6] Bauersfeld L. Loquercio A. et al Kaufmann, E. Champion-level drone racing using deep reinforcement learning. *Nature*, 2023. 2

[7] Jeongmin Lee, Minji Lee, and Dongjun Lee. Uncertain pose estimation during contact tasks using differentiable contact features, 2023. 2

[8] Jinsun Liu, Challen Enninful Adu, Lucas Lymburner, Vishrut Kaushik, Lena Trang, and Ram Vasudevan. Radius: Risk-aware, real-time, reachability-based motion planning, 2023. 2

[9] Calvin Luo. Understanding diffusion models: A unified perspective, 2022. 4

[10] Adithyavairavan Murali, Arsalan Mousavian, Clemens Eppner, Chris Paxton, and Dieter Fox. 6-dof grasping for target-driven object manipulation in clutter, 2020. 2

[11] Haochen Shi, Huazhe Xu, Samuel Clarke, Yunzhu Li, and Jiajun Wu. Robocook: Long-horizon elasto-plastic object manipulation with diverse tools, 2023. 1

[12] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J. Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation, 2019. 2

[13] Minsung Yoon, Mincheul Kang, Daehyung Park, and Sung-Eui Yoon. Learning-based initialization of trajectory optimization for path-following problems of redundant manipulators. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9686–9692, 2023. 2

[14] Peiyi Zhang, Qifan Song, and Faming Liang. A langevinized ensemble kalman filter for large-scale static and dynamic learning, 2021. 4