NDPP-Grasp: Non-Differentiable Physical Plausibility Constraint-Guided Task-Oriented Dexterous Grasp Generation

Qiuchi Xiang
Lancaster University
UK
q.xiang1@lancaster.ac.uk

Haoxuan Qu
Lancaster University
UK
h.qu5@lancaster.ac.uk

Hossein Rahmani
Lancaster University
UK
h.rahmani@lancaster.ac.uk

Jun Liu*
Lancaster University
UK
j.liu81@lancaster.ac.uk

Abstract

Task-oriented dexterous grasp generation aims to produce dexterous grasp poses that are both physically plausible and functionally suitable for specified manipulation tasks. Existing diffusion-based methods often address these two requirements in a decoupled manner: they first train a grasp diffusion model for task alignment and then rely on post-generation refinement to improve physical plausibility. However, this after-the-fact correction strategy applies physical plausibility guidance only once the grasp has already been generated, leaving the generation trajectory itself unguided by physical constraints and potentially leading to suboptimal grasps. To address this problem, we propose a novel framework that directly injects physical plausibility guidance into the denoising process of a task-aligned grasp diffusion model in a practical and effective manner, even when physical plausibility constraints are non-differentiable. This allows physical plausibility to shape grasp generation throughout denoising while preserving task alignment. Extensive experiments demonstrate the efficacy of our framework.

1 Introduction

Task-oriented dexterous grasp generation aims to synthesize dexterous hand poses for a given object and a specified manipulation task, such that the object can be grasped in a way both physically plausible and suitable for performing the task [1, 2, 3, 4], as illustrated in Fig. 1. This problem is central to dexterous robotic manipulation because the grasp quality strongly affects whether the subsequent manipulation can be successfully executed [5, 6, 7]. Since dexterous robotic manipulation is useful in diverse practical scenarios, the significance of task-oriented dexterous grasp generation spans a wide range of applications [8, 9, 10], such as industrial manufacturing [11], household services [12], and medical rehabilitation [8]. Owing to its significance, task-oriented dexterous

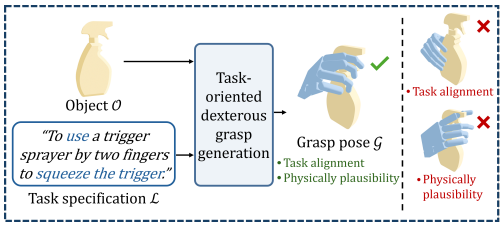

Figure 1: Illustration of task-oriented dexterous grasp generation.

*Corresponding Author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

grasp generation has recently received increasing attention [4, 3, 13, 2]. In particular, motivated by strong generative capability of diffusion models, diffusion-based approaches have attracted growing interest as a promising direction for this problem [4, 3, 13].

Despite recent progress, task-oriented dexterous grasp generation remains challenging because a practically useful grasp needs to satisfy two heterogeneous requirements simultaneously [1, 14]. Specifically, generated grasps need to align with manipulation intents in task specifications, while also being physically plausible in real-world interactions. These requirements are both essential. A grasp that satisfies the task specification may still be unusable if it is physically implausible, for example, if it penetrates the object. Conversely, a physically plausible grasp may be functionally inappropriate for specified manipulation tasks. Therefore, effectively handling both requirements is central to generating practically useful grasps.

To handle this challenge, a seemingly straightforward strategy is to train a single generative model, such as a diffusion model, to satisfy both requirements jointly [15, 16]. However, this strategy can be difficult to realize effectively in practice. On the one hand, incorporating the two heterogeneous requirements into a single training objective makes optimization challenging [3, 2, 1]. On the other hand, while alignment with the task specification can often be supervised in a differentiable manner, physical plausibility is more difficult to impose during training. Many physical plausibility criteria, such as hard hand-object non-penetration and self-collision checks, are inherently non-differentiable [17, 18]. This makes them difficult to directly optimize and often requires differentiable surrogate losses instead [3, 2, 4]. Although such surrogate objectives provide tractable training signals, the gap between the surrogate losses and the original physical plausibility constraints can limit the effectiveness of training [18, 3]. As a result, directly training a single model to satisfy both requirements often leads to suboptimal grasp generation performance [4, 3, 13].

To avoid directly optimizing both requirements within a single generative model, existing methods [4, 3, 2] often adopt a decoupled strategy. Instead of operating in a single process, they decompose task-oriented dexterous grasp generation into two stages. In the first stage, they focus on task alignment and train a generative model to produce a dexterous grasp that matches the task specification. In the second stage, they apply a refinement process to adjust the generated grasp and improve its physical plausibility [4, 3]. However, although such decoupled two-stage methods can significantly improve task alignment by allowing the diffusion model to focus on matching the task specification, they still have clear limitations. First, since physical plausibility constraints are imposed only after generation, the generated grasp may have already drifted far into physically implausible regions during the denoising process, making it difficult for the refinement stage to recover a plausible grasp [19, 20]. Second, in existing methods [4, 3], the post-generation refinement stage still often relies on differentiable surrogate objectives for physical plausibility constraints [21, 22], and thus remains limited by the mismatch with original non-differentiable physical plausibility criteria [18, 3].

As a result, the above two strategies each lead to their own problems in task-oriented dexterous grasp generation. Hence, in this work, we aim to address these problems from a different perspective. Our key idea is to first train a grasp diffusion model for task alignment, as in existing decoupled two-stage methods [4, 13]. Then, instead of using a post-generation refinement stage, we inject guidance from physical plausibility constraints into the denoising process of the trained diffusion model. This design differs from both existing strategies. Unlike methods that impose task alignment and physical plausibility together during training [15, 16], our framework keeps models focused on task alignment and avoids mixing heterogeneous objectives in the diffusion training process, which can otherwise compromise training effectiveness [3, 2]. Unlike post-generation refinement methods [4, 13], physical plausibility is not applied only after a completely generated grasp. Instead, it progressively shapes the grasp throughout the denoising process. In this way, our framework can avoid mixing heterogeneous objectives while allowing physical plausibility to guide grasp generation before it is complete.

However, effectively realizing this idea remains challenging. Existing techniques for guiding the grasp denoising process typically rely on differentiable objectives to construct guidance signals [12, 23]. Therefore, naively applying them to physical plausibility constraints would require replacing the original non-differentiable criteria with differentiable surrogate losses, leading to a mismatch of interest. To address this challenge, we propose **N**on-**D**ifferentiable **P**hysical **P**lausibility Constraint-Guided Task-Oriented Dexterous **Grasp** Generation (**NDPP-Grasp**), a novel diffusion-based framework for task-oriented dexterous grasp generation. To the best of our knowledge, NDPP-Grasp is the first framework that enables progressive denoising-time guidance from non-differentiable physical

plausibility constraints for diffusion-based task-oriented dexterous grasp generation, without relying on differentiable surrogate losses. Below, we outline how NDPP-Grasp achieves this goal.

We first observe that progressively steering the denoising trajectory toward a final grasp that satisfies physical plausibility constraints can be viewed as a sequential decision-making problem under the denoising dynamics of grasp diffusion generation. At each denoising step, the guidance determines how to adjust the current grasp sample, so that the final denoised grasp can better satisfy the non-differentiable physical plausibility constraints. This viewpoint naturally connects our problem to stochastic optimal control [24, 25, 26, 27, 28], which also studies how to make decisions over time to drive a stochastic dynamic system toward desired goals. Motivated by this connection, in our framework, we first explore whether stochastic control can help realize non-differentiable guidance during the denoising process. Through this exploration, we observe that, by combining this viewpoint with the intrinsic structure of grasp diffusion generation, it can be theoretically shown that non-differentiable guidance during grasp denoising can indeed be achieved using only forward evaluations of physical plausibility constraints, without requiring these constraints to be differentiable. This result provides a principled and effective way to directly apply non-differentiable physical plausibility constraints during grasp denoising, without relying on differentiable surrogate losses. However, while effective in principle, directly computing the derived formulation can be computationally intractable, thus impractical (see Sec. 4.2 for details). To address this issue, we further exploit the structure of grasp diffusion generation and propose an amortized lookahead strategy in our framework. With this strategy, non-differentiable guidance during grasp denoising can be performed practically in real time. Together, these designs enable our framework to achieve practical and effective non-differentiable physical plausibility guidance during the denoising process of task-oriented dexterous grasp generation.

Our contributions are: (1) We propose NDPP-Grasp, a novel framework for task-oriented dexterous grasp generation. To the best of our knowledge, this is the first effort to enable non-differentiable physical plausibility guidance during the denoising process for task-oriented dexterous grasp generation, without relying on differentiable surrogate losses. (2) To make such denoising-time guidance practical and effective, we introduce several theoretically grounded design choices into our framework. (3) NDPP-Grasp achieves superior performance on the evaluated benchmarks.

2 Related Works

Dexterous Grasp Generation. Dexterous grasp generation is a core problem in dexterous robotic manipulation. Under this task, prior studies can be broadly categorized into two groups: task-agnostic [29, 30, 12, 31] and task-oriented [4, 32, 33, 3]. The former aims to generate physically plausible grasp poses for a given object, while the latter aims to generate grasp poses that are both physically plausible and functionally suitable for the specified manipulation task. Since downstream manipulation often requires task-specific grasping behaviors, task-oriented dexterous grasp generation has recently attracted increasing attention [3, 4, 13]. Specifically, with the emergence of diffusion models as powerful generative models [34, 35, 36], diffusion-based methods have become a promising direction for this problem. DexGYS [3] cascades a diffusion model with transformer-based refinement model for task alignment and physical plausibility. DexTOG [4] refines diffusion-generated grasps with a test-time collision handling module to reduce penetration. Text2Grasp [13] uses text-guided diffusion followed by hand-object contact optimization that improves physical plausibility.

Despite their different designs, these works share a common paradigm of improving physical plausibility through a post-generation refinement stage, where physical plausibility guidance is applied only after the grasp has already been generated. In contrast, we propose a novel framework that enables directly incorporating physical plausibility guidance into the denoising process of a learned diffusion-based dexterous grasp generation model in an effective and practical manner.

Optimal Control. Optimal control provides a classical perspective in control theory for studying how to choose actions over time to drive a dynamical system toward desired behaviors [37, 24]. Recently, it has been studied in various areas [38, 39, 26, 27, 40, 41, 42, 28], such as robust locomotion [40], autonomous racing [41], and trajectory optimization [42]. Inspired by these works, in this work, we propose a novel framework that leverages an optimal control perspective and further exploits the intrinsic structure of the grasp denoising process, collectively enabling physical plausibility guidance

to be incorporated into the denoising process of a task-aligned grasp diffusion model in a practical and effective manner.

3 Preliminaries

Problem definition. Let $\mathcal{O}$ denote an object representation, typically in point-cloud form, and let $\mathcal{L}$ denote a task specification, typically given as a natural-language sentence. Given $\mathcal{O}$ and $\mathcal{L}$, task-oriented dexterous grasp generation aims to synthesize a dexterous hand grasp pose $\mathcal{G} = (r, t, q)$, such that the object can be grasped by the hand in a way that is both physically plausible and suitable for performing the task. Here, $r \in SO(3)$ denotes the wrist rotation, $t \in \mathbb{R}^3$ denotes the wrist translation, and $q \in \mathcal{Q} \subseteq \mathbb{R}^J$ denotes the joint configuration of a J-DoF dexterous hand. The space $\mathcal{Q}$ denotes the valid joint configuration space defined by the hand kinematic limits.

Diffusion-based task-oriented dexterous grasp generation. We next describe the diffusion-based process for generating task-oriented dexterous grasps. Here, since the grasp pose $\mathcal{G}$ contains a wrist rotation r in the non-Euclidean manifold $SO(3)$, directly applying standard Gaussian diffusion to generate $\mathcal{G}$ is not straightforward. Therefore, existing diffusion-based grasp generation methods [3, 4] typically generate $\mathcal{G}$ through the following procedure. They first perform Gaussian diffusion in a Euclidean parameter space $\mathbb{R}^d$, progressively denoising Gaussian noise $\mathbf{y}_T \sim \mathcal{N}(0, I)$ into a target grasp representation $\mathbf{y}_0$ in this space, via a reverse-time stochastic differential equation (SDE). The generated representation $\mathbf{y}_0$ is then decoded into the grasp pose $\mathcal{G}$ through a mapping function Φ, i.e., $\mathcal{G} = \Phi(\mathbf{y}_0)$. Specifically, the reverse-time SDE from $s = T$ to $s = 0$ can be written as:

$$\mathrm{d}\mathbf{y}_s = -\beta(s)\Big[\tfrac{1}{2}\mathbf{y}_s + \nabla_{\mathbf{y}_s} \log p_s(\mathbf{y}_s \mid \mathcal{O}, \mathcal{L})\Big]\,\mathrm{d}s + \sqrt{\beta(s)}\,\mathrm{d}\bar{\mathbf{w}}_s, \tag{1}$$

where $\beta(s)$ is the noise schedule, $\nabla_{\mathbf{y}_s} \log p_s(\mathbf{y}_s \mid \mathcal{O}, \mathcal{L})$ is conditional score function estimated by the diffusion model, $\mathrm{d}\bar{\mathbf{w}}_s$ is a Wiener process, and $\mathrm{d}s$ denotes a negative infinitesimal timestep.

4 Methodology

In task-oriented dexterous grasp generation, the goal is to generate grasps that are both task-aligned and physically plausible. Existing strategies address this challenge in different ways, but each has its own limitations. If physical plausibility constraints are added to the grasp diffusion training objective together with task alignment constraints, training can become complicated, and both task alignment and physical plausibility may be compromised [3, 2]. If physical plausibility is instead improved by a separate post-generation refinement step, the guidance is applied only after grasp generation, which can limit its effectiveness [20, 19]. In this work, we address this problem from a different perspective. We first train a grasp diffusion model to generate task-aligned grasps. Then, during inference, we inject physical plausibility guidance directly into the denoising process of the trained model. In this way, training can focus on task alignment, while physical plausibility is enforced during generation.

However, making this perspective practically effective is non-trivial. Existing techniques for guiding the grasp denoising process [12, 23] usually rely on differentiable objectives to provide guidance signals. Therefore, naively applying them to non-differentiable physical plausibility constraints would require replacing these constraints with differentiable surrogate losses, leading to a mismatch in optimization target. To address this challenge, we propose NDPP-Grasp, a novel and theoretically grounded framework for non-differentiable physical plausibility guidance in grasp diffusion. It is built upon two key design choices. First, motivated by previous approaches on optimal control and its connections to diffusion denoising process [24, 26, 27, 28], and exploiting the intrinsic structure of grasp diffusion denoising, we adopt a similar approach to obtain a denoising-time guidance mechanism that only requires forward evaluations of physical plausibility constraints, without requiring the constraint evaluator to be differentiable (Sec. 4.1). Second, although this guidance mechanism is effective in principle, directly computing it can be computationally intractable, as elaborated in Sec. 4.2. We therefore design an amortized lookahead strategy that makes the mechanism practically usable, enabling effective non-differentiable physical plausibility guidance while maintaining real-time grasp generation (Sec. 4.2).

4.1 Non-Differentiable Physical Plausibility Guidance for Grasp Denoising

In this section, we introduce the non-differentiable denoising-time guidance mechanism outlined above. Specifically, given a grasp diffusion model trained for task alignment, our goal is to guide its denoising process directly with non-differentiable physical plausibility constraints, without first converting them into differentiable surrogate losses. To derive such a guidance mechanism, below, we first specify what final grasps the guided denoising process should favor, which provides the target objective that the denoising-time guidance should optimize.

Target objective formulation. A high-quality guided grasp should satisfy two requirements: it should remain task-aligned, and it should be physically plausible. For task alignment, since the diffusion model has been trained to generate task-aligned grasps conditioned on the object representation $\mathcal{O}$ and the task specification $\mathcal{L}$, preserving task alignment means keeping the final denoised grasp representation $\mathbf{y}_0$ close to the conditional distribution $p(\mathbf{y}_0 \mid \mathcal{O}, \mathcal{L})$ learned by the trained model. To encourage physical plausibility, let c denote a physical plausibility constraint, and let f_c denote its constraint evaluator, which measures the violation of c for a grasp pose, with lower values indicating better constraint satisfaction. Following [43, 44], we model satisfaction of this constraint as an energy-based likelihood $p(c \mid \mathbf{y}_0) \propto \exp\big(-f_c(\mathbf{y}_0)\big)$. This likelihood assigns higher probability to grasps that better satisfy the physical plausibility constraint.

Combining the task-aligned generation distribution with this physical plausibility likelihood, the target objective can therefore be modeled as sampling $\mathbf{y}_0$ from the posterior

$$p(\mathbf{y}_0 \mid \mathcal{O}, \mathcal{L}, c) \;\propto\; p(\mathbf{y}_0 \mid \mathcal{O}, \mathcal{L})\, p(c \mid \mathbf{y}_0) \;=\; p(\mathbf{y}_0 \mid \mathcal{O}, \mathcal{L})\, \exp\big(-f_c(\mathbf{y}_0)\big), \tag{2}$$

where $p(\mathbf{y}_0 \mid \mathcal{O}, \mathcal{L})$ preserves task alignment, while $\exp(-f_c(\mathbf{y}_0))$ enforces physical plausibility.

Connection to stochastic optimal control. Having formulated the target posterior in Eq. 2, we now study how to guide the diffusion denoising process with non-differentiable physical plausibility constraints, so that the final samples follow this posterior.

To this end, we first draw inspiration from existing methods that guide denoising with differentiable objectives [12, 23]. These methods usually introduce an additional guidance term $\boldsymbol{\eta}_s(\mathbf{y}_s, s)$ into the reverse-time SDE in Eq. 1, leading to the guided denoising dynamics:

$$\mathrm{d}\mathbf{y}_s = -\beta(s)\Big[\tfrac{1}{2}\mathbf{y}_s + \nabla_{\mathbf{y}_s} \log p_s(\mathbf{y}_s \mid \mathcal{O}, \mathcal{L}) + \boldsymbol{\eta}_s(\mathbf{y}_s, s)\Big]\,\mathrm{d}s + \sqrt{\beta(s)}\,\mathrm{d}\bar{\mathbf{w}}_s, \tag{3}$$

In differentiable cases, $\boldsymbol{\eta}_s$ is typically constructed from the gradient of the constraint evaluator, e.g., $\boldsymbol{\eta}_s(\mathbf{y}_s, s) = -\nabla_{\mathbf{y}_s} f_c(\mathbf{y}_s)$, where we slightly abuse notation and let f_c also denote the evaluator of a differentiable constraint. This observation provides a useful way to reformulate our problem. In Eq. 3, constraint guidance is encoded by the additional denoising drift term $\boldsymbol{\eta}_s$. In our non-differentiable setting, the key question, therefore, becomes how to construct a proper $\boldsymbol{\eta}_s$ without relying on such gradients. This reformulation gives us a concrete target: we need to find a guidance term that can play the role of constraint guidance in the denoising dynamics, even when the constraint evaluator itself is non-differentiable. However, the problem remains challenging: without analytic gradients, it is unclear how to design $\boldsymbol{\eta}_s$ so that the denoising process can be steered toward satisfying the non-differentiable physical plausibility constraints.

To seek a way to construct such a guidance term, we first examine Eq. 3 more closely. We observe that $\boldsymbol{\eta}_s$ introduces an additional constraint-driven signal at each denoising step. This makes guided denoising resemble a sequential decision-making problem: at each step, one needs to decide how to guide the current sample so that the final grasp better satisfies the target constraints. This structure naturally connects to stochastic optimal control [24, 26, 27, 28], which similarly studies how to choose control signals over time to drive a stochastic dynamical system toward a desired objective.

Motivated by this connection, we observe that an optimal control perspective can help construct $\boldsymbol{\eta}_s$ for non-differentiable physical plausibility constraints. In particular, viewing $\boldsymbol{\eta}_s$ as a control signal allows its construction to be formulated as a concrete optimization problem that aims to improve final constraint satisfaction under the denoising dynamics. It can further be observed that, by leveraging the intrinsic structure of grasp diffusion denoising, this optimization problem admits an analytical solution. This solution yields a gradient-free guidance term that can be computed using only forward evaluations of the physical plausibility constraint evaluator, thereby enabling the desired non-differentiable denoising-time guidance mechanism. Below, the formulation of this optimization problem and its analytical solution are detailed.

Optimization problem formulation. We now formulate the construction of the guidance term $\boldsymbol{\eta}_s$ as a concrete optimization problem from a stochastic optimal control perspective.

Step 1. We first rewrite Eq. 3 into a standard control-system form. Since the reverse-time SDE is run from $s = T$ to 0, while optimal control is usually in forward time, we introduce the forward-time denoising state $\mathbf{g}_\tau = \mathbf{y}_{T-\tau}$, where τ runs from 0 to T. Under this change of variable, Eq. 3 can be equivalently written as the following control-affine stochastic dynamical system (Appendix E.1):

$$\mathrm{d}\mathbf{g}_\tau = \mathbf{F}(\mathbf{g}_\tau, \tau)\mathrm{d}\tau + \sigma(\tau)\big(\mathbf{u}_\tau(\mathbf{g}_\tau, \tau)\mathrm{d}\tau + \mathrm{d}\mathbf{w}_\tau\big), \tag{4}$$

where $\mathbf{u}_\tau(\mathbf{g}_\tau, \tau) = \sqrt{\beta(T-\tau)}\boldsymbol{\eta}_{T-\tau}(\mathbf{g}_\tau, T-\tau)$ is the forward-time control signal induced by the denoising-time guidance term $\boldsymbol{\eta}_s$, $\mathbf{F}(\mathbf{g}_\tau, \tau) = \frac{1}{2}\beta(T-\tau)\mathbf{g}_\tau + \beta(T-\tau)\nabla_{\mathbf{g}_\tau} \log p_{T-\tau}(\mathbf{g}_\tau \mid \mathcal{O}, \mathcal{L})$ is the uncontrolled drift, $\sigma(\tau) = \sqrt{\beta(T-\tau)}$ is the diffusion coefficient, and $\mathrm{d}\mathbf{w}_\tau$ is a standard forward-time Wiener process. Here, $\mathbf{F}$ represents the original task-aligned denoising dynamics, while $\mathbf{u}_\tau$ represents the additional control signal for physical plausibility guidance.

Because $\mathbf{u}_\tau$ and $\boldsymbol{\eta}_s$ are related by a deterministic rescaling and time reversal, constructing $\boldsymbol{\eta}_s$ is equivalent to constructing $\mathbf{u}_\tau$. Therefore, in the remainder of this section, we work directly with $\mathbf{u}_\tau$.

Step 2. With this equivalence, the problem becomes finding an optimal control law $\mathbf{u}_\tau$ for the forward-time system in Eq. 4. Guided by the target posterior in Eq. 2, the control law should guide the denoising trajectory toward final grasps with better physical plausibility satisfaction, while keeping the controlled dynamics close to the original task-aligned denoising dynamics. Since $\mathbf{u}_\tau \equiv 0$ recovers the original denoising dynamics, this closeness can be encouraged by penalizing the magnitude of the control signal. This leads to the following stochastic optimal control objective:

$$\mathbf{u}^*_{\tau:T} \;=\; \arg\min_{\mathbf{u}_{\tau:T}} \; \mathbb{E}_{\mathcal{P}^{\mathbf{u}}}\left[\int_\tau^T \tfrac{1}{2}\|\mathbf{u}_\nu\|^2 \,\mathrm{d}\nu \;+\; f_c(\mathbf{g}_T) \,\middle|\, \mathbf{g}_\tau\right], \tag{5}$$

where $\mathbf{u}_{\tau:T} = \{\mathbf{u}_\nu\}_{\nu\in[\tau,T]}$ is the sequence of control signals applied from time τ to T, and $\mathcal{P}^{\mathbf{u}}$ is the controlled denoising dynamics induced by these control signals. The terminal cost $f_c(\mathbf{g}_T)$ penalizes physical plausibility violations of the final grasp, while the running cost $\frac{1}{2}\|\mathbf{u}_\nu\|^2$ penalizes the control magnitude to keep the guided dynamics close to the original task-aligned denoising dynamics. This gives the concrete optimization problem for constructing the desired control signal.

Derivation of the gradient-free optimal control law. Having formulated the optimization problem in Eq. 5, its solution gives the optimal control law $\mathbf{u}^*_\tau$. Although this optimization problem appears difficult to solve in general, it can be shown that, by leveraging the intrinsic structure of grasp diffusion denoising, including the Markov structure of denoising trajectories and the Gaussian noise structure, the following analytical form of $\mathbf{u}^*_\tau$ can be obtained in our context:

$$\mathbf{u}^*_\tau(\mathbf{g}_\tau, \tau)\,\mathrm{d}\tau = \frac{\mathbb{E}_{\mathcal{P}^0}\big[\exp\big(-f_c(\mathbf{g}_T)\big)\,\mathrm{d}\mathbf{w}_\tau \,\big|\, \mathbf{g}_\tau\big]}{\mathbb{E}_{\mathcal{P}^0}\big[\exp\big(-f_c(\mathbf{g}_T)\big) \,\big|\, \mathbf{g}_\tau\big]}. \tag{6}$$

Here, $\mathcal{P}^0$ denotes the uncontrolled grasp denoising dynamics, i.e., the dynamics in Eq. 4 with $\mathbf{u}_\tau \equiv 0$. The constraint evaluator f_c measures the physical plausibility violation of final denoised grasp $\mathbf{g}_T$.

Eq. 6 is the key result of this section. It gives the optimal forward-time control signal for constraint-guided grasp denoising. Since $\mathbf{u}_\tau$ is deterministically related to the original guidance term $\boldsymbol{\eta}_s$ through the change of variables in Eq. 4, solving for $\mathbf{u}^*_\tau$ equivalently provides the desired denoising guidance. Importantly, Eq. 6 only requires forward evaluations of f_c along uncontrolled denoising trajectories without differentiating f_c. Thus, this control law enables non-differentiable physical plausibility guidance for grasp denoising without requiring differentiability or gradient computation.

4.2 Practical Instantiation via Amortized Lookahead

Above, we obtain a principled form of the control law $\mathbf{u}^*_\tau$ that enables non-differentiable physical plausibility guidance for grasp denoising, without requiring differentiability or gradient computation.

Remaining practical challenge. Yet, this principled control law cannot be directly computed in practice. As shown in Eq. 6, $\mathbf{u}^*_\tau$ is expressed as an expectation under the uncontrolled grasp denoising dynamics $\mathcal{P}^0$, which is intractable to compute in closed form. Therefore, to use this control law as a practical guidance algorithm, we need an effective way to instantiate this expectation.

A natural finite-lookahead instantiation. Since Eq. 6 involves an expectation under the uncontrolled denoising dynamics, a direct way to instantiate it is Monte Carlo sampling [45]. Specifically, at each denoising step τ, we sample M uncontrolled future denoising trajectories. Then, for each sampled trajectory, we evaluate the physical plausibility of its final grasp. Thus, the key quantity we need from the m-th trajectory is its terminal grasp $\mathbf{g}_T^{(m)}$, which allows us to compute the terminal physical plausibility cost $f_c(\mathbf{g}_T^{(m)})$. With these sampled trajectories and their terminal costs, the expectation in Eq. 6 can be instantiated by Monte Carlo sampling. See Appendix G.1 for more details.

This reduces the practical problem of instantiating $\mathbf{u}_\tau^*$ to the problem of obtaining terminal grasps $\mathbf{g}_T^{(m)}$ for the sampled future trajectories. In principle, this can be achieved by simulating the uncontrolled denoising dynamics step by step, i.e., $\mathbf{g}_\tau \to \mathbf{g}_{\tau+1}^{(m)} \to \cdots \to \mathbf{g}_T^{(m)}$. However, this requires simulating the entire remaining denoising process for every sampled trajectory. Since the guidance term needs to be instantiated at every denoising step, and each instantiation requires M trajectories, this direct simulation introduces a substantial computational burden. This is particularly undesirable for dexterous grasp generation, which is often used in robotic manipulation pipelines where real-time inference is important [46].

To tackle this problem, following the common use of terminal-data prediction in diffusion guidance [44, 47], a more practical alternative is to use Tweedie-based finite lookahead [48]. Specifically, instead of simulating each trajectory all the way to the terminal step, we only simulate H future denoising steps, where H is a hyperparameter, i.e., $\mathbf{g}_\tau \to \mathbf{g}_{\tau+1}^{(m)} \to \cdots \to \mathbf{g}_{\tau+H}^{(m)}$. Then, we use Tweedie's formula [48] to predict the terminal grasp from the lookahead endpoint: $\hat{\mathbf{g}}_T^{(m)} = \mathcal{T}_{\tau+H}(\mathbf{g}_{\tau+H}^{(m)})$, where $\mathcal{T}_{\tau+H}(\cdot)$ denotes the Tweedie prediction operator at time $\tau + H$, and $\hat{\mathbf{g}}_T^{(m)}$ denotes the predicted terminal grasp. We then evaluate $f_c(\hat{\mathbf{g}}_T^{(m)})$ and use it in the Monte Carlo instantiation of Eq. 6. In this way, we obtain a natural finite-lookahead instantiation of Eq. 6: it keeps the Monte Carlo structure of the control-law instantiation, but replaces expensive full future trajectory simulation with Tweedie-based terminal-grasp prediction from H-step lookahead trajectories.

Limitation of this natural solution. Although the natural finite-lookahead instantiation is more practical than simulating the full future trajectory, it still suffers from a key tension between effectiveness and practicality. Specifically, to obtain a reliable terminal grasp prediction, the lookahead horizon H is usually expected to be sufficiently large, so that the terminal grasp prediction can rely on more future denoising context when applying Tweedie's formula. However, a larger H also means that, at every denoising step, each sampled trajectory still needs to be simulated from $\mathbf{g}_\tau$ to $\mathbf{g}_{\tau+H}$. Since this process is repeated for M trajectories at every denoising step, the instantiation can still be costly during sampling. In contrast, choosing a smaller H reduces computation, but makes the terminal grasp prediction rely more heavily on noisier intermediate states and can weaken guidance effectiveness. Therefore, the above natural finite-lookahead instantiation still does not make Eq. 6 both practical and effective in a satisfactory way.

Our strategy. To address this issue, we propose an amortized lookahead strategy. The key idea behind our strategy is that consecutive H-step lookahead instantiations have highly overlapping future time ranges. Specifically, at denoising step τ, the natural finite-lookahead instantiation samples future states $(\mathbf{g}_{\tau+1}^{(m)}, \ldots, \mathbf{g}_{\tau+H}^{(m)})$ for each trajectory. At the next denoising step $\tau + 1$, if we performed a fresh instantiation from scratch, we would need to sample new future states from the actual current state $\mathbf{g}_{\tau+1}$, namely $(\tilde{\mathbf{g}}_{\tau+2}^{(m)}, \ldots, \tilde{\mathbf{g}}_{\tau+H+1}^{(m)})$. These two lookahead instantiations therefore cover same future time indices from $\tau + 2$ to $\tau + H$. This motivates us to ask whether, at denoising step $\tau + 1$, we can reuse the already generated states $(\mathbf{g}_{\tau+2}^{(m)}, \ldots, \mathbf{g}_{\tau+H}^{(m)})$ and only add one new transition to obtain $\mathbf{g}_{\tau+H+1}^{(m)}$. If this reuse is valid, then even with a large lookahead horizon H, each subsequent denoising step would only need to add one new transition per trajectory, instead of generating a complete H-step trajectory from scratch. This would directly address the tension above: we could preserve the effectiveness of a larger lookahead horizon while keeping per-step computation practical.

However, such reuse is not directly valid. The stored future states from step τ were generated after passing through the sampled candidate state $\mathbf{g}_{\tau+1}^{(m)}$, whereas the fresh instantiation at step $\tau + 1$ should start from the actual current state $\mathbf{g}_{\tau+1}$. Since these two starting states are different, directly treating the stored future states as freshly sampled ones would correspond to using an incorrect future trajectory distribution. Fortunately, this mismatch can be corrected in our context. Specifically, by

using the Markov property of the uncontrolled grasp denoising dynamics and the Gaussian form of its one-step transition, we show that the mismatch between the desired fresh future segment $(\tilde{\mathbf{g}}^{(m)}_{\tau+2}, \ldots, \tilde{\mathbf{g}}^{(m)}_{\tau+H})$ and the reused stored future segment $(\mathbf{g}^{(m)}_{\tau+2}, \ldots, \mathbf{g}^{(m)}_{\tau+H})$ can be captured by a closed-form correction weight. Formally, we show that (derivation provided in Appendix E.5):

$$\mathbb{E}_{\mathcal{P}^0_{\tau+2:\tau+H}(\cdot|\mathbf{g}_{\tau+1})}\left[\Psi(\tilde{\mathbf{g}}^{(m)}_{\tau+2}, \ldots, \tilde{\mathbf{g}}^{(m)}_{\tau+H})\right] = \mathbb{E}_{\mathcal{P}^0_{\tau+2:\tau+H}(\cdot|\mathbf{g}^{(m)}_{\tau+1})}\left[\rho^{(m)}_{\tau+1}\,\Psi(\mathbf{g}^{(m)}_{\tau+2}, \ldots, \mathbf{g}^{(m)}_{\tau+H})\right]. \quad (7)$$

Here, Ψ denotes a trajectory functional, i.e., a function applied to a lookahead trajectory to compute the quantities needed in the finite-lookahead Monte Carlo instantiation, see Appendix E.4 for details. The distribution $\mathcal{P}^0_{\tau+2:\tau+H}(\cdot \mid \cdot)$ denotes the conditional distribution of the uncontrolled future states from time $\tau+2$ to $\tau+H$, given the state at time $\tau+1$. The scalar $\rho^{(m)}_{\tau+1}$ is the correction weight for reusing the stored future states, whose closed-form expression is derived in Appendix E.5.

Based on this result, we instantiate Eq. 6 through an amortized lookahead procedure. The key idea is to reuse stored future states across consecutive denoising steps with correction weights, and only append one new transition per step to maintain the lookahead horizon. The procedure consists of five operations: the first is performed at the initial denoising step, while the remaining four are repeated at each subsequent denoising step. Below, we detail these five operations. (1) Initialization. At the initial denoising step, since no previous lookahead trajectories are available for reuse, we perform the natural finite-lookahead instantiation once and store the resulting M lookahead trajectories of length H for later reuse. (2) Reuse with correction. At a subsequent denoising step $\tau+1$, for each stored trajectory, we reuse the stored future states $(\mathbf{g}^{(m)}_{\tau+2}, \ldots, \mathbf{g}^{(m)}_{\tau+H})$ instead of simulating them again from scratch. Since these states were generated from the stored candidate state $\mathbf{g}^{(m)}_{\tau+1}$ rather than the actual current state $\mathbf{g}_{\tau+1}$, we compute the correction weight $\rho^{(m)}_{\tau+1}$ and use it to correct the corresponding Monte Carlo contribution. (3) Extension. To maintain an H-step lookahead horizon at denoising step $\tau+1$, we append one additional uncontrolled transition from $\mathbf{g}^{(m)}_{\tau+H}$ to $\mathbf{g}^{(m)}_{\tau+H+1}$ using the original uncontrolled denoising dynamics. (4) Monte Carlo computation. After reuse and extension, we use the resulting future segment $(\mathbf{g}^{(m)}_{\tau+2}, \ldots, \mathbf{g}^{(m)}_{\tau+H}, \mathbf{g}^{(m)}_{\tau+H+1})$ to perform the finite-lookahead Monte Carlo computation at the current state $\mathbf{g}_{\tau+1}$, with its contribution multiplied by $\rho^{(m)}_{\tau+1}$. This gives the current-step Monte Carlo instantiation of the expectation in Eq. 6, see Appendix G.2 for more details. (5) Storage for the next denoising step. After the current guidance term is computed, we store the extended future trajectory $(\mathbf{g}^{(m)}_{\tau+2}, \ldots, \mathbf{g}^{(m)}_{\tau+H+1})$ for reuse in the next denoising step.

Through these five operations, the amortized procedure avoids regenerating a complete H-step lookahead trajectory from scratch at every denoising step. Thus, we can use a sufficiently large lookahead horizon H to improve the reliability of terminal grasp prediction, while after initialization, each denoising step only needs to simulate one additional transition per trajectory. This turns Eq. 6 into a practical and effective algorithm for computing non-differentiable physical plausibility guidance during grasp denoising.

5 Experiments

Datasets & Evaluation Metrics. We evaluate NDPP-Grasp on two widely used datasets: DexGYSNet [3] and DexTOG-80K [4]. On DexGYSNet, we use the standard train-test split [3] and additionally evaluate on an open-set split. Following [3], we evaluate physical quality using success rate (Succ.), Q_1, and maximal penetration depth (Pen.), evaluate task alignment using Fréchet Inception Distance (FID), Chamfer Distance (CD), and Contact Distance (Con.), and evaluate diversity using the standard deviations of translation δ_t, rotation δ_r, and joint angles δ_q. On DexTOG-80K, we use its standard split. Following [4], we report the overall success rate (Overall-Succ.), which captures both physical quality and task alignment. We also follow [4] to report three metrics focused on physical plausibility: Q_1, Pen., and the collision-free rate η_f. On both datasets, we also report the average per-grasp inference time, denoted as "Time" in the tables. See Appendix D for more details.

Experimental Setups & Implementation Details. Across the two datasets, we apply our method to grasp diffusion models trained following the respective training procedures of three recent diffusion-based task-oriented dexterous grasp generation works: Text2Grasp [13], DexTOG [4], and DexGYS [3]. Following [3, 2], we consider three non-differentiable physical plausibility constraints: the hand-object non-penetration constraint, self-collision avoidance, and the joint-limit constraint. We

set hyperparameters $M = 8$ and $H = 20$. All experiments are conducted on NVIDIA A6000 GPU. See Appendix D for more experimental setups and implementation details.

5.1 Main Results

In Tab. 1 and Tab. 2, we report results on DexGYSNet and DexTOG-80K, respectively. As shown, when using the same grasp diffusion model, our method consistently achieves substantial improvements in physical plausibility compared with existing methods that enhance physical plausibility through various post-generation refinement strategies, demonstrating the efficacy of our approach. When compared with other recent task-oriented dexterous grasp generation methods, our method also achieves superior performance, further demonstrating its effectiveness.

Table 1: Performance comparison on DexGYSNet. "Ours (training following X)" denotes our method applied to the grasp diffusion model trained following the training procedure of X (see Appendix F).

Method	Standard train-test split on the DexGYSNet dataset									Open-set train-test split on the DexGYSNet dataset									Time (ms) ↓
	Physical Qual.			Task Align.			Diversity			**Physical Qual.**			Task Align.			Diversity			
	Succ. ↑	Q_1 ↑	Pen. ↓	FID ↓	CD ↓	Con. ↓	δ_t ↑	δ_r ↑	δ_q ↑	Succ. ↑	Q_1 ↑	Pen. ↓	FID ↓	CD ↓	Con. ↓	δ_t ↑	δ_r ↑	δ_q ↑	
GraspCVAE [49]	29.12%	0.054	0.551	31.26	3.138	0.096	0.179	1.762	0.179	24.36%	0.048	0.716	41.57	3.985	0.128	0.158	1.583	0.157	48.3
GraspTTA [50]	43.46%	0.071	0.188	35.41	12.19	0.111	2.111	6.150	3.869	38.71%	0.064	0.246	47.10	15.48	0.148	1.876	5.527	3.438	1873.4
DGTR [51]	51.91%	0.078	0.163	23.31	2.895	0.078	2.037	14.01	4.299	46.23%	0.071	0.215	31.12	3.684	0.104	1.812	12.47	3.821	41.7
SceneDiffuser [31]	62.24%	0.083	0.253	20.44	1.679	0.045	0.346	3.455	0.387	57.48%	0.076	0.330	27.26	2.139	0.060	0.306	3.117	0.343	44.1
AffordDexGrasp [2]	62.81%	0.082	0.228	6.538	1.198	0.039	5.924	53.86	5.918	59.17%	0.077	0.279	8.712	1.523	0.047	5.307	48.58	5.368	493.1
Text2Grasp [13]	60.43%	0.080	0.197	21.83	1.812	0.051	0.398	4.123	0.452	55.36%	0.073	0.272	28.43	2.301	0.065	0.357	3.642	0.405	287.4
Ours (training following [13])	**63.27%**	**0.084**	**0.163**	**21.78**	**1.798**	**0.049**	**0.402**	**4.131**	**0.461**	**58.18%**	**0.077**	**0.224**	**28.37**	**2.287**	**0.063**	**0.361**	**3.651**	**0.413**	**20.5**
DexTOG [4]	62.67%	0.082	0.234	13.82	1.452	0.041	3.827	32.15	3.912	57.82%	0.075	0.253	18.43	1.849	0.054	3.394	28.97	3.472	389.6
Ours (training following [4])	**65.94%**	**0.086**	**0.192**	**13.79**	**1.438**	**0.040**	**3.842**	**32.23**	**3.931**	**61.08%**	**0.079**	**0.213**	**18.27**	**1.836**	**0.052**	**3.412**	**29.04**	**3.489**	**17.5**
DexGYS [3]	63.31%	0.083	0.223	6.538	1.198	0.036	6.118	55.68	6.118	58.42%	0.076	0.291	8.724	1.526	0.048	5.383	48.95	5.392	23.2
Ours (training following [3])	**67.05%**	**0.088**	**0.182**	**6.487**	**1.184**	**0.035**	**6.132**	**55.74**	**6.127**	**61.89%**	**0.081**	**0.239**	**8.683**	**1.519**	**0.047**	**5.391**	**49.12**	**5.401**	**11.6**

Table 2: Performance comparison on DexTOG-80K.

Method	Overall-Succ. ↑	Q_1 ↑	Pen. ↓	η_f ↑	Time (ms) ↓
GraspTTA [50]	3.20%	0.031	0.524	18.7%	1901.7
AffordDexGrasp [2]	22.41%	0.076	0.462	35.2%	501.3
Text2Grasp [13]	17.85%	0.062	0.391	28.7%	308.4
Ours (training following [13])	**20.41%**	**0.067**	**0.323**	**34.2%**	**20.7**
DexGYS [3]	23.87%	0.078	0.448	36.8%	23.6
Ours (training following [3])	**26.53%**	**0.083**	**0.374**	**42.4%**	**11.8**
DexTOG [4]	28.60%	0.093	0.385	50.5%	395.8
Ours (training following [4])	**31.92%**	**0.098**	**0.318**	**56.8%**	**17.7**

5.2 Further Analysis and Ablation Studies

We conduct extensive analyses and ablation studies on the standard train-test split of DexGYSNet, using the grasp diffusion model trained following DexGYS [3]. **More analyses and ablation studies are provided in Appendix A.**

Impact of gradient-free optimal control law. To evaluate the effectiveness of the optimal control law, we compare against the following four variants (details in Appendix A). **Variant A** uses surrogate losses of physical plausibility constraints during training. **Variant B** uses the gradients of surrogates to guide the denoising process at inference time. **Variant C** estimates gradients of the non-differentiable constraints via zeroth-order finite-difference approximation [52]. **Variant D** applies reinforcement learning [53] to approximate gradient guidance for non-differentiable constraints. As shown in Tab. 3, NDPP-Grasp outperforms all variants, demonstrating the effectiveness of the gradient-free optimal control law.

Table 3: Evaluation of the optimal control law.

Method	**Physical Qual.**			Task Align.			Diversity		
	Succ. ↑	Q_1 ↑	Pen. ↓	FID ↓	CD ↓	Con. ↓	δ_t ↑	δ_r ↑	δ_q ↑
Variant A	54.27%	0.071	0.378	12.64	1.987	0.061	4.213	41.28	4.317
Variant B	62.91%	0.084	0.211	6.583	1.192	0.036	6.094	55.21	6.092
Variant C	60.74%	0.079	0.238	6.728	1.203	0.038	6.024	54.41	6.038
Variant D	59.86%	0.077	0.247	6.793	1.214	0.039	5.978	53.97	5.991
Ours	**67.05%**	**0.088**	**0.182**	**6.487**	**1.184**	**0.035**	**6.132**	**55.74**	**6.127**

Impact of the amortized lookahead strategy. To evaluate the effectiveness of our amortized lookahead strategy, we compare two variants. In the (**Complete-trajectory sampling**), at each

denoising step, we sample complete trajectories that denoise until the terminal step, to estimate the optimal control signal. In the (H**-step trajectory resampling**), we only sample H-step trajectories and apply Tweedie's formula to estimate. As shown in Tab. 4, our method achieves comparable performance with substantially lower computational cost, showing the practical effectiveness of our amortized lookahead strategy.

Table 4: Evaluation of the amortized lookahead.

Method	Physical Qual.			Task Align.			Diversity			Time
	Succ. ↑	Q_1 ↑	Pen. ↓	FID ↓	CD ↓	Con. ↓	δ_t ↑	δ_r ↑	δ_q ↑	(ms) ↓
Complete-trajectory sampling	67.06%	0.089	0.183	6.484	1.183	0.034	6.135	55.78	6.130	261.4
H-step trajectory resampling	67.05%	0.088	0.182	6.488	1.184	0.035	6.131	55.73	6.126	109.8
Ours	**67.05%**	**0.088**	**0.182**	**6.487**	**1.184**	**0.035**	**6.132**	**55.74**	**6.127**	**11.6**

Qualitative Results. In addition to quantitative comparisons, we also provide qualitative comparisons in Fig. 2. As shown, our framework produces grasps with strong physical plausibility while preserving alignment with the task specifications, whereas the compared method DexGYS [3] produces physically implausible grasps that fail to satisfy the physical constraints, further showing the efficacy of our method. **More qualitative results and comparisons with more methods are in Appendix B.**

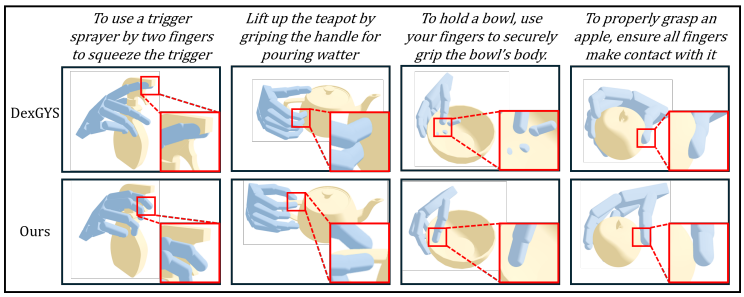

Figure 2: Qualitative comparisons on DexGYSNet, with the key contact regions zoomed in red boxes.

6 Conclusion

In this paper, we have proposed NDPP-Grasp, a novel and theoretically grounded framework for task-oriented dexterous grasp generation that enables practical and effective non-differentiable physical plausibility guidance directly within the denoising process of a task-aligned grasp diffusion model. Extensive experiments demonstrate the effectiveness of our framework.

Acknowledgments and Disclosure of Funding

This work received no external funding. The authors declare no competing interests.

References

[1] Juntao Jian, Xiuping Liu, Zixuan Chen, Manyi Li, Jian Liu, and Ruizhen Hu. G-dexgrasp: Generalizable dexterous grasping synthesis via part-aware prior retrieval and prior-assisted generation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 11447–11457, 2025.

[2] Yi-Lin Wei, Mu Lin, Yuhao Lin, Jian-Jian Jiang, Xiao-Ming Wu, Ling-An Zeng, and Wei-Shi Zheng. Afforddexgrasp: Open-set language-guided dexterous grasp with generalizable-instructive affordance. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 11818–11828, 2025.

[3] Yi-Lin Wei, Jian-Jian Jiang, Chengyi Xing, Xian-Tuo Tan, Xiao-Ming Wu, Hao Li, Mark Cutkosky, and Wei-Shi Zheng. Grasp as you say: Language-guided dexterous grasp generation. Advances in Neural Information Processing Systems, 37:46881–46907, 2024.

[4] Jieyi Zhang, Wenqiang Xu, Zhenjun Yu, Pengfei Xie, Tutian Tang, and Cewu Lu. Dextog: Learning task-oriented dexterous grasp with language condition. IEEE Robotics and Automation Letters, 10(2):995–1002, 2025.

[5] Ruicheng Wang, Jialiang Zhang, Jiayi Chen, Yinzhen Xu, Puhao Li, Tengyu Liu, and He Wang. Dexgraspnet: A large-scale robotic dexterous grasp dataset for general objects based on simulation. In 2023 IEEE International Conference on Robotics and Automation (ICRA), pages 11359–11366. IEEE, 2023.

[6] Kuan Fang, Yuke Zhu, Animesh Garg, Andrey Kurenkov, Viraj Mehta, Li Fei-Fei, and Silvio Savarese. Learning task-oriented grasping for tool manipulation from simulated self-supervision. The International Journal of Robotics Research, 39(2-3):202–216, 2020.

[7] Yufei Guo, Yinan Wu, Haoran Duan, Guiguang Ding, and Jungong Han. PhaseLoRA: Control-regime-conditioned low-rank adaptation for continuous-action vision-language-action policies. arXiv preprint arXiv:2608.15285, 2026.

[8] Yayu Huang, Dongxuan Fan, Haonan Duan, Dashun Yan, Wen Qi, Jia Sun, Qian Liu, and Peng Wang. Human-like dexterous manipulation for anthropomorphic five-fingered hands: A review. Biomimetic Intelligence and Robotics, 5(1):100212, 2025.

[9] Toshiaki Tsuji, Yasuhiro Kato, Gokhan Solak, Heng Zhang, Tadej Petrič, Francesco Nori, and Arash Ajoudani. A survey on imitation learning for contact-rich tasks in robotics. The International Journal of Robotics Research, 2026.

[10] Qiuchi Xiang, Haoxuan Qu, and Jun Liu. Vision-language-action models for diverse application scenarios: A survey of input modalities, 2026.

[11] Jiexin Zhang, Huan Zhao, Kuangda Chen, Guanbo Fei, Xiangfei Li, Yiwei Wang, Zeyuan Yang, Shengwei Zheng, Shiqi Liu, and Han Ding. Dexterous hand towards intelligent manufacturing: A review of technologies, trends, and potential applications. Robotics and Computer-Integrated Manufacturing, 95:103021, 2025.

[12] Yiming Zhong, Qi Jiang, Jingyi Yu, and Yuexin Ma. Dexgrasp anything: Towards universal robotic dexterous grasping with physics awareness. In Proceedings of the Computer Vision and Pattern Recognition Conference, pages 22584–22594, 2025.

[13] Xiaoyun Chang and Yi Sun. Text2grasp: Synthesis of grasps by text prompts for object grasping parts. In International Symposium on Neural Networks, pages 514–524. Springer, 2025.

[14] Juntao Jian, Yi-Lin Wei, Chengjie Mou, Yuhao Lin, Xing Zhu, Yujun Shen, Wei-Shi Zheng, and Ruizhen Hu. Zerodexgrasp: Zero-shot task-oriented dexterous grasp synthesis with prompt-based multi-stage semantic reasoning. arXiv preprint arXiv:2511.13327, 2025.

[15] Tianqiang Zhu, Rina Wu, Jinglue Hang, Xiangbo Lin, and Yi Sun. Toward human-like grasp: Functional grasp by dexterous robotic hand via object-hand semantic representation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(10):12521–12534, 2023.

[16] Jiayi Chen, Yuxing Chen, Jialiang Zhang, and He Wang. Task-oriented dexterous hand pose synthesis using differentiable grasp wrench boundary estimator. In 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 5281–5288. IEEE, 2024.

[17] Jiayi Chen, Wei Zhao, Liangwang Ruan, Baoquan Chen, and He Wang. Robust differentiable collision detection for general objects. arXiv preprint arXiv:2511.06267, 2025.

[18] Anselm Paulus, Andreas René Geist, Pierre Schumacher, Vít Musil, Simon Rappenecker, and Georg Martius. Differentiable simulation of hard contacts with soft gradients for learning and control. In The Fourteenth International Conference on Learning Representations, 2026.

[19] Wonsuhk Jung, Utkarsh Aashu Mishra, Nadun Ranawaka Arachchige, Yongxin Chen, Danfei Xu, and Shreyas Kousik. Joint model-based model-free diffusion for planning with constraints. In 9th Annual Conference on Robot Learning, 2025.

[20] Jinhao Liang, Jacob K Christopher, Sven Koenig, and Ferdinando Fioretto. Simultaneous multi-robot motion planning with projected diffusion models. In International Conference on Machine Learning, pages 37162–37180. PMLR, 2025.

[21] Yana Hasson, Gul Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 11807–11816, 2019.

[22] Patrick Grady, Chengcheng Tang, Christopher D Twigg, Minh Vo, Samarth Brahmbhatt, and Charles C Kemp. Contactopt: Optimizing contact to improve grasps. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 1471–1481, 2021.

[23] Tao Zhong and Christine Allen-Blanchette. Gagrasp: Geometric algebra diffusion for dexterous grasping. In 2025 IEEE International Conference on Robotics and Automation (ICRA), pages 6771–6778, 2025.

[24] Jiongmin Yong and Xun Yu Zhou. Stochastic controls: Hamiltonian systems and HJB equations, volume 43. Springer Science & Business Media, 1999.

[25] Kushagra Pandey, Farrin Marouf Sofian, Felix Draxler, Theofanis Karaletsos, and Stephan Mandt. Variational control for guidance in diffusion models. In International Conference on Machine Learning, pages 47755–47780. PMLR, 2025.

[26] Yujia Huang, Adishree Ghatare, Yuanzhe Liu, Ziniu Hu, Qinsheng Zhang, Chandramouli S Sastry, Siddharth Gururani, Sageev Oore, and Yisong Yue. Symbolic music generation with non-differentiable rule guided diffusion. In Proceedings of the 41st International Conference on Machine Learning, pages 19772–19797, 2024.

[27] Xiaofei Hui, Bo Yan, Haoxuan Qu, Hossein Rahmani, and Jun Liu. Training-free controllable human motion generation under heterogeneous constraints. In European Conference on Computer Vision, pages 276–296. Springer, 2026.

[28] Qinsheng Zhang and Yongxin Chen. Path integral sampler: A stochastic control approach for sampling. In International Conference on Learning Representations, 2022.

[29] Zehang Weng, Haofei Lu, Danica Kragic, and Jens Lundell. Dexdiffuser: Generating dexterous grasps with diffusion models. IEEE Robotics and Automation Letters, 9(12):11834–11840, 2024.

[30] Jiaxin Lu, Hao Kang, Haoxiang Li, Bo Liu, Yiding Yang, Qixing Huang, and Gang Hua. Ugg: Unified generative grasping. In European Conference on Computer Vision, pages 414–433. Springer, 2024.

[31] Siyuan Huang, Zan Wang, Puhao Li, Baoxiong Jia, Tengyu Liu, Yixin Zhu, Wei Liang, and Song-Chun Zhu. Diffusion-based generation, optimization, and planning in 3d scenes. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 16750–16761, 2023.

[32] Weishang Wu, Yifei Shi, Zhizhong Chen, and Zhipong Cai. Partdextog: Generating dexterous task-oriented grasping via language-driven part analysis. arXiv preprint arXiv:2505.12294, 2025.

[33] Kailin Li, Jingbo Wang, Lixin Yang, Cewu Lu, and Bo Dai. Semgrasp: Semantic grasp generation via language aligned discretization. In European Conference on Computer Vision, pages 109–127, 2024.

[34] Zhuoling Li, Hossein Rahmani, Qiuhong Ke, and Jun Liu. LongDiff: Training-free long video generation in one go. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 17789–17798, 2025.

[35] Zhuoling Li, Hossein Rahmani, Jiarui Zhang, Yu Xue, Majid Mirmehdi, Jason Kuen, Jiuxiang Gu, and Jun Liu. DiffGraph: An automated agent-driven model merging framework for in-the-wild text-to-image generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 36713–36723, 2026.

[36] Yu Xue, Haoxuan Qu, Zhuoling Li, Hongbin Xu, Jianxiong Yin, Simon See, Hossein Rahmani, and Jun Liu. HRDiT: Training-free high-resolution image generation with off-the-shelf diffusion transformer models. In European Conference on Computer Vision, pages 536–554. Springer, 2026.

[37] D Subbaram Naidu. Optimal control systems. CRC press, 2018.

[38] Carles Domingo-Enrich, Michal Drozdzal, Brian Karrer, and Ricky TQ Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. In The Thirteenth International Conference on Learning Representations, 2025.

[39] Ugo Rosolia, Ashwin Carvalho, and Francesco Borrelli. Autonomous racing using learning model predictive control. In 2017 American control conference (ACC), pages 5115–5120. IEEE, 2017.

[40] C Dario Bellicoso, Fabian Jenelten, Christian Gehring, and Marco Hutter. Dynamic locomotion through online nonlinear motion optimization for quadrupedal robots. IEEE Robotics and Automation Letters, 3(3):2261–2268, 2018.

[41] Alexander Liniger, Alexander Domahidi, and Manfred Morari. Optimization-based autonomous racing of 1: 43 scale rc cars. Optimal Control Applications and Methods, 36(5):628–647, 2015.

[42] Lander Vanroye, Ajay Sathya, Joris De Schutter, and Wilm Decré. Fatrop: A fast constrained optimal control problem solver for robot trajectory optimization and control. In 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 10036–10043. IEEE, 2023.

[43] Hyungjin Chung, Jeongsol Kim, Michael Thompson Mccann, Marc Louis Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. In The Eleventh International Conference on Learning Representations, 2023.

[44] Jiaming Song, Qinsheng Zhang, Hongxu Yin, Morteza Mardani, Ming-Yu Liu, Jan Kautz, Yongxin Chen, and Arash Vahdat. Loss-guided diffusion models for plug-and-play controllable generation. In International Conference on Machine Learning, pages 32483–32498. PMLR, 2023.

[45] Christian P Robert and George Casella. Monte Carlo statistical methods. Springer, second edition, 2004.

[46] Zhenyu Wei, Zhixuan Xu, Jingxiang Guo, Yiwen Hou, Chongkai Gao, Zhehao Cai, Jiayu Luo, and Lin Shao. D (r, o) grasp: A unified representation of robot and object interaction for cross-embodiment dexterous grasping. In CoRL Workshop on Learning Robot Fine and Dexterous Manipulation: Perception and Control, 2024.

[47] Yifei Shen, Xinyang Jiang, Yifan Yang, Yezhen Wang, Dongqi Han, and Dongsheng Li. Understanding and improving training-free loss-based diffusion guidance. Advances in Neural Information Processing Systems, 37:108974–109002, 2024.

[48] Bradley Efron. Tweedie's formula and selection bias. Journal of the American Statistical Association, 106(496):1602–1614, 2011.

[49] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. Advances in neural information processing systems, 28, 2015.

[50] Hanwen Jiang, Shaowei Liu, Jiashun Wang, and Xiaolong Wang. Hand-object contact consistency reasoning for human grasps generation. In Proceedings of the IEEE/CVF international conference on computer vision, pages 11107–11116, 2021.

[51] Guo-Hao Xu, Yi-Lin Wei, Dian Zheng, Xiao-Ming Wu, and Wei-Shi Zheng. Dexterous grasp transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 17933–17942, 2024.

[52] James C Spall. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. IEEE transactions on automatic control, 37(3):332–341, 1992.

[53] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine learning, 8(3):229–256, 1992.

[54] Tim Salimans, Jonathan Ho, Xi Chen, Szymon Sidor, and Ilya Sutskever. Evolution strategies as a scalable alternative to reinforcement learning. arXiv preprint arXiv:1703.03864, 2017.

[55] Michael Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. Journal of Machine Learning Research, 26(209):1–80, 2025.

[56] Michael S Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. arXiv preprint arXiv:2209.15571, 2022.

[57] Qiuchi Xiang, Jintao Cheng, Jiehao Luo, Jin Wu, Rui Fan, Xieyuanli Chen, and Xiaoyu Tang. Overlapmamba: Novel shift state space model for lidar-based place recognition. arXiv preprint arXiv:2405.07966, 2024.

[58] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In Proceedings of the IEEE/CVF international conference on computer vision, pages 4015–4026, 2023.

[59] Bernt Oksendal. Stochastic differential equations: an introduction with applications. Springer Science & Business Media, 2013.

[60] Alexander Robey, Zachary Ravichandran, Vijay Kumar, Hamed Hassani, and George J Pappas. Jailbreaking llm-controlled robots. In 2025 IEEE International Conference on Robotics and Automation (ICRA), pages 11948–11956. IEEE, 2025.

[61] Qiuchi Xiang, Haoxuan Qu, Hossein Rahmani, and Jun Liu. Is monitoring enough? strategic agent selection for stealthy attack in multi-agent discussions. In European Conference on Computer Vision, pages 416–434. Springer, 2026.

[62] Qiuchi Xiang, Haoxuan Qu, Hossein Rahmani, and Jun Liu. New wide-net-casting jailbreak attacks risk large models. arXiv preprint arXiv:2605.17128, 2026.

[63] Zhuoling Li, Haoxuan Qu, Jason Kuen, Jiuxiang Gu, Qiuhong Ke, Jun Liu, and Hossein Rahmani. DiffIP: Representation fingerprints for robust IP protection of diffusion models. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 17035–17045, 2025.

A Further Analysis and Additional Ablation Studies

Below, we conduct further analysis and additional ablation studies on the standard train-test split of DexGYSNet with training following DexGYS [3], unless otherwise specified.

Additional analysis on the gradient-free optimal control law. In the main paper, we propose the NDPP-Grasp framework, which derives a gradient-free optimal control law to enforce non-differentiable physical plausibility constraints during grasp denoising. To evaluate this design, we have compared NDPP-Grasp against four variants in Tab. 3 of the main paper. Below, we provide more details about these variants. **Variant A** introduces differentiable surrogate losses for non-differentiable physical plausibility constraints, and incorporates these surrogate losses into the diffusion training objective alongside task alignment. For example, for the hand-object penetration constraint, we adopt the penetration loss in [3], which approximates the non-differentiable penetration constraint via the signed distance field of the object. At inference time, this variant performs standard denoising without any denoising-time guidance. **Variant B** uses the same surrogate losses as Variant A, but applies them at inference time rather than during training. Specifically, it adopts existing denoising-time guidance techniques [44] that leverage the gradients of these surrogate losses to compute the guidance term at each denoising step. **Variant C** approximates the gradients of the non-differentiable constraint evaluators via zeroth-order finite-difference approximation [52]. At each denoising step, it samples random perturbation vectors, queries the constraint evaluator on the corresponding terminal grasps, and aggregates the evaluations into a finite-difference gradient estimate that is used as the guidance term. **Variant D** applies reinforcement learning [53] to derive a guidance term from the non-differentiable constraint evaluator. At each denoising step, it draws noise increments, treats the negative physical plausibility cost as a reward, and updates the guidance term along the estimated policy-gradient direction.

Moreover, we further extend this analysis with three additional variants that all operate during the denoising process, and report the results in Tab. 5. **Variant E** adopts the evolution strategy [54] to derive a guidance term from the non-differentiable constraint evaluator. At each denoising step, it draws a population of noise increments, evaluates each on the corresponding terminal grasp, and aggregates the evaluations into a population-level finite-difference estimate as the guidance term. **Variant F** trains a separate binary classifier for each non-differentiable physical plausibility constraint to predict its satisfaction on a grasp, and uses the gradient of each classifier as the guidance term during denoising. Note that, unlike NDPP-Grasp, this variant requires additional classifier training. **Variant G** draws M candidate noise increments at each denoising step, evaluates each on the corresponding terminal grasp, and directly selects the candidate that yields the best evaluation as the noise increment of this step, without computing any guidance term. As shown in Tab. 5, NDPP-Grasp consistently outperforms all variants, further demonstrating the effectiveness of our gradient-free optimal control law for leveraging non-differentiable physical plausibility constraints during grasp denoising.

Table 5: Evaluation of our derived gradient-free optimal control law.

Method	Physical Qual.			Task Align.			Diversity		
	Succ. ↑	Q_1 ↑	Pen. ↓	FID ↓	CD ↓	Con. ↓	δ_t ↑	δ_r ↑	δ_q ↑
Variant A	54.27%	0.071	0.378	12.64	1.987	0.061	4.213	41.28	4.317
Variant B	62.91%	0.084	0.211	6.583	1.192	0.036	6.094	55.21	6.092
Variant C	60.74%	0.079	0.238	6.728	1.203	0.038	6.024	54.41	6.038
Variant D	59.86%	0.077	0.247	6.793	1.214	0.039	5.978	53.97	5.991
Variant E	61.32%	0.080	0.231	6.706	1.213	0.038	6.057	54.83	6.061
Variant F	62.45%	0.082	0.221	6.687	1.205	0.037	6.072	54.96	6.078
Variant G	64.74%	0.085	0.208	6.642	1.198	0.038	6.076	53.42	5.892
Ours	**67.05%**	**0.088**	**0.182**	**6.487**	**1.184**	**0.035**	**6.132**	**55.74**	**6.127**

Impact of the lookahead horizon H. In our framework, the lookahead horizon H controls how many future denoising steps the amortized lookahead strategy uses to predict the terminal grasp via Tweedie's formula when estimating the optimal control law (see Sec. 4.2). We set H to 20, and here further examine the effect of different choices of H, ranging from 1 to 50. The results are summarized in Tab. 6. As shown, the performance of our framework improves notably as H increases from 1 to 20, while further increasing H from 20 to 50 yields saturated gains. We therefore set $H = 20$ in our framework.

Table 6: Evaluation of the lookahead horizon H.

Method	Physical Qual.			Task Align.			Diversity		
	Succ. ↑	Q_1 ↑	Pen. ↓	FID ↓	CD ↓	Con. ↓	δ_t ↑	δ_r ↑	δ_q ↑
$H = 1$	61.92%	0.080	0.218	6.892	1.247	0.040	5.974	54.21	5.967
$H = 10$	65.42%	0.086	0.197	6.642	1.213	0.037	6.072	55.04	6.068
$H = 20$	67.05%	0.088	0.182	6.487	1.184	0.035	6.132	55.74	6.127
$H = 30$	67.06%	0.088	0.182	6.486	1.184	0.035	6.132	55.74	6.127
$H = 50$	67.07%	0.089	0.183	6.485	1.184	0.035	6.133	55.74	6.127

Impact of non-differentiable physical plausibility constraints. In the main paper, we evaluate NDPP-Grasp with three non-differentiable physical plausibility constraints. To further verify the effectiveness of NDPP-Grasp, we consider three additional non-differentiable physical plausibility constraints and conduct experiments with different numbers of active constraints K. The three additional constraints are as follows. The **contact sufficiency** constraint encourages a sufficient number of fingertips to contact the object, with a smoothed contact-distance objective as its differentiable surrogate. The **force-closure** constraint encourages the generated grasp to satisfy force closure, with a continuous force-closure energy as its differentiable surrogate. The **simulation-based stability** constraint encourages the generated grasp to remain stable under physical perturbations, with a contact-aware stability objective [5] as its differentiable surrogate. In the first setting, we consider $K = 4$ constraints, comprising the three from the main paper plus the contact sufficiency constraint. In the second setting, we consider $K = 5$ constraints, further adding the force-closure constraint. In the third setting, we consider $K = 6$ constraints, further adding the simulation-based stability constraint. As shown in Tab. 7, NDPP-Grasp consistently improves over the DexGYS backbone across different numbers of active constraints, demonstrating its effectiveness under varying numbers of non-differentiable constraints.

Table 7: Evaluation of NDPP-Grasp under different numbers of non-differentiable constraints.

Number of constraints	Method	Physical Qual.			Task Align.			Diversity		
		Succ. ↑	Q_1 ↑	Pen. ↓	FID ↓	CD ↓	Con. ↓	δ_t ↑	δ_r ↑	δ_q ↑
$K = 3$	DexGYS [3]	63.31%	0.083	0.223	6.538	1.198	0.036	6.118	55.68	6.118
	Ours (training following [3])	67.05%	0.088	0.182	6.487	1.184	0.035	6.132	55.74	6.127
$K = 4$	DexGYS [3]	63.28%	0.083	0.225	6.541	1.200	0.037	6.117	55.67	6.117
	Ours (training following [3])	67.42%	0.088	0.179	6.485	1.184	0.035	6.134	55.75	6.128
$K = 5$	DexGYS [3]	63.25%	0.082	0.227	6.544	1.202	0.037	6.116	55.66	6.116
	Ours (training following [3])	67.71%	0.089	0.176	6.483	1.183	0.035	6.136	55.76	6.129
$K = 6$	DexGYS [3]	63.26%	0.082	0.230	6.547	1.204	0.038	6.115	55.65	6.115
	Ours (training following [3])	67.96%	0.089	0.174	6.481	1.183	0.034	6.137	55.77	6.130

Additional analysis on flow matching. In the main paper, we develop NDPP-Grasp under the diffusion formulation. To further validate the broader applicability of NDPP-Grasp to other diffusion-related generative paradigms, we adapt it to AffordDexGrasp [2], a flow-matching-based dexterous grasp generation method that generates grasps by deterministic ODE integration with a learned velocity field $v_\theta(\mathbf{x}_t, t \mid c)$. Our adaptation proceeds as follows. **(1)** We construct a stochastic counterpart of the flow-matching ODE in the form $d\mathbf{x}_t = \tilde{v}_\theta(\mathbf{x}_t, t \mid c)\, dt + \sigma(t)\, d\mathbf{w}_t$ with a corrected drift $\tilde{v}_\theta = v_\theta + \frac{\sigma^2(t)}{2} s_\theta$, where $\sigma(t)$ is a noise schedule and s_θ is the score function, following the connection between deterministic and stochastic flow-matching sampling established in stochastic interpolant theory [55, 56]. In the ideal continuous-time limit with exact velocity and score fields, this stochastic counterpart preserves the same marginal path as the original flow-matching ODE. **(2)** Under the linear-interpolant convention, we obtain a score estimator from the learned velocity field as $s_\theta(\mathbf{x}_t, t \mid c) = -(\mathbf{x}_t - t \cdot v_\theta)/(1 - t)$, which avoids retraining the velocity model. **(3)** We apply the gradient-free optimal control law and the amortized lookahead strategy of NDPP-Grasp on the resulting stochastic dynamics, with the terminal grasp predicted via the flow-matching endpoint estimate $\hat{\mathbf{x}}_1 = \mathbf{x}_t + (1 - t)v_\theta$. As shown in Tab. 8, NDPP-Grasp consistently improves the physical plausibility of the generated grasps over the AffordDexGrasp baseline while maintaining the task alignment and diversity, demonstrating the broader applicability of NDPP-Grasp to flow-matching-based grasp generation.

Impact of the sampling number M. In our framework, the sampling number M specifies how many uncontrolled trajectories the amortized lookahead strategy uses for Monte Carlo estimation of the optimal control law (see Sec. 4.2). We set M to 8, and here further examine the effect of different choices of M, ranging from 1 to 32. The results are summarized in Tab. 9. As shown, the

Table 8: Evaluation of NDPP-Grasp on the flow-matching-based AffordDexGrasp.

Method	Physical Qual.			Task Align.			Diversity			Time (ms) ↓
	Succ. ↑	Q_1 ↑	Pen. ↓	FID ↓	CD ↓	Con. ↓	δ_t ↑	δ_r ↑	δ_q ↑	
AffordDexGrasp [2]	62.81%	0.082	0.228	6.538	1.198	0.039	5.924	53.86	5.918	493.1
Ours (training following [2])	**65.34%**	**0.087**	**0.201**	**6.502**	**1.189**	**0.037**	**5.951**	**54.07**	**5.943**	**28.4**

performance of our framework improves notably as M increases from 1 to 8, while further increasing M from 8 to 32 yields saturated gains with inference time continuing to grow. We therefore set $M = 8$ in our framework to balance performance and computational cost.

Table 9: Evaluation of the sampling number M.

Method	Physical Qual.			Task Align.			Diversity			Time (ms) ↓
	Succ. ↑	Q_1 ↑	Pen. ↓	FID ↓	CD ↓	Con. ↓	δ_t ↑	δ_r ↑	δ_q ↑	
$M = 1$	63.71%	0.083	0.220	6.948	1.262	0.041	5.918	53.87	5.912	8.9
$M = 4$	65.18%	0.086	0.198	6.658	1.217	0.037	6.064	55.02	6.058	10.1
$M = 8$	67.05%	0.088	0.182	6.487	1.184	0.035	6.132	55.74	6.127	11.6
$M = 16$	67.05%	0.088	0.182	6.486	1.184	0.035	6.132	55.74	6.127	14.7
$M = 32$	67.07%	0.089	0.182	6.485	1.184	0.035	6.133	55.74	6.127	21.0

Additional analysis on task-agnostic dexterous grasp generation. In this paper, we propose the NDPP-Grasp framework for task-oriented dexterous grasp generation. Given that diverse diffusion-based methods also exist for task-agnostic dexterous grasp generation, here we further validate the broader applicability of NDPP-Grasp by integrating it into two representative task-agnostic methods, SceneDiffuser [31] and DexDiffuser [29]. We evaluate the integrated methods on the DexGraspNet [5] dataset, following the standard task-agnostic dexterous grasping train-test split of [31]. Specifically, we report the Success Rate under all six axis-aligned external force directions, the Success Rate under at least one direction, the Maximum Penetration Depth between the hand and the object, and the Diversity measured as the mean standard deviation of pose parameters across successful grasps. As shown in Tab. 10, NDPP-Grasp consistently improves the physical plausibility of the generated grasps across both task-agnostic methods while maintaining the diversity, demonstrating the broader applicability of NDPP-Grasp.

Table 10: Evaluation on task-agnostic dexterous grasp generation methods.

Method	Success Rate (all directions) ↑	Success Rate (any direction) ↑	Maximum Penetration (mm) ↓	Diversity ↑
SceneDiffuser [31]	26.6	66.9	31.0	0.15
Ours (training following [31])	**31.4**	**70.1**	**22.7**	**0.16**
DexDiffuser [29]	38.2	75.4	27.8	0.13
Ours (training following [29])	**41.7**	**79.0**	**20.4**	**0.14**

Impact of H in H-step trajectory resampling. In the main paper, we discussed that the **H-step trajectory resampling** (i.e., the natural finite-lookahead instantiation [44, 47, 48]) suffers from a key tension between effectiveness and practicality. To verify the effectiveness of our method, we conduct experiments comparing our method against H-step trajectory resampling with different choices of H from 1 to 50. As shown in Tab. 11, our method achieves comparable or better physical-plausibility quality than this method at all H while consistently requiring substantially low inference time, demonstrating the effectiveness of our method which can resolve the tension.

Impact of the correction weight. In the main paper, we noted that directly reusing stored future states across denoising steps would correspond to using an incorrect future trajectory distribution, and we derived a closed-form correction weight $\rho_{\tau+1}^{(m)}$ to address this mismatch. To verify the effectiveness of this correction, we test a variant **Reuse without correction** that sets $\rho_{\tau+1}^{(m)} = 1$ to directly reuse stored future states without correction. As shown in Tab. 12, our method outperforms this variant, demonstrating the effectiveness of the correction weight for valid reuse of stored trajectories across denoising steps in our method.

User study. To further evaluate NDPP-Grasp from a human perspective, we conduct a user study comparing NDPP-Grasp against the DexGYS. For each method, we generate 100 grasps, resulting in a total of 200 grasps for evaluation. We then recruit 10 participants for the user study. For each grasp,

Table 11: Evaluation of H in H-step trajectory resampling.

Method	Physical Qual.			Task Align.			Diversity			Time
	Succ. ↑	Q_1 ↑	Pen. ↓	FID ↓	CD ↓	Con. ↓	δ_t ↑	δ_r ↑	δ_q ↑	(ms) ↓
H-step trajectory resampling ($H = 1$)	61.92%	0.080	0.218	6.892	1.247	0.040	5.974	54.21	5.967	10.7
H-step trajectory resampling ($H = 10$)	65.42%	0.086	0.197	6.642	1.213	0.037	6.072	55.04	6.068	58.5
H-step trajectory resampling ($H = 20$)	67.05%	0.088	0.182	6.488	1.184	0.035	6.131	55.73	6.126	109.8
H-step trajectory resampling ($H = 30$)	67.05%	0.088	0.182	6.486	1.184	0.035	6.132	55.74	6.127	164.7
H-step trajectory resampling ($H = 50$)	67.06%	0.089	0.183	6.485	1.184	0.035	6.133	55.74	6.127	274.5
Ours	67.05%	0.088	0.182	6.487	1.184	0.035	6.132	55.74	6.127	11.6

Table 12: Evaluation of the correction weight.

Method	Physical Qual.			Task Align.			Diversity		
	Succ. ↑	Q_1 ↑	Pen. ↓	FID ↓	CD ↓	Con. ↓	δ_t ↑	δ_r ↑	δ_q ↑
Reuse without correction	51.43%	0.064	0.413	9.187	1.582	0.058	5.421	51.23	5.402
Ours	67.05%	0.088	0.182	6.487	1.184	0.035	6.132	55.74	6.127

we render it as an interactive 3D visualization in HTML, allowing participants to freely rotate and zoom in their browser to inspect the grasp from arbitrary viewpoints. Participants are presented with the rendered grasps from both methods in random order to eliminate potential bias, together with the corresponding task instruction. Participants are then asked to score each grasp on a scale of 1 to 5 according to two evaluation criteria, including the physical plausibility and the task alignment. The average scores for each method are reported in Tab. 13. As shown, NDPP-Grasp consistently outperforms the DexGYS on both criteria, demonstrating its effectiveness from a human perspective.

Table 13: User study comparing NDPP-Grasp against the DexGYS baseline.

Method	Physical Plausibility ↑	Task Alignment ↑
DexGYS [3]	3.2	3.8
Ours (training following [3])	**4.2**	**4.0**

B Additional Qualitative Results

In the main paper, we provide qualitative comparisons between our method and DexGYS [3] in Fig. 2. Here, we further provide qualitative comparisons between our method and two additional existing methods, DexTOG [4] and Text2Grasp [13], in Fig. 3. As shown, our method consistently produces grasps that better satisfy physical plausibility constraints while preserving task alignment and grasp diversity, while both existing methods often produce grasps with physical infeasibilities.

C Real-world Experiments

Experimental setup.

Geometric perception provides spatial information for robotic decision-making [3, 57]. We follow the real-world experimental setup of [3] and adopt their DexGYS model as our backbone, using a six-degree-of-freedom robotic arm equipped with a dexterous hand and an RGB-D camera for visual perception. Specifically, the dexterous hand we use is the OmniHand Lingdong 2025, an anthropomorphic five-finger hand with 16 degrees of freedom (4 for the thumb and 3 for each of the remaining four fingers), of which 10 are actively driven. We collect several daily life objects, including both in-distribution objects from the DexGYSNet dataset and out-of-distribution objects. For each object-task pair, we conduct 10 grasping trials and report the success rate, where a trial is counted as successful if the dexterous hand stably grasps the object in a task-aligned configuration.

Experiment Pipeline. Following [3], we first obtain the partial point cloud of the target object via visual grounding and SAM [58], and then complete it into a full point cloud through a point cloud completion network as the input to our framework. We deploy the ShadowHand-parameterized output via a pre-defined kinematic mapping from ShadowHand joints to the OmniHand's 10 actuated DoFs, while inactive joints follow their default coupling. During execution, we first move the robotic arm to the pose of the predicted dexterous hand root node, and then drive the joints of the dexterous hand to the predicted pose to complete the grasp.

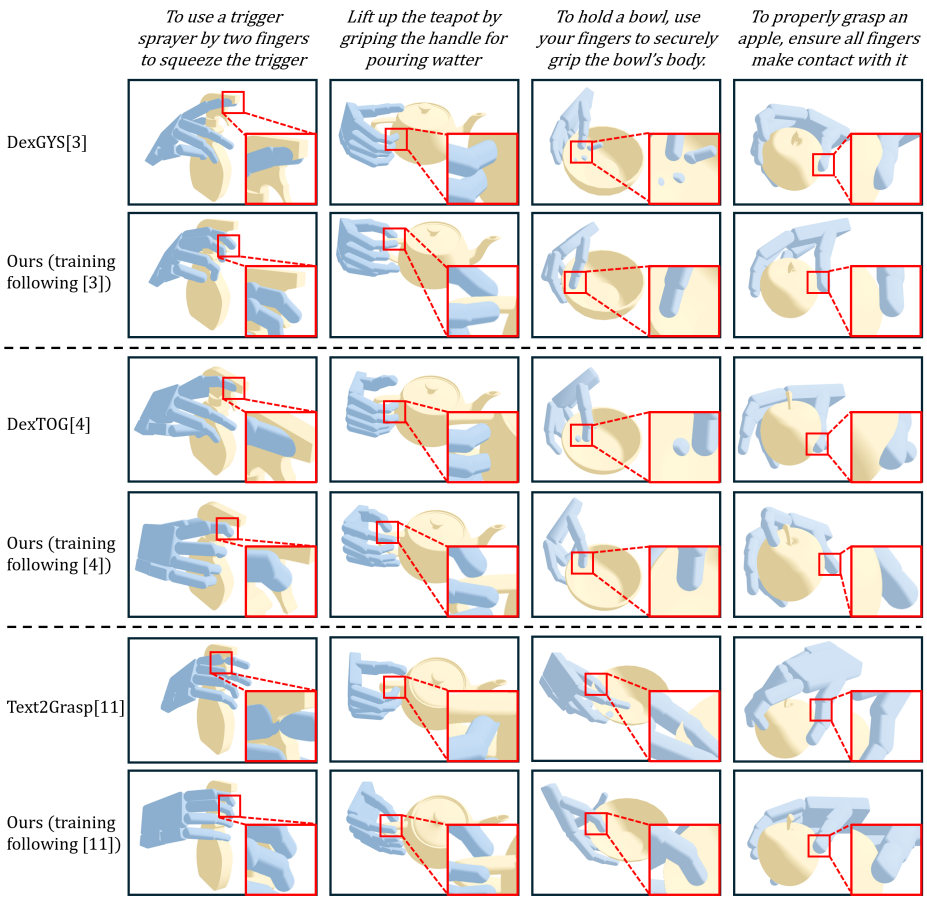

Figure 3: Additional qualitative comparisons between our method and two additional baselines, DexTOG and Text2Grasp, on DexGYSNet. Key contact regions are zoomed in red boxes.

Table 14: Real-world grasping success rates of our method across six daily life objects.

Objects	Lotion pump	Cup	Apple	Banana	Trigger sprayer	Mug
Success rate	7/10	6/10	9/10	8/10	6/10	8/10

Experimental results. As shown in Fig. 4, our method enables the dexterous hand to perform task-oriented grasping in a task-aligned and physically plausible manner across various objects. Quantitatively, we report per-object success rates in Tab. 14. Our method achieves an average success rate of 73.3% across all object-task pairs, demonstrating the effectiveness of our method in real-world deployment.

D More Details about Experiments

Datasets. We evaluate NDPP-Grasp on two widely adopted dexterous grasping benchmarks: DexGYSNet [3] and DexTOG-80K [4]. DexGYSNet is a language-guided dexterous grasping dataset comprising 50,000 ShadowHand grasps on 1,800 common household objects, where each grasp is paired with fine-grained natural language guidance describing how the hand interacts with the object. DexTOG-80K is a task-oriented dexterous grasping dataset comprising 80,000 ShadowHand grasps on 80 objects across five manipulation tasks (stapler clicking, spray bottle pressing, spray bottle

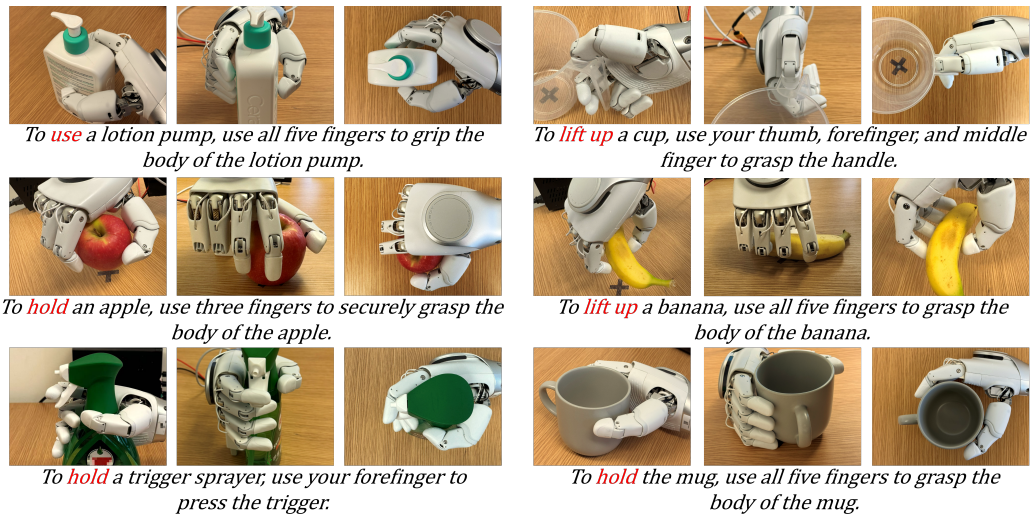

Figure 4: Real-world grasping results of NDPP-Grasp on a variety of daily life objects, including objects from both in-distribution and out-of-distribution categories with respect to DexGYSNet. Across all objects, our method drives the dexterous hand to perform task-aligned and physically plausible task-oriented grasps in the real world.

triggering, bottle cap twisting, and ballpoint pen pressing), where each task is described in natural language.

The open-set evaluation on DexGYSNet. To more comprehensively evaluate the generalization ability of NDPP-Grasp beyond the standard train-test split, we follow the open-set split strategy of [2], originally introduced for their open-set table-top dexterous grasping benchmark, and apply it to DexGYSNet. Specifically, we first partition all object categories in DexGYSNet into seen and unseen categories. Then, 80% of the objects from the seen categories are used for training, while the remaining 20% of the seen-category objects together with all objects from the unseen categories form the test set. This setup jointly evaluates two complementary aspects of generalization: in-category generalization to unseen instances within seen categories, and cross-category generalization to entirely unseen categories.

Metrics. We provide more details on the metrics used to evaluate NDPP-Grasp in our main paper, organized by the three aspects of physical quality, task alignment, and diversity. For physical quality, the **success rate (Succ.)** on DexGYSNet is evaluated in the Isaac Gym simulator, where each generated grasp is subjected to gravity from six different directions, and a grasp is counted as successful if it can withstand at least one direction [3]. The **overall success rate (Overall-Succ.)** on DexTOG-80K is evaluated by executing each generated grasp with a learned task-execution policy in simulation; a grasp is counted as successful only if the policy completes the corresponding manipulation task with the grasp [4]. The Q_1 metric measures the norm of the smallest external wrench that disrupts grasp stability, with the contact threshold set to 1 cm and the penetration threshold set to 5 mm; grasps with maximal penetration depth above 5 mm are deemed invalid and assigned $Q_1 = 0$. The **maximal penetration depth (Pen.)** reports the largest penetration distance from the object point cloud to the generated hand mesh in centimeters. The **collision-free rate** (η_f) on DexTOG-80K reports the proportion of generated grasps with no hand-object penetration. For task alignment on DexGYSNet, the **Fréchet Inception Distance (FID)** measures the distribution-level distance between generated and ground-truth grasps using point cloud features; the **Chamfer Distance (CD)** measures the chamfer distance between the generated and ground-truth hand point clouds; and the **Contact Distance (Con.)** measures the L_2 distance between the generated and ground-truth object-side contact maps. For diversity on DexGYSNet, δ_t, δ_r, and δ_q report the standard deviations of grasp translation, rotation (in Euler angles, in degrees), and joint angles across eight grasps generated under the same object and language guidance [3].

Implementation details of compared methods. In our main paper, following DexGYS [3] and DexTOG [4], we compare NDPP-Grasp against several task-agnostic dexterous grasp methods that do not natively support language conditioning. Below we provide more details on how these task-agnostic methods are adapted on each dataset. On DexGYSNet, following [3], we reimplement CVAE [49], GraspTTA [50], SceneDiffuser [31], and DGTR [51] using the same point cloud encoder, training loss, and CLIP language encoder as DexGYS [3]. The CLIP language feature is then concatenated with the object feature and the latent or noise input of each backbone to inject language conditioning, and GraspTTA further employs its test-time adaptation on top of the conditioned CVAE. On DexTOG-80K, following [4], we adapt GraspTTA [50] from its original CVAE-based grasp-type control to language-conditioned generation by replacing the CVAE latent with the text embedding of the task description, while keeping the rest of its training setting unchanged. The remaining compared methods (AffordDexGrasp [2], DexTOG [4], Text2Grasp [13], and DexGYS [3]) natively support language conditioning, and we train them on each dataset following the details in their original paper.

Details of the physical plausibility constraints. In our main paper, following [3, 2], we restore the original non-differentiable physical plausibility constraints from the differentiable surrogates used in existing methods, including hand-object non-penetration, self-collision avoidance, and joint-limit constraints. Below we provide more details on each. The **hand-object non-penetration constraint** requires that no point on the hand mesh lies inside the object volume, evaluated via a hard mesh-based intersection check between the hand mesh and the object mesh, in contrast to the smooth signed-distance-based penetration losses adopted as differentiable surrogates in existing methods [3, 2]. The **self-collision avoidance constraint** requires that no two distinct finger links of the dexterous hand intersect with each other, evaluated via pairwise collision checks across all finger-link pairs. The **joint-limit constraint** requires that every joint angle of the dexterous hand lies within its mechanically valid range, evaluated via per-joint range checks against the joint-limit specification of the dexterous hand model. In addition to the above non-differentiable physical plausibility constraints, following AffordDexGrasp [2], we additionally consider two differentiable constraints based on our contact-map representation: the contact attraction loss, which pulls hand contact candidates toward the contact regions on the object surface, and the contact fingertip loss, which encourages the fingertips to remain within the contact regions. The contact map is computed at each denoising step from the current grasp configuration and object geometry alone.

Implementation details of NDPP-Grasp. To obtain the predicted terminal grasp $\hat{\mathbf{g}}_T^{(m)}$ at the lookahead endpoint, we apply Tweedie's formula [48] on each backbone's denoising output, with the parameterization-specific recovery formulas in Eq. 58 and Eq. 59. The mapping Φ that converts the grasp pose $\mathcal{G}$ to the Euclidean diffusion representation $\mathbf{y}$ is implemented per backbone following each backbone's grasp parameterization, with Euler angles for the wrist rotation r, Euclidean coordinates for the wrist translation t scaled to $[-1, 1]$, and ShadowHand joint angles for q normalized via the corresponding joint-limit ranges. We handle differentiable constraints following the standard procedure in [44]. For the per-grasp inference time reported in Tab. 1 and Tab. 2 of the main paper, we measure the average inference time over the entire test set, and repeat the measurement three times to report the mean.

E Theoretical Analysis about NDPP-Grasp

E.1 Derivation of the control-affine stochastic dynamical system in Eq. 4 of the main paper

To derive the control-affine system in Eq. 4 of the main paper, we start from the guided reverse-time SDE in Eq. 3:

$$\mathrm{d}\mathbf{y}_s = -\beta(s)\Big[\tfrac{1}{2}\mathbf{y}_s + \nabla_{\mathbf{y}_s}\log p_s(\mathbf{y}_s \mid \mathcal{O}, \mathcal{L}) + \boldsymbol{\eta}_s(\mathbf{y}_s, s)\Big]\,\mathrm{d}s + \sqrt{\beta(s)}\,\mathrm{d}\bar{\mathbf{w}}_s. \tag{8}$$

We introduce the forward-time variable $\tau = T - s$ and let $\mathbf{g}_\tau = \mathbf{y}_{T-\tau}$. Under this change of variable, the drift sign flips since $\mathrm{d}s = -\mathrm{d}\tau$, the reverse-time Wiener increment $\sqrt{\beta(s)}\,\mathrm{d}\bar{\mathbf{w}}_s$ becomes the forward-time increment $\sqrt{\beta(T-\tau)}\,\mathrm{d}\mathbf{w}_\tau$ in distribution, and substituting $s = T - \tau$ into all time-dependent quantities of Eq. 8 yields

$$\mathrm{d}\mathbf{g}_\tau = \beta(T-\tau)\Big[\tfrac{1}{2}\mathbf{g}_\tau + \nabla_{\mathbf{g}_\tau}\log p_{T-\tau}(\mathbf{g}_\tau \mid \mathcal{O}, \mathcal{L}) + \boldsymbol{\eta}_{T-\tau}(\mathbf{g}_\tau, T-\tau)\Big]\mathrm{d}\tau + \sqrt{\beta(T-\tau)}\,\mathrm{d}\mathbf{w}_\tau. \tag{9}$$

Recall the definitions $\mathbf{F}(\mathbf{g}_\tau, \tau) = \frac{1}{2}\beta(T-\tau)\mathbf{g}_\tau + \beta(T-\tau)\nabla_{\mathbf{g}_\tau} \log p_{T-\tau}(\mathbf{g}_\tau \mid \mathcal{O}, \mathcal{L})$, $\sigma(\tau) = \sqrt{\beta(T-\tau)}$, and $\mathbf{u}_\tau(\mathbf{g}_\tau, \tau) = \sqrt{\beta(T-\tau)}\,\boldsymbol{\eta}_{T-\tau}(\mathbf{g}_\tau, T-\tau)$ from Eq. 4 of the main paper. Regrouping Eq. 9 according to these definitions yields the control-affine stochastic dynamical system in Eq. 4 of the main paper:

$$\mathrm{d}\mathbf{g}_\tau = \mathbf{F}(\mathbf{g}_\tau, \tau)\,\mathrm{d}\tau + \sigma(\tau)\big(\mathbf{u}_\tau(\mathbf{g}_\tau, \tau)\,\mathrm{d}\tau + \mathrm{d}\mathbf{w}_\tau\big). \tag{10}$$

E.2 Derivation of the gradient-free optimal control law in Eq. 6 of the main paper

To derive the gradient-free optimal control law in Eq. 6 of the main paper, we start from the stochastic optimal control problem in Eq. 5 of the main paper, restated here as

$$\mathbf{u}^*_{\tau:T} = \arg\min_{\mathbf{u}_{\tau:T}} \mathbb{E}_{\mathcal{P}^u}\left[\int_\tau^T \tfrac{1}{2}\|\mathbf{u}_\nu\|^2\,\mathrm{d}\nu + f_c(\mathbf{g}_T)\,\middle|\,\mathbf{g}_\tau\right], \tag{11}$$

where $f_c(\mathbf{g}_T)$ is the terminal cost induced by the constraint evaluator that penalizes physical plausibility violation of the final grasp, and $\frac{1}{2}\|\mathbf{u}_\nu\|^2$ is the running cost that penalizes the magnitude of the control signal. Eq. 11 requires searching over the entire sequence of control signals from time τ to T, which is intractable to solve directly. To make the problem solvable, we first transform it into a problem of finding a function over the state space rather than over control trajectories.

Probabilistic representation via the desirability function. To this end, we proceed by first introducing the value function and formulating the partial differential equation (PDE) it satisfies, then linearizing this PDE via the Hopf-Cole transformation, and finally applying the Feynman-Kac formula to obtain the desired probabilistic representation of the optimal control law.

Step (1). To carry out this transformation, we introduce the value function, defined as the minimum expected future cost achievable from state $\mathbf{g}_\tau$ at time τ:

$$V(\mathbf{g}_\tau, \tau) := \inf_{\mathbf{u}_{\tau:T}} \mathbb{E}_{\mathcal{P}^u}\left[\int_\tau^T \tfrac{1}{2}\|\mathbf{u}_\nu\|^2\,\mathrm{d}\nu + f_c(\mathbf{g}_T)\,\middle|\,\mathbf{g}_\tau\right]. \tag{12}$$

By summarizing all future costs into a function of the current state alone, V enables the optimal control to be obtained pointwise at each state, rather than searched over the entire control trajectory. Specifically, by standard stochastic optimal control theory [24], once V is known, the optimal control at any state can be directly read off as the (negatively scaled) gradient of V:

$$\mathbf{u}^*_\tau(\mathbf{g}_\tau, \tau) = -\sigma(\tau)\,\nabla_{\mathbf{g}_\tau} V(\mathbf{g}_\tau, \tau). \tag{13}$$

Therefore, solving for $\mathbf{u}^*_\tau$ reduces to solving for V. To solve for V, we apply the dynamic programming principle, which states that V at time τ must be consistent with V at the next infinitesimal time step $\tau + \mathrm{d}\tau$ once the running cost incurred during this step is taken into account. Combining this consistency condition with the controlled dynamics in Eq. 4 of the main paper, V is shown to satisfy the stochastic Hamilton-Jacobi-Bellman (HJB) equation [24]:

$$\frac{\partial V}{\partial \tau} - \frac{1}{2}\sigma^2(\tau)(\nabla_{\mathbf{g}_\tau} V)^\top(\nabla_{\mathbf{g}_\tau} V) + (\nabla_{\mathbf{g}_\tau} V)^\top \mathbf{F}(\mathbf{g}_\tau, \tau) + \frac{1}{2}\sigma^2(\tau)\,\mathrm{Tr}(\nabla^2_{\mathbf{g}_\tau\mathbf{g}_\tau} V) = 0, \tag{14}$$

with the terminal condition $V(\mathbf{g}_T, T) = f_c(\mathbf{g}_T)$. For notational simplicity, we omit the explicit dependence on $(\mathbf{g}_\tau, \tau)$ in V in Eq. 14.

Step (2). However, Eq. 14 is non-linear in V, due to the quadratic term $-\frac{1}{2}\sigma^2(\tau)(\nabla_{\mathbf{g}_\tau} V)^\top(\nabla_{\mathbf{g}_\tau} V)$. This non-linearity makes Eq. 14 intractable to solve directly in our high-dimensional grasp pose space. To remove this non-linearity, we apply the Hopf-Cole transformation, which is designed to convert HJB-type equations into linear ones by changing the variable from V to its exponential counterpart. Specifically, we introduce the desirability function

$$\varphi(\mathbf{g}_\tau, \tau) := \exp(-V(\mathbf{g}_\tau, \tau)), \tag{15}$$

where higher values of φ correspond to lower expected future cost, indicating that the corresponding state is more desirable. Under this transformation, the derivatives of V are related to those of φ via the chain rule:

$$\nabla_{\mathbf{g}_\tau} V = -\frac{\nabla_{\mathbf{g}_\tau}\varphi}{\varphi}, \quad \nabla^2_{\mathbf{g}_\tau\mathbf{g}_\tau} V = \frac{(\nabla_{\mathbf{g}_\tau}\varphi)(\nabla_{\mathbf{g}_\tau}\varphi)^\top}{\varphi^2} - \frac{\nabla^2_{\mathbf{g}_\tau\mathbf{g}_\tau}\varphi}{\varphi}, \quad \frac{\partial V}{\partial \tau} = -\frac{1}{\varphi}\frac{\partial \varphi}{\partial \tau}. \tag{16}$$

Substituting Eq. 16 into Eq. 14, the previously problematic quadratic term cancels exactly with the corresponding part of the trace term, which is the design goal of the Hopf-Cole transformation. Multiplying the resulting equation by $-\varphi$, the non-linear HJB equation is converted into the linear PDE in φ:

$$-\frac{\partial \varphi(\mathbf{g}_\tau, \tau)}{\partial \tau} = \mathbf{F}(\mathbf{g}_\tau, \tau)^\top \nabla_{\mathbf{g}_\tau} \varphi(\mathbf{g}_\tau, \tau) + \frac{1}{2}\sigma^2(\tau)\,\mathrm{Tr}(\nabla^2_{\mathbf{g}_\tau \mathbf{g}_\tau} \varphi(\mathbf{g}_\tau, \tau)), \tag{17}$$

with the terminal condition $\varphi(\mathbf{g}_T, T) = \exp(-f_c(\mathbf{g}_T))$.

Step (3). Now that the equation in φ is linear, we can leverage classical results from stochastic analysis to obtain its solution. We observe that Eq. 17 is exactly the backward Kolmogorov equation associated with the uncontrolled denoising dynamics $\mathcal{P}^0$, defined as Eq. 4 of the main paper with $\mathbf{u}_\tau \equiv 0$, which describes how a function evolves backward in time under the uncontrolled process. By the Feynman-Kac formula [59], the solution to such an equation admits a probabilistic representation as the conditional expectation of the terminal condition under the uncontrolled dynamics:

$$\varphi(\mathbf{g}_\tau, \tau) = \mathbb{E}_{\mathcal{P}^0}\left[\exp(-f_c(\mathbf{g}_T)) \,\middle|\, \mathbf{g}_\tau\right]. \tag{18}$$

This representation gives the desirability function a clear interpretation: it measures how likely a trajectory starting from $\mathbf{g}_\tau$ leads to a constraint-satisfying grasp under the uncontrolled denoising dynamics. Combining Eq. 13 with $V = -\log\varphi$, the optimal control law is now expressed in terms of φ as

$$\mathbf{u}^*_\tau(\mathbf{g}_\tau, \tau) = \sigma(\tau)\,\nabla_{\mathbf{g}_\tau} \log \varphi(\mathbf{g}_\tau, \tau), \tag{19}$$

which replaces the abstract value function V with an explicit expectation under $\mathcal{P}^0$. However, evaluating Eq. 19 still requires computing the gradient $\nabla_{\mathbf{g}_\tau} \log\varphi$. The remaining task, which we accomplish below, is to convert this gradient into a form that depends only on forward evaluations of f_c along uncontrolled trajectories, without requiring any gradient computation on f_c itself.

Eliminating the gradient via Markov and Gaussian structures. The probabilistic representation derived above expresses the optimal control law via $\nabla_{\mathbf{g}_\tau} \log\varphi(\mathbf{g}_\tau, \tau)$, which still contains a gradient operator. Computing this gradient via the chain rule applied to the Feynman-Kac representation in Eq. 18 would require differentiating f_c, but as described in Sec. 4.1 of the main paper, f_c is a non-differentiable constraint evaluator and therefore admits no such gradient. Our key insight, therefore, is to eliminate the gradient operator at the level of $\nabla_{\mathbf{g}_\tau} \log\varphi$ itself, so that the optimal control law can be evaluated using only forward queries of f_c. Specifically, we proceed by first using the Markov property of $\mathcal{P}^0$ to transfer the gradient operator from the conditioning state to the next-step state, then exploiting the Gaussian structure of the Brownian increment via Stein's lemma to convert this gradient into a weighted expectation, and finally collapsing the resulting nested expectations into a single expectation over the entire uncontrolled trajectory.

Step (1). To begin, we transfer the gradient from the conditioning state $\mathbf{g}_\tau$ to the random next-step state $\mathbf{g}_{\tau+\mathrm{d}\tau}$. This transfer is essential because gradients with respect to a random variable can later be eliminated through its distributional structure, while gradients with respect to a fixed conditioning variable cannot. By the Markov property of $\mathcal{P}^0$ together with the tower rule of conditional expectations, φ at the current state can be rewritten as a one-step rollout to the next-step state under $\mathcal{P}^0$:

$$\varphi(\mathbf{g}_\tau, \tau) = \mathbb{E}_{\mathcal{P}^0}\left[\varphi(\mathbf{g}_{\tau+\mathrm{d}\tau}, \tau + \mathrm{d}\tau) \,\middle|\, \mathbf{g}_\tau\right], \tag{20}$$

where $\mathbf{g}_{\tau+\mathrm{d}\tau} = \mathbf{g}_\tau + \mathbf{F}(\mathbf{g}_\tau, \tau)\,\mathrm{d}\tau + \sigma(\tau)\,\mathrm{d}\mathbf{w}_\tau$ is obtained from one Euler-Maruyama step of $\mathcal{P}^0$ from $\mathbf{g}_\tau$. Taking the gradient on both sides of Eq. 20 with respect to $\mathbf{g}_\tau$, and applying the standard interchange of differentiation and expectation followed by the chain rule, we obtain

$$\nabla_{\mathbf{g}_\tau} \varphi(\mathbf{g}_\tau, \tau) = \mathbb{E}_{\mathcal{P}^0}\left[\nabla_{\mathbf{g}_{\tau+\mathrm{d}\tau}} \varphi(\mathbf{g}_{\tau+\mathrm{d}\tau}, \tau + \mathrm{d}\tau) \cdot \left(\mathbf{I} + \nabla_{\mathbf{g}_\tau} \mathbf{F}(\mathbf{g}_\tau, \tau)\,\mathrm{d}\tau\right) \,\middle|\, \mathbf{g}_\tau\right], \tag{21}$$

since $\partial \mathbf{g}_{\tau+\mathrm{d}\tau} / \partial \mathbf{g}_\tau = \mathbf{I} + \nabla_{\mathbf{g}_\tau} \mathbf{F}(\mathbf{g}_\tau, \tau)\,\mathrm{d}\tau$. The Jacobian term $\nabla_{\mathbf{g}_\tau} \mathbf{F}\,\mathrm{d}\tau$ is of order $O(\mathrm{d}\tau)$ and vanishes in the continuous-time limit, yielding

$$\nabla_{\mathbf{g}_\tau} \varphi(\mathbf{g}_\tau, \tau) = \mathbb{E}_{\mathcal{P}^0}\left[\nabla_{\mathbf{g}_{\tau+\mathrm{d}\tau}} \varphi(\mathbf{g}_{\tau+\mathrm{d}\tau}, \tau + \mathrm{d}\tau) \,\middle|\, \mathbf{g}_\tau\right]. \tag{22}$$

Through this step, the gradient operation has been transferred from the conditioning state to the random next-step state, setting the stage for its elimination in Step (2).

Step (2). Although Step (1) has transferred the gradient from $\mathbf{g}_\tau$ to $\mathbf{g}_{\tau+\mathrm{d}\tau}$, the gradient operator itself still remains. To eliminate this operator, we exploit the Gaussian structure of the Brownian increment

driving the next-step state. Specifically, we observe that $\mathbf{g}_{\tau+\mathrm{d}\tau}$ depends linearly on the Brownian increment $\mathrm{d}\mathbf{w}_\tau$, which can be standardized as $\mathrm{d}\mathbf{w}_\tau = \sqrt{\mathrm{d}\tau}\,\boldsymbol{\xi}$ with $\boldsymbol{\xi} \sim \mathcal{N}(0, \mathbf{I})$. This standardized form admits the application of Stein's lemma, which states that for any standard Gaussian variable $\boldsymbol{\xi} \sim \mathcal{N}(0, \mathbf{I})$ and any differentiable function $h(\boldsymbol{\xi})$ with bounded gradient,

$$\mathbb{E}[\boldsymbol{\xi}\, h(\boldsymbol{\xi})] = \mathbb{E}[\nabla_{\boldsymbol{\xi}} h(\boldsymbol{\xi})]\,. \tag{23}$$

Stein's lemma is exactly the tool we need: it converts the expectation of a gradient (right-hand side) into an expectation weighted by the Gaussian variable itself (left-hand side), thereby eliminating the gradient operator entirely.

To apply Stein's lemma, we set $h(\boldsymbol{\xi}) := \varphi(\mathbf{g}_{\tau+\mathrm{d}\tau}, \tau + \mathrm{d}\tau)$ where $\mathbf{g}_{\tau+\mathrm{d}\tau} = \mathbf{g}_\tau + \mathbf{F}(\mathbf{g}_\tau, \tau)\,\mathrm{d}\tau + \sigma(\tau)\sqrt{\mathrm{d}\tau}\,\boldsymbol{\xi}$. By the chain rule, $\nabla_{\boldsymbol{\xi}} h = \nabla_{\mathbf{g}_{\tau+\mathrm{d}\tau}} \varphi \cdot \sigma(\tau)\sqrt{\mathrm{d}\tau}$. Substituting into Eq. 23 and rearranging gives

$$\mathbb{E}_{\mathcal{P}^0}\left[\nabla_{\mathbf{g}_{\tau+\mathrm{d}\tau}} \varphi(\mathbf{g}_{\tau+\mathrm{d}\tau}, \tau + \mathrm{d}\tau) \,\middle|\, \mathbf{g}_\tau\right] = \frac{1}{\sigma(\tau)\sqrt{\mathrm{d}\tau}}\, \mathbb{E}_{\mathcal{P}^0}\left[\boldsymbol{\xi}\, \varphi(\mathbf{g}_{\tau+\mathrm{d}\tau}, \tau + \mathrm{d}\tau) \,\middle|\, \mathbf{g}_\tau\right]. \tag{24}$$

Combining Eq. 24 with Eq. 22 from Step (1) and recalling $\mathrm{d}\mathbf{w}_\tau = \sqrt{\mathrm{d}\tau}\,\boldsymbol{\xi}$, we obtain

$$\sigma(\tau)\, \nabla_{\mathbf{g}_\tau} \varphi(\mathbf{g}_\tau, \tau)\, \mathrm{d}\tau = \mathbb{E}_{\mathcal{P}^0}\left[\varphi(\mathbf{g}_{\tau+\mathrm{d}\tau}, \tau + \mathrm{d}\tau)\, \mathrm{d}\mathbf{w}_\tau \,\middle|\, \mathbf{g}_\tau\right]. \tag{25}$$

The gradient operator has now been completely eliminated: the right-hand side involves only an expectation of φ at the next-step state, weighted by the Brownian increment $\mathrm{d}\mathbf{w}_\tau$.

Step (3). Eq. 25 still contains the desirability function $\varphi(\mathbf{g}_{\tau+\mathrm{d}\tau}, \tau + \mathrm{d}\tau)$ at the next-step state, which is itself defined as an expectation under $\mathcal{P}^0$ and therefore cannot be directly evaluated by forward computation on f_c. To obtain a final expression that depends only on the terminal cost f_c, we expand this inner φ via the Feynman-Kac representation in Eq. 18, applied at the next-step state:

$$\varphi(\mathbf{g}_{\tau+\mathrm{d}\tau}, \tau + \mathrm{d}\tau) = \mathbb{E}_{\mathcal{P}^0}\left[\exp(-f_c(\mathbf{g}_T)) \,\middle|\, \mathbf{g}_{\tau+\mathrm{d}\tau}\right]. \tag{26}$$

Substituting Eq. 26 into Eq. 25, the right-hand side becomes a nested conditional expectation, in which the outer expectation is conditioned on $\mathbf{g}_\tau$ and the inner one is conditioned on $\mathbf{g}_{\tau+\mathrm{d}\tau}$. Since the Brownian increment $\mathrm{d}\mathbf{w}_\tau$ is determined by $(\mathbf{g}_\tau, \mathbf{g}_{\tau+\mathrm{d}\tau})$, it is measurable with respect to the inner conditioning and can therefore be absorbed into the inner expectation. The two nested conditional expectations can then be merged via the tower rule into a single expectation over the entire trajectory from $\mathbf{g}_\tau$ to $\mathbf{g}_T$ under $\mathcal{P}^0$:

$$\sigma(\tau)\, \nabla_{\mathbf{g}_\tau} \varphi(\mathbf{g}_\tau, \tau)\, \mathrm{d}\tau = \mathbb{E}_{\mathcal{P}^0}\left[\exp(-f_c(\mathbf{g}_T))\, \mathrm{d}\mathbf{w}_\tau \,\middle|\, \mathbf{g}_\tau\right]. \tag{27}$$

Combining Eq. 27 with $\mathbf{u}^*_\tau = \sigma(\tau)\, \nabla_{\mathbf{g}_\tau} \log \varphi = \sigma(\tau)\, \nabla_{\mathbf{g}_\tau} \varphi / \varphi$ from Eq. 19, where the denominator $\varphi(\mathbf{g}_\tau, \tau)$ takes the Feynman-Kac representation in Eq. 18, we arrive at the gradient-free optimal control law in Eq. 6 of the main paper:

$$\mathbf{u}^*_\tau(\mathbf{g}_\tau, \tau)\, \mathrm{d}\tau = \frac{\mathbb{E}_{\mathcal{P}^0}\left[\exp(-f_c(\mathbf{g}_T))\, \mathrm{d}\mathbf{w}_\tau \,\middle|\, \mathbf{g}_\tau\right]}{\mathbb{E}_{\mathcal{P}^0}\left[\exp(-f_c(\mathbf{g}_T)) \,\middle|\, \mathbf{g}_\tau\right]}. \tag{28}$$

The resulting expression involves only forward evaluations of f_c along uncontrolled trajectories under $\mathcal{P}^0$, with no gradient computation on f_c at any point.

E.3 Theoretical guarantee of posterior sampling under the optimal control law

In Sec. 4.1 of the main paper, we stated that the controlled denoising process under the optimal control law $\mathbf{u}^*_\tau$ samples its terminal distribution from the target posterior in Eq. 2 of the main paper. This section provides the formal proof of this guarantee. The proof is based on a path-integral representation that connects the transition probability of the controlled process under $\mathbf{u}^*_\tau$ to that of the uncontrolled process via the desirability function, combined with marginalization over the initial Gaussian distribution. We first state the main result as Theorem 1, then introduce the path-integral representation as Lemma 1, and finally give the proof.

Theorem 1 (Posterior sampling under the optimal control law) *Consider the controlled process in Eq. 4 of the main paper with initialization* $\mathbf{g}_0 \sim \mathcal{N}(0, \mathbf{I})$. *Let* $\mathcal{P}^0$ *denote the corresponding*

uncontrolled process obtained by setting $\mathbf{u}_\tau \equiv 0$*, whose terminal marginal coincides with the text-conditioned prior, i.e.,*

$$\mathcal{P}^0(\mathbf{g}_T) = p(\mathbf{g}_T \mid \mathcal{O}, \mathcal{L}). \tag{29}$$

For terminal cost $f_c(\mathbf{g}_T) = -\log p(c \mid \mathbf{g}_T) + \text{const}$*, the optimal control law* $\mathbf{u}^*_\tau$ *derived in Eq. 6 of the main paper induces a controlled process* $\mathcal{P}^{\mathbf{u}^*}$ *whose terminal marginal equals the target posterior in Eq. 2 of the main paper, i.e.,*

$$\mathcal{P}^{\mathbf{u}^*}(\mathbf{g}_T) = p(\mathbf{g}_T \mid \mathcal{O}, \mathcal{L}, c). \tag{30}$$

To prove Theorem 1, we first invoke a classical result from stochastic optimal control theory [24], which expresses the transition probability of the controlled process under $\mathbf{u}^*_\tau$ as the corresponding transition probability of the uncontrolled process modulated by the desirability function.

Lemma 1 (Path-integral representation) *For any two time points* $0 \le \tau_1 < \tau_2 \le T$*, the transition probability of the controlled process under the optimal control law* $\mathbf{u}^*_\tau$ *from* $\mathbf{g}_{\tau_1}$ *to* $\mathbf{g}_{\tau_2}$ *satisfies*

$$\mathcal{P}^{\mathbf{u}^*}_{\tau_1,\tau_2}(\mathbf{g}_{\tau_1}, \mathbf{g}_{\tau_2}) = \mathcal{P}^0_{\tau_1,\tau_2}(\mathbf{g}_{\tau_1}, \mathbf{g}_{\tau_2}) \cdot \frac{\varphi(\mathbf{g}_{\tau_2}, \tau_2)}{\varphi(\mathbf{g}_{\tau_1}, \tau_1)}, \tag{31}$$

where $\mathcal{P}^0_{\tau_1,\tau_2}$ *denotes the corresponding transition probability of the uncontrolled process, and* φ *is the desirability function defined in Eq. 15.*

The proof has four steps. We first rewrite the terminal cost in an analytically convenient form and compute the desirability function at the initial time. We next evaluate the desirability function at the terminal time. We then apply Lemma 1 to obtain the transition probability of the controlled process from $\tau = 0$ to $\tau = T$. We finally marginalize over the initial Gaussian distribution and apply Bayes' rule to recover the target posterior.

Step (1). We evaluate the desirability function at the initial time. The terminal cost $f_c(\mathbf{g}_T) = -\log p(c \mid \mathbf{g}_T) + \text{const}$ is unique only up to an additive constant, since adding a constant to f_c leaves the optimal control law $\mathbf{u}^*_\tau$ unchanged given that $\mathbf{u}^*_\tau = \sigma(\tau)\nabla_{\mathbf{g}_\tau} \log \varphi(\mathbf{g}_\tau, \tau)$ depends only on the gradient of $\log \varphi$, which is invariant under multiplying φ by a positive constant. We exploit this freedom to choose the additive constant so that the terminal cost takes the analytically convenient form

$$f_c(\mathbf{g}_T) = \log \frac{p_{\mathbf{g}}(\mathbf{g}_T)}{p_{\mathbf{g}}(\mathbf{g}_T \mid c)}, \tag{32}$$

where $p_{\mathbf{g}}(\mathbf{g}_T) := \mathcal{P}^0_{0,T}(\mathbf{g}, \mathbf{g}_T)$ is the terminal marginal of the uncontrolled process started from $\mathbf{g}_0 = \mathbf{g}$, and $p_{\mathbf{g}}(\mathbf{g}_T \mid c) := p(c \mid \mathbf{g}_T)p_{\mathbf{g}}(\mathbf{g}_T)/p(c)$ is the corresponding posterior at the terminal time. Here we use the fact that the initial pure noise $\mathbf{g}_0$ is independent of the constraint c, so $p(c \mid \mathbf{g}) = p(c)$. The equivalence of Eq. 32 with $-\log p(c \mid \mathbf{g}_T) + \text{const}$ follows from $\log \frac{p_{\mathbf{g}}(\mathbf{g}_T)}{p_{\mathbf{g}}(\mathbf{g}_T|c)} = \log \frac{p(c)}{p(c|\mathbf{g}_T)} = -\log p(c \mid \mathbf{g}_T) + \log p(c)$, where $\log p(c)$ is the chosen additive constant.

With this form of f_c, the desirability function at the initial time admits a closed-form value. By the Feynman-Kac representation in Eq. 18,

$$\varphi(\mathbf{g}, 0) = \mathbb{E}_{\mathcal{P}^0}[\exp(-f_c(\mathbf{g}_T)) \mid \mathbf{g}_0 = \mathbf{g}] = \int \frac{p_{\mathbf{g}}(\mathbf{g}_T \mid c)}{p_{\mathbf{g}}(\mathbf{g}_T)}\, p_{\mathbf{g}}(\mathbf{g}_T)\, \mathrm{d}\mathbf{g}_T = \int p_{\mathbf{g}}(\mathbf{g}_T \mid c)\, \mathrm{d}\mathbf{g}_T = 1, \tag{33}$$

where the last equality uses the fact that $p_{\mathbf{g}}(\cdot \mid c)$ is a normalized probability distribution.

Step (2). We evaluate the desirability function at the terminal time. By the boundary condition of the desirability function, $\varphi(\mathbf{g}_T, T) = \exp(-f_c(\mathbf{g}_T))$. Substituting Eq. 32,

$$\varphi(\mathbf{g}_T, T) = \frac{p_{\mathbf{g}}(\mathbf{g}_T \mid c)}{p_{\mathbf{g}}(\mathbf{g}_T)}. \tag{34}$$

Step (3). We now apply Lemma 1 with $\tau_1 = 0$ and $\tau_2 = T$. Substituting Eq. 33 and Eq. 34 into Eq. 31,

$$\mathcal{P}^{\mathbf{u}^*}_{0,T}(\mathbf{g}, \mathbf{g}_T) = \mathcal{P}^0_{0,T}(\mathbf{g}, \mathbf{g}_T) \cdot \frac{\varphi(\mathbf{g}_T, T)}{\varphi(\mathbf{g}, 0)} = p_{\mathbf{g}}(\mathbf{g}_T) \cdot \frac{p_{\mathbf{g}}(\mathbf{g}_T \mid c)/p_{\mathbf{g}}(\mathbf{g}_T)}{1} = p_{\mathbf{g}}(\mathbf{g}_T \mid c). \tag{35}$$

The transition probability of the controlled process from $\mathbf{g}_0 = \mathbf{g}$ to $\mathbf{g}_T$ is therefore exactly the posterior terminal distribution conditioned on the same initial state.

Step (4). To obtain the unconditional terminal marginal of the controlled process, we marginalize the transition probability $\mathcal{P}^{\mathbf{u}^*}_{0,T}$ over the initial Gaussian distribution $p_0(\mathbf{g}) = \mathcal{N}(0, \mathbf{I})$. Using Eq. 35 and substituting the definition $p_{\mathbf{g}}(\mathbf{g}_T \mid c) = p(c \mid \mathbf{g}_T) p_{\mathbf{g}}(\mathbf{g}_T)/p(c)$,

$$\mathcal{P}^{\mathbf{u}^*}(\mathbf{g}_T) = \int \mathcal{P}^{\mathbf{u}^*}_{0,T}(\mathbf{g}, \mathbf{g}_T)\, p_0(\mathbf{g})\, \mathrm{d}\mathbf{g} = \int p_{\mathbf{g}}(\mathbf{g}_T \mid c)\, p_0(\mathbf{g})\, \mathrm{d}\mathbf{g} = \frac{p(c \mid \mathbf{g}_T)}{p(c)} \int p_{\mathbf{g}}(\mathbf{g}_T)\, p_0(\mathbf{g})\, \mathrm{d}\mathbf{g}. \tag{36}$$

By the construction of the uncontrolled process, marginalizing the conditional terminal marginals against the initial distribution recovers the text-conditioned prior, i.e.,

$$\int p_{\mathbf{g}}(\mathbf{g}_T)\, p_0(\mathbf{g})\, \mathrm{d}\mathbf{g} = \mathcal{P}^0(\mathbf{g}_T) = p(\mathbf{g}_T \mid \mathcal{O}, \mathcal{L}), \tag{37}$$

which gives

$$\mathcal{P}^{\mathbf{u}^*}(\mathbf{g}_T) = \frac{p(c \mid \mathbf{g}_T)\, p(\mathbf{g}_T \mid \mathcal{O}, \mathcal{L})}{p(c)}. \tag{38}$$

By Bayes' rule, the right-hand side of Eq. 38 equals $p(\mathbf{g}_T \mid \mathcal{O}, \mathcal{L}, c)$, the target posterior in Eq. 2 of the main paper. This completes the proof.

E.4 Derivation of the finite-lookahead form of the optimal control law

The optimal control law in Eq. 28 requires expectations over full trajectories from $\mathbf{g}_\tau$ all the way to the terminal step $\mathbf{g}_T$. Simulating such full trajectories at every denoising step is prohibitively expensive in practice, and as discussed in Sec. 4.2 of the main paper, our framework instead uses an H-step lookahead with Tweedie-based terminal-grasp prediction. This section provides the formal derivation that converts the full-trajectory expectations in Eq. 28 into H-step trajectory expectations.

Step (1). To begin, we observe that the integrand in both the numerator and denominator of Eq. 28 depends on the trajectory beyond $\mathbf{g}_{\tau+H}$ only through the terminal state $\mathbf{g}_T$, since $\mathrm{d}\mathbf{w}_\tau$ is determined by $(\mathbf{g}_\tau, \mathbf{g}_{\tau+1})$ alone. This structure allows us to condition on the H-step lookahead trajectory and absorb everything beyond it into a single inner conditional expectation. Specifically, we denote the H-step lookahead trajectory by $\Gamma^H_\tau := (\mathbf{g}_{\tau+1}, \ldots, \mathbf{g}_{\tau+H})$, and apply the tower rule of conditional expectations together with the Markov property of $\mathcal{P}^0$. The numerator of Eq. 28 can then be rewritten as

$$\mathbb{E}_{\mathcal{P}^0}\left[\exp(-f_c(\mathbf{g}_T))\, \mathrm{d}\mathbf{w}_\tau \,\middle|\, \mathbf{g}_\tau\right] = \mathbb{E}_{\mathcal{P}^0(\Gamma^H_\tau \mid \mathbf{g}_\tau)}\left[\mathbb{E}_{\mathcal{P}^0}\left[\exp(-f_c(\mathbf{g}_T)) \,\middle|\, \mathbf{g}_{\tau+H}\right] \cdot \mathrm{d}\mathbf{w}_\tau\right], \tag{39}$$

where the outer expectation is over the H-step lookahead trajectory Γ^H_τ given $\mathbf{g}_\tau$, and the inner expectation is over the remaining trajectory beyond $\mathbf{g}_{\tau+H}$. By the Feynman-Kac formula in Eq. 18, the inner expectation is exactly the desirability function evaluated at the lookahead endpoint:

$$\mathbb{E}_{\mathcal{P}^0}\left[\exp(-f_c(\mathbf{g}_T)) \,\middle|\, \mathbf{g}_{\tau+H}\right] = \varphi(\mathbf{g}_{\tau+H}, \tau + H). \tag{40}$$

The same argument applies to the denominator. Substituting Eq. 40 into Eq. 39, and applying the corresponding identity for the denominator, Eq. 28 reduces to

$$\mathbf{u}^*_\tau(\mathbf{g}_\tau, \tau)\, \mathrm{d}\tau = \frac{\mathbb{E}_{\mathcal{P}^0(\Gamma^H_\tau \mid \mathbf{g}_\tau)}[\varphi(\mathbf{g}_{\tau+H}, \tau + H)\, \mathrm{d}\mathbf{w}_\tau]}{\mathbb{E}_{\mathcal{P}^0(\Gamma^H_\tau \mid \mathbf{g}_\tau)}[\varphi(\mathbf{g}_{\tau+H}, \tau + H)]}. \tag{41}$$

The expectations are now taken over only the H-step lookahead trajectory, with the dependence on the remaining trajectory entirely absorbed into the endpoint quantity $\varphi(\mathbf{g}_{\tau+H}, \tau + H)$.

Step (2). Eq. 41 still contains the endpoint desirability function $\varphi(\mathbf{g}_{\tau+H}, \tau + H)$, which is itself an expectation over the remaining trajectory and therefore cannot be evaluated by forward computation on f_c. To remove this remaining expectation, we apply Tweedie's formula [48] to predict the terminal grasp from the lookahead endpoint, as introduced in Sec. 4.2 of the main paper. Under the Tweedie operator $\mathcal{T}_{\tau+H}(\cdot)$ defined in the main paper, the single-step prediction of the terminal grasp from the lookahead endpoint is $\hat{\mathbf{g}}_T(\mathbf{g}_{\tau+H}) = \mathcal{T}_{\tau+H}(\mathbf{g}_{\tau+H})$, which we use to approximate the endpoint desirability function as

$$\varphi(\mathbf{g}_{\tau+H}, \tau + H) = \exp\big(-f_c(\hat{\mathbf{g}}_T(\mathbf{g}_{\tau+H}))\big). \tag{42}$$

Defining the lookahead weight as

$$W_\tau^H(\Gamma_\tau^H) := \exp\big(-f_c(\hat{\mathbf{g}}_T(\mathbf{g}_{\tau+H}))\big), \tag{43}$$

we obtain the finite-lookahead form of the optimal control law:

$$\mathbf{u}_\tau^*(\mathbf{g}_\tau, \tau)\,\mathrm{d}\tau = \frac{\mathbb{E}_{\mathcal{P}^0(\Gamma_\tau^H|\mathbf{g}_\tau)}\big[W_\tau^H(\Gamma_\tau^H)\,\mathrm{d}\mathbf{w}_\tau\big]}{\mathbb{E}_{\mathcal{P}^0(\Gamma_\tau^H|\mathbf{g}_\tau)}[W_\tau^H(\Gamma_\tau^H)]}. \tag{44}$$

Although W_τ^H depends on Γ_τ^H only through its endpoint $\mathbf{g}_{\tau+H}$ under the single-step Tweedie prediction, we deliberately write its argument as the entire trajectory Γ_τ^H. This trajectory-level notation provides a uniform interface for the importance-sampled reuse equality derived in Sec. E.5, where the proposal and target distributions differ in the entire trajectory rather than just the endpoint.

Step (3). Both the numerator and the denominator in Eq. 44 are expectations of H-step trajectory functions with respect to $\mathcal{P}^0(\Gamma_\tau^H \mid \mathbf{g}_\tau)$: the numerator integrand is $W_\tau^H(\Gamma_\tau^H)\cdot\mathrm{d}\mathbf{w}_\tau$, and the denominator integrand is $W_\tau^H(\Gamma_\tau^H)$ itself. To handle these and other such expectations uniformly in the amortized lookahead derivation in Sec. E.5, we abstract them into a single generic form. Let

$$\Psi(\Gamma_\tau^H) = \Psi(\mathbf{g}_{\tau+1}, \ldots, \mathbf{g}_{\tau+H}) \tag{45}$$

denote a real-valued function applied to the H-step lookahead trajectory Γ_τ^H, used to compute the quantities entering the finite-lookahead Monte Carlo instantiation. We refer to such a Ψ as a trajectory functional. Under this notation, both the numerator and the denominator in Eq. 44 take the unified form $\mathbb{E}_{\mathcal{P}^0(\Gamma_\tau^H|\mathbf{g}_\tau)}[\Psi(\Gamma_\tau^H)]$, recovered respectively by setting $\Psi(\Gamma_\tau^H) = W_\tau^H(\Gamma_\tau^H)\cdot\mathrm{d}\mathbf{w}_\tau$ and $\Psi(\Gamma_\tau^H) = W_\tau^H(\Gamma_\tau^H)$. This unified form is exactly the form taken by the left-hand side of the reuse equality in Eq. 7 of the main paper. The corresponding right-hand side, the importance-sampled correction-weighted form that enables reuse of stored trajectories, is derived in Sec. E.5.

E.5 Derivation of the reuse equality and the closed-form correction weight

In the amortized lookahead procedure described in Sec. 4.2 of the main paper, the H-step lookahead segment at denoising step $\tau+1$ overlaps with the segment stored from step τ in the time indices $\tau+2$ to $\tau+H$, motivating the reuse of stored states instead of fresh sampling. However, the stored segment was generated from the candidate state $\mathbf{g}_{\tau+1}^{(m)}$ rather than the actual current state $\mathbf{g}_{\tau+1}$, so directly treating the stored states as fresh samples would correspond to using an incorrect conditional distribution. To correct this distributional mismatch, we apply importance sampling, deriving the reuse equality in Eq. 7 of the main paper together with the closed-form expression of the correction weight $\rho_{\tau+1}^{(m)}$.

Step (1). To convert an expectation under the target distribution into one under the proposal distribution, we invoke the standard importance sampling identity, which states that for any two distributions p and q with q absolutely continuous with respect to p, and any function f, we have $\mathbb{E}_p[f] = \mathbb{E}_q[(p/q)\cdot f]$. In our setting, we identify the target distribution as $\mathcal{P}^0_{\tau+2:\tau+H}(\cdot \mid \mathbf{g}_{\tau+1})$, the conditional distribution of the H-step lookahead segment given the actual current state $\mathbf{g}_{\tau+1}$, and the proposal distribution as $\mathcal{P}^0_{\tau+2:\tau+H}(\cdot \mid \mathbf{g}_{\tau+1}^{(m)})$, the same distribution given the stored candidate state $\mathbf{g}_{\tau+1}^{(m)}$. We use $(\tilde{\mathbf{g}}_{\tau+2}^{(m)}, \ldots, \tilde{\mathbf{g}}_{\tau+H}^{(m)})$ to denote a sample drawn from the target distribution and $(\mathbf{g}_{\tau+2}^{(m)}, \ldots, \mathbf{g}_{\tau+H}^{(m)})$ to denote the stored sample drawn from the proposal distribution at the previous denoising step. Applying the importance sampling identity then gives

$$\mathbb{E}_{\mathcal{P}^0_{\tau+2:\tau+H}(\cdot|\mathbf{g}_{\tau+1})}\Big[\Psi(\tilde{\mathbf{g}}_{\tau+2}^{(m)}, \ldots, \tilde{\mathbf{g}}_{\tau+H}^{(m)})\Big] = \mathbb{E}_{\mathcal{P}^0_{\tau+2:\tau+H}(\cdot|\mathbf{g}_{\tau+1}^{(m)})}\Big[\rho_{\tau+1}^{(m)}\,\Psi(\mathbf{g}_{\tau+2}^{(m)}, \ldots, \mathbf{g}_{\tau+H}^{(m)})\Big], \tag{46}$$

which is precisely Eq. 7 of the main paper. Here $\rho_{\tau+1}^{(m)}$ is the trajectory-level likelihood ratio between the target and proposal distributions:

$$\rho_{\tau+1}^{(m)} := \frac{\mathcal{P}^0_{\tau+2:\tau+H}(\mathbf{g}_{\tau+2}^{(m)}, \ldots, \mathbf{g}_{\tau+H}^{(m)} \mid \mathbf{g}_{\tau+1})}{\mathcal{P}^0_{\tau+2:\tau+H}(\mathbf{g}_{\tau+2}^{(m)}, \ldots, \mathbf{g}_{\tau+H}^{(m)} \mid \mathbf{g}_{\tau+1}^{(m)})}. \tag{47}$$

Through this step, the reuse equality has been established, with $\rho_{\tau+1}^{(m)}$ defined as a trajectory-level likelihood ratio. However, evaluating this ratio directly requires computing the joint density of the

entire H-step segment under both distributions, which is computationally expensive. The next step reduces this trajectory-level ratio to a one-step transition density ratio.

Step (2). To simplify the trajectory-level ratio in Eq. 47, we exploit the Markov property of $\mathcal{P}^0$, under which the joint conditional distribution factorizes as a product of one-step transition densities:

$$\mathcal{P}^0_{\tau+2:\tau+H}(\mathbf{g}^{(m)}_{\tau+2}, \ldots, \mathbf{g}^{(m)}_{\tau+H} \mid \mathbf{g}_{\tau+1}) = p^0(\mathbf{g}^{(m)}_{\tau+2} \mid \mathbf{g}_{\tau+1}) \prod_{k=\tau+2}^{\tau+H-1} p^0(\mathbf{g}^{(m)}_{k+1} \mid \mathbf{g}^{(m)}_{k}), \tag{48}$$

and similarly for the proposal,

$$\mathcal{P}^0_{\tau+2:\tau+H}(\mathbf{g}^{(m)}_{\tau+2}, \ldots, \mathbf{g}^{(m)}_{\tau+H} \mid \mathbf{g}^{(m)}_{\tau+1}) = p^0(\mathbf{g}^{(m)}_{\tau+2} \mid \mathbf{g}^{(m)}_{\tau+1}) \prod_{k=\tau+2}^{\tau+H-1} p^0(\mathbf{g}^{(m)}_{k+1} \mid \mathbf{g}^{(m)}_{k}), \tag{49}$$

where $p^0(\cdot \mid \cdot)$ denotes the one-step transition density of $\mathcal{P}^0$. The two factorizations differ only in the first transition density, since the conditioning state ($\mathbf{g}_{\tau+1}$ vs. $\mathbf{g}^{(m)}_{\tau+1}$) only affects the transition into $\mathbf{g}^{(m)}_{\tau+2}$, while all subsequent transitions $p^0(\mathbf{g}^{(m)}_{k+1} \mid \mathbf{g}^{(m)}_{k})$ depend only on the preceding state $\mathbf{g}^{(m)}_{k}$ along the trajectory and are therefore identical between the two distributions. Substituting Eq. 48 and Eq. 49 into Eq. 47, all subsequent transition densities cancel exactly, leaving only the first-step transition density ratio:

$$\rho^{(m)}_{\tau+1} = \frac{p^0(\mathbf{g}^{(m)}_{\tau+2} \mid \mathbf{g}_{\tau+1})}{p^0(\mathbf{g}^{(m)}_{\tau+2} \mid \mathbf{g}^{(m)}_{\tau+1})}. \tag{50}$$

Through this step, the trajectory-level likelihood ratio has been reduced to a single one-step transition density ratio, but $p^0(\cdot \mid \cdot)$ itself remains abstract. The next step gives its closed form by exploiting the Gaussian structure of the discretized $\mathcal{P}^0$.

Step (3). The one-step transition density $p^0(\cdot \mid \cdot)$ is determined by the Euler-Maruyama discretization of the uncontrolled dynamics $\mathcal{P}^0$, namely Eq. 4 of the main paper with $\mathbf{u}_\tau \equiv 0$. Under this discretization, the one-step transition from a state $\mathbf{g}_s$ to the next state $\mathbf{g}_{s+1}$ is Gaussian with mean $\mathbf{g}_s + \mathbf{F}(\mathbf{g}_s, s)$ and covariance $\sigma^2(s)\,\mathbf{I}$:

$$p^0(\mathbf{g}_{s+1} \mid \mathbf{g}_s) = \mathcal{N}\big(\mathbf{g}_{s+1};\ \mathbf{g}_s + \mathbf{F}(\mathbf{g}_s, s),\ \sigma^2(s)\,\mathbf{I}\big). \tag{51}$$

Substituting Eq. 51 into Eq. 50 for both the numerator (with $\mathbf{g}_s = \mathbf{g}_{\tau+1}$) and the denominator (with $\mathbf{g}_s = \mathbf{g}^{(m)}_{\tau+1}$), the Gaussian normalization constants are identical and cancel exactly, leaving only the exponential terms. The correction weight $\rho^{(m)}_{\tau+1}$ then admits the closed-form expression

$$\rho^{(m)}_{\tau+1} = \exp\Big(-\frac{\big\|\mathbf{g}^{(m)}_{\tau+2} - \mathbf{g}_{\tau+1} - \mathbf{F}(\mathbf{g}_{\tau+1}, \tau+1)\big\|^2}{2\sigma^2(\tau+1)} + \frac{\big\|\mathbf{g}^{(m)}_{\tau+2} - \mathbf{g}^{(m)}_{\tau+1} - \mathbf{F}(\mathbf{g}^{(m)}_{\tau+1}, \tau+1)\big\|^2}{2\sigma^2(\tau+1)}\Big). \tag{52}$$

where a larger $\rho^{(m)}_{\tau+1}$ indicates that the m-th stored trajectory is more representative under the current target distribution, and a smaller value indicates the opposite. When all $\rho^{(m)}_{\tau+1}$ become small, we resample fresh trajectories from $\mathbf{g}_{\tau+1}$, which is still efficient and shares the per-step cost of the initial sampling. This closed-form expression is forward-evaluable using $\mathbf{F}$ and σ already available from the trained diffusion model, and can be efficiently computed at each denoising step. Substituting $\rho^{(m)}_{\tau+1}$ into Eq. 46 yields the importance-sampled, correction-weighted estimator over the stored trajectory, which forms the basis of the amortized lookahead procedure described in Sec. 4.2 of the main paper.

F Overall Process of NDPP-Grasp

Our method can be flexibly integrated into various existing diffusion-based methods for task-oriented dexterous grasp generation. During training, we do not modify the training procedure of their grasp diffusion models. Instead, we follow the original methods to train a grasp diffusion model that generates task-aligned grasps conditioned on the object and task specification. During inference, however, we replace the post-generation refinement stage commonly used for physical plausibility. Rather than first generating a task-aligned grasp and then refining it, we directly apply the proposed non-differentiable physical plausibility guidance during the denoising process of the trained diffusion model. In this way, the generated grasp is guided to be both task-aligned and physically plausible throughout denoising. The pseudo-algorithm of our NDPP-Grasp framework is summarized in Algorithm 1.

Algorithm 1 NDPP-Grasp inference

Require: Trained grasp diffusion model with score $\nabla \log p_s$, object $\mathcal{O}$, task specification $\mathcal{L}$, cost evaluator f_c, lookahead horizon H, sample size M, total denoising steps T
Ensure: Generated grasp $\mathcal{G}$

$\mathbf{g}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
Sample M initial H-step uncontrolled suffixes from $\mathbf{g}_0$ under $\mathcal{P}^0$, storing both states and Brownian increments
for $\tau = 0, 1, \ldots, T-1$ **do**
  $H_\tau \leftarrow \min(H, T-\tau)$ ▷ Truncate horizon near terminal
  for $m = 1, \ldots, M$ **do**
    $\hat{\mathbf{g}}_T^{(m)} \leftarrow \mathrm{Tweedie}(\mathbf{g}_{\tau+H_\tau}^{(m)})$ ▷ Endpoint terminal grasp prediction
    $W_\tau^{H,(m)} \leftarrow \exp(-f_c(\hat{\mathbf{g}}_T^{(m)}))$ ▷ Forward query of f_c, no gradient
    Compute $\rho_\tau^{(m)}$ via Eq. 52 ▷ $\rho_0^{(m)} \equiv 1$ at τ=0
  end for
  Compute $\mathbf{u}_\tau \, \mathrm{d}\tau$ via Eq. 57 using weights $\{\rho_\tau^{(m)} W_\tau^{H,(m)}\}$ and stored increments $\{\mathrm{d}\mathbf{w}_\tau^{(m)}\}$
  Sample fresh $\mathrm{d}\mathbf{w}_\tau^{\mathrm{act}} \sim \mathcal{N}(\mathbf{0}, \mathrm{d}\tau \, \mathbf{I})$
  Advance $\mathbf{g}_{\tau+1}$ from $\mathbf{g}_\tau$ via Eq. 4 of the main paper with $\mathbf{u}_\tau$ and $\mathrm{d}\mathbf{w}_\tau^{\mathrm{act}}$
  Shift and extend each stored suffix by one new uncontrolled transition under $\mathcal{P}^0$
end for
$\mathcal{G} \leftarrow \Phi(\mathbf{g}_T)$
return $\mathcal{G}$

G More Details about NDPP-Grasp

G.1 Monte Carlo instantiation under natural finite-lookahead

The optimal control law in Eq. 44 expresses $\mathbf{u}_\tau^* \, \mathrm{d}\tau$ as a ratio of two expectations over H-step lookahead trajectories under the uncontrolled dynamics $\mathcal{P}^0$. This subsection describes the Monte Carlo procedure used to evaluate this ratio at each denoising step in the natural finite-lookahead instantiation introduced in Sec. 4.2 of the main paper.

At denoising step τ, given the current state $\mathbf{g}_\tau$, we draw M independent H-step lookahead trajectories under $\mathcal{P}^0$. For $m = 1, \ldots, M$, the m-th trajectory $\Gamma_\tau^{H,(m)} = (\mathbf{g}_{\tau+1}^{(m)}, \ldots, \mathbf{g}_{\tau+H}^{(m)})$ is generated by Euler-Maruyama discretization of $\mathcal{P}^0$. The first step samples

$$\mathbf{g}_{\tau+1}^{(m)} = \mathbf{g}_\tau + \mathbf{F}(\mathbf{g}_\tau, \tau) \, \mathrm{d}\tau + \sigma(\tau) \, \mathrm{d}\mathbf{w}_\tau^{(m)}, \qquad \mathrm{d}\mathbf{w}_\tau^{(m)} \sim \mathcal{N}(0, \mathrm{d}\tau \, \mathbf{I}), \tag{53}$$

and the subsequent $H-1$ steps sample $\mathbf{g}_{k+1}^{(m)}$ from $\mathbf{g}_k^{(m)}$ for $k = \tau+1, \ldots, \tau+H-1$ via the same Euler-Maruyama discretization. For each trajectory, we then apply the Tweedie operator $\mathcal{T}_{\tau+H}$ to the lookahead endpoint to obtain the predicted terminal grasp $\hat{\mathbf{g}}_T^{(m)} = \mathcal{T}_{\tau+H}(\mathbf{g}_{\tau+H}^{(m)})$, and compute the lookahead weight

$$W_\tau^{H,(m)} = \exp\big(-f_c(\hat{\mathbf{g}}_T^{(m)})\big), \tag{54}$$

which corresponds to the m-th sample of the trajectory functional W_τ^H defined in Eq. 43.

With the M sampled lookahead weights $\{W_\tau^{H,(m)}\}_{m=1}^M$ and the corresponding first-step Brownian increments $\{\mathrm{d}\mathbf{w}_\tau^{(m)}\}_{m=1}^M$, the two expectations in Eq. 44 are estimated by their empirical averages, and the resulting Monte Carlo instantiation of the optimal control law is

$$\mathbf{u}_\tau^*(\mathbf{g}_\tau, \tau) \, \mathrm{d}\tau = \frac{\sum_{m=1}^M W_\tau^{H,(m)} \, \mathrm{d}\mathbf{w}_\tau^{(m)}}{\sum_{m=1}^M W_\tau^{H,(m)}}, \tag{55}$$

where the empirical-average normalization factors cancel between numerator and denominator. This estimator gives the current-step Monte Carlo instantiation of Eq. 6 of the main paper under the natural finite-lookahead procedure introduced in Sec. 4.2 of the main paper.

G.2 Monte Carlo instantiation under amortized lookahead

In the amortized lookahead procedure introduced in Sec. 4.2 of the main paper, the lookahead trajectory at each denoising step is largely reused from the previous step rather than sampled fresh. This subsection describes the Monte Carlo procedure used to evaluate the optimal control law at the current step under this reuse, building on the reuse equality and the closed-form correction weight derived in Sec. E.5.

Suppose at the previous denoising step τ, we have stored M trajectories $\{(\mathbf{g}_{\tau+1}^{(m)}, \mathbf{g}_{\tau+2}^{(m)}, \dots, \mathbf{g}_{\tau+H}^{(m)})\}_{m=1}^{M}$, where each trajectory was generated by Euler-Maruyama discretization of $\mathcal{P}^0$ starting from the candidate state $\mathbf{g}_{\tau+1}^{(m)}$. At the current step $\tau+1$, given the actual current state $\mathbf{g}_{\tau+1}$, we evaluate the optimal control law using these stored trajectories in place of fresh samples.

For each $m = 1, \dots, M$, we first reuse the overlapping segment $(\mathbf{g}_{\tau+2}^{(m)}, \dots, \mathbf{g}_{\tau+H}^{(m)})$ directly from storage, and extend the trajectory by one step under $\mathcal{P}^0$ by sampling $\mathbf{g}_{\tau+H+1}^{(m)}$ from $\mathbf{g}_{\tau+H}^{(m)}$ via Euler-Maruyama discretization. We apply the Tweedie operator $\mathcal{T}_{\tau+H+1}$ at the new endpoint to obtain the predicted terminal grasp $\hat{\mathbf{g}}_T^{(m)} = \mathcal{T}_{\tau+H+1}(\mathbf{g}_{\tau+H+1}^{(m)})$, and compute the lookahead weight at the current step, where $W_{\tau+1}^{H,(m)} = \exp\big(-f_c(\hat{\mathbf{g}}_T^{(m)})\big)$.

We then compute the correction weight $\rho_{\tau+1}^{(m)}$ in closed form using Eq. 52, evaluated with the actual current state $\mathbf{g}_{\tau+1}$, the stored candidate state $\mathbf{g}_{\tau+1}^{(m)}$, and the stored next-step state $\mathbf{g}_{\tau+2}^{(m)}$. Finally, we compute the first-step Brownian increment $\mathrm{d}\mathbf{w}_{\tau+1}^{(m)}$ that enters the numerator of the control law in Eq. 44. Since this control law is evaluated under the target distribution conditional on the actual current state $\mathbf{g}_{\tau+1}$, this Brownian increment combines the stored next-step state $\mathbf{g}_{\tau+2}^{(m)}$ with the actual current state $\mathbf{g}_{\tau+1}$,

$$\mathrm{d}\mathbf{w}_{\tau+1}^{(m)} = \frac{\mathbf{g}_{\tau+2}^{(m)} - \mathbf{g}_{\tau+1} - \mathbf{F}(\mathbf{g}_{\tau+1}, \tau+1)\,\mathrm{d}\tau}{\sigma(\tau+1)}, \tag{56}$$

as required by the importance sampling framework in Sec. E.5.

Substituting these quantities into the reuse equality in Eq. 46, the two expectations in Eq. 44 (instantiated at step $\tau+1$) are estimated by their importance-sampling-weighted empirical averages over the stored trajectories, giving the Monte Carlo instantiation of the optimal control law at the current step,

$$\mathbf{u}_{\tau+1}^{*}(\mathbf{g}_{\tau+1}, \tau+1)\,\mathrm{d}\tau = \frac{\sum_{m=1}^{M} \rho_{\tau+1}^{(m)}\, W_{\tau+1}^{H,(m)}\, \mathrm{d}\mathbf{w}_{\tau+1}^{(m)}}{\sum_{m=1}^{M} \rho_{\tau+1}^{(m)}\, W_{\tau+1}^{H,(m)}}, \tag{57}$$

where the empirical-average normalization factors cancel between numerator and denominator. The extended trajectory $(\mathbf{g}_{\tau+2}^{(m)}, \dots, \mathbf{g}_{\tau+H+1}^{(m)})$ is then stored for the next denoising step, where $\mathbf{g}_{\tau+2}^{(m)}$ takes over the role of the candidate state and $(\mathbf{g}_{\tau+3}^{(m)}, \dots, \mathbf{g}_{\tau+H+1}^{(m)})$ becomes the reused overlapping segment. Notably, the memory cost of these stored trajectories is negligible, since the M lookahead trajectories consist of low-dimensional grasp states and amount to only $\mathcal{O}(MH)$ floats per generation.

G.3 Score function recovery under different denoising parameterizations.

The reverse-time SDE in Eq. 1 of the main paper requires the score function $\nabla_{\mathbf{y}_s} \log p_s(\mathbf{y}_s \mid \mathcal{O}, \mathcal{L})$ to drive the denoising trajectory. In practice, however, different diffusion-based grasp generation backbones train the network to predict different quantities. To make our framework compatible with these backbones, here we show how the score function can be recovered in closed form from the two most common denoising parameterizations, so that the NDPP-Grasp framework can be plugged in without modification to its underlying control formulation. Specifically, when the denoising network adopts the clean-sample prediction (i.e., $\mathbf{y}_0$-parameterization) and predicts the clean grasp representation $\mathbf{y}_0^{\theta}(\mathbf{y}_s, s)$ from the noisy input $\mathbf{y}_s$, the score function is recovered as

$$\nabla_{\mathbf{y}_s} \log p_s(\mathbf{y}_s \mid \mathcal{O}, \mathcal{L}) = \frac{\sqrt{\bar{\alpha}_s}\, \mathbf{y}_0^{\theta}(\mathbf{y}_s, s) - \mathbf{y}_s}{1 - \bar{\alpha}_s}, \tag{58}$$

where $\bar{\alpha}_s$ is the cumulative noise schedule coefficient. When the denoising network adopts the noise prediction (i.e., $\boldsymbol{\epsilon}$-parameterization) and predicts the added noise $\boldsymbol{\epsilon}_\theta(\mathbf{y}_s, s)$, the score function is recovered as

$$\nabla_{\mathbf{y}_s} \log p_s(\mathbf{y}_s \mid \mathcal{O}, \mathcal{L}) = -\frac{\boldsymbol{\epsilon}_\theta(\mathbf{y}_s, s)}{\sqrt{1-\bar{\alpha}_s}}. \tag{59}$$

Note that the two parameterizations are mathematically equivalent under the forward diffusion relationship $\mathbf{y}_s = \sqrt{\bar{\alpha}_s}\,\mathbf{y}_0 + \sqrt{1-\bar{\alpha}_s}\,\boldsymbol{\epsilon}$, and our framework operates at the level of the score function rather than a specific network architecture. As a result, the SOC formulation in Sec. 4.1 of the main paper and the gradient-free derivation in Sec. E.2 apply directly under either parameterization, allowing NDPP-Grasp to be plugged into different diffusion-based backbone trained under either parameterization without modifying the underlying control formulation.

H Limitations and Future Work

Failure cases and sim-to-real transfer. When grasping objects with very smooth surfaces such as mugs, although grasp generation methods can generate a valid grasp, the dexterous hand's rigid fingertips may provide insufficient friction, potentially causing slippage. Nevertheless, we observe that our method achieves a 60% real-robot experiment success rate even in this challenging setting, outperforming 20% achieved by DexGYS [3]. The inherent sim-to-real gap is a common challenge in the embodied intelligence community, and handling this challenge is an important and useful future direction in this whole area.

Potential safety implications. (i) Physical operation risks. During human-robot interaction, attempting to grasp objects that exceed the robotic hand's payload capacity may cause physical harm. **(ii) Adversarial risks.** Adversarial attacks on learning-based decision-making systems may cause task failure or, in robotic deployments, even dangerous actions [60, 61, 62]. Future work in this area can include (a) further improving the robotic hand's manipulation and payload capabilities and (b) designing stronger defenses to deal with potential adversarial attacks and unauthorized reuse of diffusion models [63].

I Licenses

We use the DexGYSNet dataset [3] by following the license of here, and use the DexTOG-80K dataset [4] by following the license of here.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In both the abstract and the introduction, we have clearly mentioned the task (scope) and the contributions of this paper

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss potential failure cases, sim-to-real transfer, and safety implications in Appendix H.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: This paper does include such complete and correct theoretical analysis (proof) in Appendix.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: This paper builds its code based on the diffusion-based task-oriented dexterous generation methods. We have clearly mentioned (via descriptions and equations) how to reproduce the main experimental results upon the existing diffusion-based task-oriented dexterous generation methods.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: At this submission stage, we are sorry that we do not get enough approval to open-source our code.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: This paper has specified its data splits, its introduced hyperparameters, and other details in the Experiment section and Appendix.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: To make a fair comparison with the existing works, we follow their experimental settings which do not including error bars.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We introduce the type of GPU and the time of execution in the Experiments section.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The authors have read through the NeurIPS Code of Ethics and carefully conform with it.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No]

    Justification: This paper focuses on handling a crucial problem in task-oriented dexterous grasp generation. To the best of our knowledge, this work is methodological and we do not foresee direct societal impacts specific to this work.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: To the best of our knowledge, this paper does not newly bring any risk for misuse and is thus not applicable to this question.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have discussed the licenses in our Appendix.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: This paper does not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLM is not an important, original, or non-standard component of the core methodology in this study.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.