CoSCoDiff: Consensus Splitting for Coordinating Objectives and Constraints in Diffusion Models

Anonymous authors
Paper under double-blind review

Abstract

Controllable diffusion models are used for task-specific generation, which often optimize objectives while satisfying feasibility constraints. Existing methods commonly optimize these requirements jointly, where conflicting updates can degrade sample quality and weaken constraint satisfaction. To address this issue, we explicitly separate constraints from objectives and formulate controllable diffusion sampling as a constrained stochastic optimal control problem amenable to variable splitting. Building on this formulation, we propose **CoSCoDiff**, a training-free framework that coordinates the two components through terminal-state consensus splitting. To approximate the intractable trajectory-level optimization, CoSCoDiff combines sequential rollout with Tweedie prediction to construct tractable local control updates. These updates propagate objective and consensus signals to intermediate noisy states, while a constraint operator enforces feasibility on the clean sample. The framework accommodates existing objective-guided methods and supports both primal-only and primal–dual coordination, allowing historical dual feedback to be retained or omitted in the presence of approximation errors. Comprehensive experiments demonstrate that CoSCoDiff effectively coordinates objective optimization and hard constraint satisfaction, and achieves the best overall performance among the baselines across heterogeneous control requirements.

1 Introduction

Diffusion models (Ho et al., 2020; Song et al., 2021) have emerged as a powerful class of generative models, demonstrating strong performance across diverse domains ranging from image synthesis and video generation to trajectory planning and protein design (Ho et al., 2020; Rombach et al., 2022a; Ho et al., 2022; Janner et al., 2022; Watson et al., 2023). Owing to their principled stochastic formulation, diffusion models can be steered at inference time toward task objectives and constraints, which has made controllable diffusion widely used for task-specific generation (Song et al., 2021; Dhariwal & Nichol, 2021; Ho & Salimans, 2022; Bansal et al., 2023). Most existing approaches achieve controllability toward desirable properties through differentiable task-specific objectives, such as rewards (Chung et al., 2022; Kim et al., 2025), energies (Yu et al., 2023), and general guidance losses (Song et al., 2023; He et al., 2023; Guo et al., 2024; Shen et al., 2024; Ye et al., 2024). Many scientific and engineering applications require sample generation to obey constraints, such as explicit geometric, structural, or physical requirements (Lou & Ermon, 2023). Controllable diffusion methods incorporate such constraints through geometry-aware dynamics (Fishman et al., 2024), mirror maps (Liu et al., 2023a), constraint-specific noising and sampling (Fishman et al., 2023), or diffusion bridges on constrained domains (Liu et al., 2023b). These approaches are typically tied to a prescribed constraint domain and require constraint-specific model design or training.

In practice, task-specific generation often involves both task objectives and feasibility constraints, as in collision-free trajectory planning (Giaretta et al., 2026) and controllable image generation (Shen et al., 2024). Existing approaches commonly incorporate such requirements through loss-based guidance (Song et al., 2023) or per-step constrained optimization (Khalafi et al., 2024). However, when the preferred objective solution lies outside the feasible set, objective and constraint updates can become misaligned during denoising. Their joint treatment can therefore make feasibility sensitive to weighting choices and yield a local trade-off between task performance and constraint sat-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

isfaction. Recent constrained diffusion methods make feasibility more explicit through projection-based constrained sampling (Christopher et al., 2024; Rochman-Sharabi & Louppe, 2026), primal–dual or augmented-Lagrangian inference (Zhang et al., 2026), and terminal clean-state constrained correction (Giaretta et al., 2026). However, objective optimization and constraint enforcement remain typically coupled within the same denoising update or constrained subproblem, or handled by a simple post-hoc correction. The former can induce competing local updates when the two requirements conflict, whereas the latter does not coordinate feasibility correction with the generation trajectory and may distort the resulting sample.

To address this limitation, we propose CoSCoDiff, which formulates *objective-guided constrained diffusion* as a stochastic optimal control problem to allow consensus splitting at the terminal clean state. Unlike prior splitting-based diffusion methods, CoSCoDiff splits the feasibility constraint from objective-guided diffusion. This creates a new coordination problem because feasibility is defined on the clean sample, while diffusion evolves through noisy intermediate states. CoSCoDiff addresses this mismatch with local control updates derived from sequential rollout and Tweedie prediction (Chung et al., 2022), which propagate objective and consensus signals to intermediate noisy states. The two blocks are coordinated through primal-only or primal–dual updates, and an SDEdit-style warm restart (Meng et al., 2022) re-noises the latest feasible sample between outer iterations, preserving feasible structure while allowing further objective improvement. Our main contributions are summarized as follows.

- To alleviate the compromise between task objectives and constraints, we formulate controllable sampling as a constrained stochastic optimal control problem and decouple it into an objective block and a feasibility block via consensus splitting.
- We propose CoSCoDiff, which propagates terminal consensus through sequential rollout and Tweedie prediction, and uses an SDEdit-inspired warm restart to preserve feasible progress across successive rollouts. The framework supports both primal-only and primal-dual coordination and accommodates existing guidance methods and task-specific feasibility operators.
- Experiments across four tasks show that CoSCoDiff maintains high constraint satisfaction while improving task objectives, achieving the best overall performance among the evaluated baselines.

2 Related Work

Controllable Diffusion Models. Controllable diffusion models steer generation toward task-specific requirements through either training-based adaptation or inference-time control. Training-based methods learn controllability through conditional training, task-specific fine-tuning, or additional conditioning modules (Rombach et al., 2022b; Zhang et al., 2023; Mou et al., 2023; Li et al., 2023; Huang et al., 2023). Despite their effectiveness, training-based approaches generally require additional adaptation when new control signals are introduced.

Training-free methods instead modify the sampling process of pretrained diffusion models directly at inference time. Prompt-to-Prompt (Hertz et al., 2022) enables text-guided editing by manipulating cross-attention maps, while SDEdit (Meng et al., 2022) performs image-guided editing. More recent plug-and-play methods enable objective-driven control through likelihoods (Chung et al., 2022), losses (Shen et al., 2024), energy functions (Yu et al., 2023), rewards (Kim et al., 2025), and general differentiable objectives (Guo et al., 2024). TFG (Ye et al., 2024) further provides a unified view of a broad class of training-free guidance methods. CoSCoDiff is compatible with these methods, which can instantiate its objective-guided diffusion block.

Another body of work considers explicit geometric, structural, physical, or feasibility requirements in diffusion generation (Liu et al., 2023a; Fishman et al., 2023; 2024). Recent methods strengthen constraint satisfaction through projected sampling (Christopher et al., 2024; Rochman-Sharabi & Louppe, 2026), primal-dual updates (Zhang et al., 2026), or clean-state correction (Giaretta et al., 2026). These approaches provide increasingly effective mechanisms for constraint enforcement, but objective optimization and feasibility are typically handled within the same sampling procedure or constrained subproblem. CoSCoDiff preserves their distinct optimization roles and coordinates objective-guided diffusion with feasibility constraints through consensus splitting.

Optimization Perspectives on Diffusion Models. Many works have recast different components of diffusion-based generation as optimization, especially optimal-control problems. Guo et al. (2024)

interprets gradient-guided diffusion as a regularized optimization problem, while Chang et al. (2026) further formulates training-free reward-guided diffusion as a trajectory optimal-control problem. From the constraint side, Christopher et al. (2024) reinterpret diffusion sampling as constrained optimization with projected updates, while Zampini et al. (2025) extend it to stable diffusion models, enabling training-free generation under explicit domain-specific constraints. These works show that both objective guidance and constraint enforcement admit natural optimization-based interpretations, providing a common foundation for optimization-driven controllable diffusion.

Splitting methods provide a general mechanism for decoupling heterogeneous optimization components that are difficult to handle jointly. In diffusion sampling, Wizadwongsa & Suwajanakorn (2023) employ operator splitting to separate the diffusion dynamics from the guidance term, while DiffPIR (Zhu et al., 2023) incorporates diffusion priors into a half-quadratic splitting (HQS) framework that alternates denoising and data-consistency updates without dual accumulation. ADMM (Gabay & Mercier, 1976; Glowinski, R. & Marroco, A., 1975) extends such variable splitting with dual feedback: Plug-and-Play ADMM (Chan et al., 2017) separates data fidelity from prior regularization, ADMMDiff (Zhang et al., 2025) decouples the pretrained diffusion prior from a differentiable guidance objective, and Du et al. (2026) further introduce dual-variable coupling between measurement consistency and diffusion priors. In contrast, CoSCoDiff separates objective-guided diffusion from explicit feasibility constraint at the clean state and coordinates the two through consensus splitting, supporting both a HQS-style primal-only realization and an ADMM-style primal–dual realization with accumulated dual feedback.

3 Preliminaries

Diffusion Models. Given data $x_0 \sim p_{\mathrm{data}}$, a continuous-time diffusion model gradually perturbs the data distribution through the forward stochastic differential equation (SDE)

$$dx_t = f(x_t, t)\, dt + g(t)\, dW_t, \qquad t : 0 \to T, \tag{1}$$

where f and g denote the drift and diffusion coefficients, respectively, and W_t is a standard Wiener process. For a sufficiently large T, the terminal distribution p_T approaches a tractable noise distribution, typically a standard Gaussian. The corresponding reverse-time SDE is

$$dx_t = \big[f(x_t, t) - g^2(t)\nabla_{x_t} \log p_t(x_t)\big]\, dt + g(t)\, d\bar{W}_t, \qquad t : T \to 0, \tag{2}$$

where $\nabla_{x_t} \log p_t(x_t)$ is the score function and $\bar{W}_t$ denotes the reverse-time Wiener process. In practice, the unknown score is approximated by a neural network $s_\theta(x_t, t) \approx \nabla_{x_t} \log p_t(x_t)$.

Controllable diffusion modifies the reverse sampling process to steer generated samples toward desired properties. For guidance-based methods, this can be expressed by augmenting the pretrained score with a guidance term $G(x_t, t)$:

$$dx_t = \big[f(x_t, t) - g^2(t)\big(s_\theta(x_t, t) + G(x_t, t)\big)\big]\, dt + g(t)\, d\bar{W}_t. \tag{3}$$

When $G \equiv 0$, the process reduces to unconditional generation, whereas different choices of G recover various forms of conditional or controllable sampling. This formulation provides the basis for introducing the residual guidance control in CoSCoDiff.

Optimal Control and Consensus Splitting. Consider a finite-horizon optimal control problem over the state–control trajectory $(\mathbf{X}, \mathbf{U})$:

$$\min_{\mathbf{X},\mathbf{U}} \quad F(\mathbf{X}, \mathbf{U}) \qquad \text{s.t.} \quad (\mathbf{X}, \mathbf{U}) \in \mathcal{A}, \tag{4}$$

where F denotes the trajectory-level control objective and $\mathcal{A}$ is the admissible set induced by system dynamics and other feasibility requirements. Using the indicator function $I_{\mathcal{A}}$, Eq.4 can be equivalently written as the minimization of $F(\mathbf{X}, \mathbf{U}) + I_{\mathcal{A}}(\mathbf{X}, \mathbf{U})$.

Variable splitting introduces an auxiliary trajectory $(\tilde{\mathbf{X}}, \tilde{\mathbf{U}})$ and separates the two components through a consensus constraint:

$$\min_{\mathbf{X},\mathbf{U},\tilde{\mathbf{X}},\tilde{\mathbf{U}}} \quad F(\mathbf{X}, \mathbf{U}) + I_{\mathcal{A}}(\tilde{\mathbf{X}}, \tilde{\mathbf{U}}) \qquad \text{s.t.} \quad (\mathbf{X}, \mathbf{U}) = (\tilde{\mathbf{X}}, \tilde{\mathbf{U}}). \tag{5}$$

The two blocks can then be coordinated by a quadratic consensus penalty. Primal-only splitting methods, such as HQS, alternate between the two primal updates without retaining historical consensus errors. ADMM additionally introduces a scaled dual variable that accumulates the consensus residual across iterations. These primal-only and primal–dual variants provide the general splitting template that we later specialize to controllable diffusion.

4 Problem Formulation of Controllable Diffusion Sampling

Controllable diffusion often needs to optimize a task objective under feasibility constraints. To avoid coupling these requirements in the denoising process, we formulate the problem as constrained stochastic optimal control, where task optimization, prior preservation, and feasibility can be represented separately. We consider a task objective and feasibility constraints defined on the clean sample x_0.

- **Task objective.** A task-specific cost $h(x_0)$, typically differentiable, where smaller values indicate better task performance.
- **Constraints.** The generated sample must satisfy $x_0 \in \mathcal{C}$, where the feasible set $\mathcal{C}$ may encode differentiable or non-smooth geometric, physical, safety, or other task-specific requirements.

Given a pretrained diffusion model parameterized by a score network $s_\theta(x_t, t)$, we introduce a guidance control $u_t \equiv G(x_t, t)$ to steer the reverse process toward the task requirements. The controlled reverse SDE follows Eq. 3 with the guidance term replaced by u_t. We denote the resulting state–control trajectory by $(\mathbf{X}, \mathbf{U}) = \{(x_t, u_t)\}_{t=0}^{T}$. To preserve the pretrained prior, deviations of the controlled process from the uncontrolled one should be penalized. By Girsanov's theorem (Karatzas & Shreve, 1998), the KL divergence between the two path measures is $\frac{1}{2}\mathbb{E}\int_0^T g^2(t)\|u_t\|^2\,dt$. We therefore formulate controllable diffusion sampling as

$$\begin{aligned} \min_{\mathbf{X},\mathbf{U}} \quad & \mathbb{E}\left[h(x_0) + \frac{1}{2}\int_0^T g^2(t)\|u_t\|^2\,dt\right] \\ \text{subject to} \quad & \begin{cases} x_0 \in \mathcal{C}, \\ dx_t = \big[f(x_t,t) - g^2(t)\big(s_\theta(x_t,t) + u_t\big)\big]\,dt + g(t)\,d\bar{W}_t, \quad t \in [0,T), \\ x_T \sim \mathcal{N}(0, I), \end{cases} \end{aligned} \tag{6}$$

where the two objective terms correspond to task optimization and prior preservation, and $x_0 \in \mathcal{C}$ imposes feasibility as a hard terminal constraint.

5 The Proposed CoSCoDiff Framework

In this section, we present CoSCoDiff to approximately solve Eq. 6 by treating objective-guided diffusion and feasibility enforcement as two coupled blocks. The resulting trajectory-level primal subproblem is approximated by sequential rollout with Tweedie prediction, which converts terminal consensus into local control updates at noisy states. The two blocks are then coordinated through either primal-only or primal–dual updates, and an SDEdit-inspired warm restart carries the latest feasible sample across outer iterations. The complete procedure is summarized in Algorithm 1.

Consensus Splitting for Controllable Diffusion Sampling. To decouple objective-guided diffusion from terminal feasibility enforcement, we perform the general consensus-splitting formulation to the clean sample x_0. We define the admissible set $\mathcal{A}$ as the collection of state–control trajectories $(\mathbf{X}, \mathbf{U})$ satisfying the controlled reverse SDE in Eq.6 and the prescribed prior $x_T \sim \mathcal{N}(0, I)$. Since the feasibility requirement is imposed only on the clean sample, we introduce an auxiliary terminal variable $\tilde{x}_0$ and rewrite $x_0 \in \mathcal{C}$ as $x_0 = \tilde{x}_0$, $\tilde{x}_0 \in \mathcal{C}$. With an indicator function $I_{\mathcal{C}}$ of the feasible set, Eq.6 is reformulated as

$$\begin{aligned} \min_{\mathbf{X},\mathbf{U},\tilde{x}_0} \quad & \mathbb{E}\left[h(x_0) + \frac{1}{2}\int_0^T g^2(t)\|u_t\|_2^2\,dt + I_{\mathcal{C}}(\tilde{x}_0)\right] \\ \text{subject to} \quad & (\mathbf{X}, \mathbf{U}) \in \mathcal{A}, \qquad x_0 = \tilde{x}_0. \end{aligned} \tag{7}$$

The objective block now involves only the diffusion trajectory, while the feasibility block involves only $\tilde{x}_0$. Relaxing the consensus constraint with a quadratic penalty, we alternate between a primal update and a feasibility update,

$$(\mathbf{X}^{k+1}, \mathbf{U}^{k+1}) = \arg \min_{(\mathbf{X},\mathbf{U})\in\mathcal{A}} \mathbb{E}\left[h(x_0) + \frac{1}{2}\int_0^T g^2(t)\|u_t\|_2^2\, dt + \frac{\rho}{2}\left\|x_0 - \tilde{x}_0^k - z_0^k\right\|_2^2 \right], \quad (8)$$

$$\tilde{x}_0^{k+1} = \mathcal{P}_{\mathcal{C}}\left(x_0^{k+1} - z_0^k\right), \quad (9)$$

where $\rho > 0$ is the consensus penalty parameter, initialization $\tilde{x}_0^0 = \mathcal{P}_{\mathcal{C}}(x_0^0)$, and $\mathcal{P}_{\mathcal{C}}$ denotes a task-specific feasibility operator. It reduces to the projection $\Pi_{\mathcal{C}}$ when an exact projection is available, while more general constraints may employ iterative or domain-specific feasibility corrections.

After the primal and feasibility updates, the treatment of the consensus residual determines how information is carried across outer iterations. This yields two coordination variants:

$$z_0^{k+1} = \begin{cases} 0, & \text{CoSCoDiff-P}, \\ z_0^k + \tilde{x}_0^{k+1} - x_0^{k+1}, & \text{CoSCoDiff-PD}. \end{cases} \quad (10)$$

With $z_0^0 = 0$, CoSCoDiff-P performs primal-only consensus splitting, whereas CoSCoDiff-PD retains scaled dual feedback through an ADMM-style primal–dual update. The role and limitations of historical dual feedback are further discussed in Appendix B. The three steps are detailed below:

Step 1: Guided Rollout (Eq.8). For fixed $(\tilde{x}_0^k, z_0^k)$, the primal update is a trajectory-level stochastic optimal control problem with terminal consensus loss $\mathcal{L}_{\text{con}}^k(x_0) = h(x_0) + \frac{\rho}{2}\left\|x_0 - \tilde{x}_0^k - z_0^k\right\|_2^2$. Introducing the value function $V^k(t, x_t)$ and applying Bellman's principle of optimality gives the local control problem

$$u_t^* = \arg\min_{u_t} \mathbb{E}\left[\frac{1}{2} g^2(t)\|u_t\|_2^2 \Delta t + V^k(t - \Delta t, x_{t-\Delta t}) \,\middle|\, x_t, u_t \right]. \quad (11)$$

Since direct evaluation of the future value is intractable, we adopt a nominal continuation policy π_0 with zero future control, i.e., $u_\tau = 0$ for $\tau < t$, and approximate

$$\mathbb{E}\left[V^k(t - \Delta t, x_{t-\Delta t}) \mid x_t, u_t\right] \approx \mathbb{E}_{\pi_0}\left[\mathcal{L}_{\text{con}}^k(\hat{x}_0) \mid x_t, u_t\right], \quad (12)$$

where $\hat{x}_0$ denotes the Tweedie prediction of the clean sample.

Let $\bar{x}_0$ denote the control-conditioned clean prediction: $\bar{x}_0 := \mathbb{E}[\hat{x}_0 \mid x_t, u_t] \approx \hat{x}_{0,\pi_0} + \kappa_t u_t$, where κ_t is the timestep-dependent Tweedie response coefficient, with $\kappa_t = \sigma_t^2/\alpha_t$ for VP diffusion and $\kappa_t = \sigma_t^2$ for VE diffusion. A first-order approximation around $\bar{x}_0$ gives $\mathbb{E}_{\pi_0}\left[\mathcal{L}_{\text{con}}^k(\hat{x}_0) \mid x_t, u_t\right] \approx \mathcal{L}_{\text{con}}^k(\bar{x}_0)$, so the trajectory-level control problem is reduced to the local surrogate

$$u_t^* = \arg\min_{u_t} \left\{ \frac{1}{2} g^2(t)\|u_t\|_2^2 \Delta t + \mathcal{L}_{\text{con}}^k\left(\hat{x}_{0,\pi_0} + \kappa_t u_t\right) \right\}. \quad (13)$$

For the exact objective gradient, the first-order optimality condition of Eq.13 is

$$g^2(t) u_t^* \Delta t + \kappa_t\left[\nabla_{\bar{x}_0} h(\bar{x}_0^*) + \rho\left(\bar{x}_0^* - \tilde{x}_0^k - z_0^k\right)\right] = 0, \quad (14)$$

where $\bar{x}_0^* = \hat{x}_{0,\pi_0} + \kappa_t u_t^*$. To accommodate existing guidance methods, we write the objective guidance in the general form $a_t(x)\,\widehat{\nabla_x h}(x)$, where $\widehat{\nabla_x h}(x)$ is an available estimate of the objective gradient and $a_t(x)$ sets its state- and timestep-dependent scale. For example, $a_t(x) = 1$, or $a_t(x) = \eta_t/(\|\widehat{\nabla_x h}(x)\|_2 + \varepsilon_g)$ correspond to raw and normalized guidance, respectively, with strength $\eta_t > 0$ and stability constant $\varepsilon_g > 0$.

Replacing the exact objective gradient in Eq. 14 gives the practical control rule

$$g^2(t) u_t^* \Delta t + \kappa_t\left[a_t(\bar{x}_0^*)\widehat{\nabla_{\bar{x}_0} h}(\bar{x}_0^*) + \rho\left(\bar{x}_0^* - \tilde{x}_0^k - z_0^k\right)\right] = 0, \quad (15)$$

which contains objective and consensus guidance. Once u_t^* is locally resolved, the state is advanced to $x_{t-\Delta t}$ through the discretized reverse dynamics. Further objective-gradient estimators and scaling rules are detailed in Appendix A.1.

Algorithm 1 CoSCoDiff

Require: Frozen prior s_θ, objective h, feasibility operator $\mathcal{P}_\mathcal{C}$, number of rollout rounds K, SDEdit restart time τ, consensus penalty ρ, variant $m \in \{\mathrm{P}, \mathrm{PD}\}$

$x_T^1 \sim p_T$
$x_0^1 \leftarrow \mathrm{Rollout}_\theta(x_T^1, T; h)$ ▷ Initial objective-guided rollout
$\tilde{x}_0^1 \leftarrow \mathcal{P}_\mathcal{C}(x_0^1)$; $z_0^1 \leftarrow 0$ ▷ Initialize feasible reference
for $k = 1, \ldots, K-1$ **do**
  $x_\tau^k \sim q_\tau(\cdot \mid \tilde{x}_0^k)$ ▷ SDEdit restart (Eq. 16)
  $\mathcal{L}_k(x) \leftarrow h(x) + \frac{\rho}{2}\|x - \tilde{x}_0^k - z_0^k\|_2^2$
  $x_0^{k+1} \leftarrow \mathrm{Rollout}_\theta(x_\tau^k, \tau; \mathcal{L}_k)$ ▷ Guided rollout (Eq. 8)
  $\tilde{x}_0^{k+1} \leftarrow \mathcal{P}_\mathcal{C}(x_0^{k+1} - z_0^k)$ ▷ Feasibility update (Eq. 9)
  $z_0^{k+1} \leftarrow \mathbf{1}_{\{m=\mathrm{PD}\}}\left(z_0^k + \tilde{x}_0^{k+1} - x_0^{k+1}\right)$ ▷ Coordination update (Eq. 10)
end for
return $\tilde{x}_0^K$

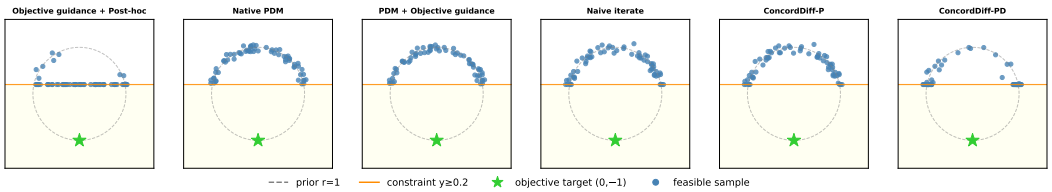

Figure 1: 2D toy experiment. The dashed circle denotes the prior manifold, the orange line the constraint boundary, and the green star the objective target. Blue points denote feasible samples.

Step 2: Feasibility Update (Eq.9). The auxiliary clean sample $\tilde{x}_0$ is updated by applying $\mathcal{P}_\mathcal{C}$ to the shifted terminal sample $x_0^{k+1} - z_0^k$. Depending on the constraint structure, $\mathcal{P}_\mathcal{C}$ may correspond to an exact Euclidean projection or an approximate feasibility correction implemented by an iterative or domain-specific solver. In the basic terminal-only realization, the operator is evaluated once per rollout at $t = 0$, avoiding the per-step projection overhead of PDM (Christopher et al., 2024). Its feasibility information still reaches intermediate noisy states through the consensus term in Step 1. Further realizations of $\mathcal{P}_\mathcal{C}$ are detailed in Appendix A.2.

Step 3: Coordination Update (Eq.10). Finally, CoSCoDiff-P discards historical consensus discrepancies, whereas CoSCoDiff-PD accumulates them through the scaled dual variable.

SDEdit Warm Restart. To facilitate iterative refinement across ADMM iterations, we initialize the rollout from the latest feasible sample, following the forward noising procedure of SDEdit (Meng et al., 2022). At outer iteration k, we perturb $\tilde{x}_0^k$ to an intermediate noise level t_r:

$$x_{t_r}^{k+1} = \alpha_{t_r}\tilde{x}_0^{k+1} + \sigma_{t_r}\epsilon, \qquad \epsilon \sim \mathcal{N}(0, I). \tag{16}$$

The resulting $x_{t_r}^{k+1}$ initializes the next controlled reverse rollout. The warm restart preserves information accumulated in the previous feasible iterate while reintroducing stochasticity to enable further objective optimization.

6 Experimental Results

We evaluate CoSCoDiff on four tasks: a toy problem, trajectory planning, physics-informed video frame generation, and text-conditioned human motion generation. We compare against objective guidance with post-hoc projection (Obj.+Post-hoc), PDM (Christopher et al., 2024), PDM with objective guidance (PDM+Obj.), Naive Iterate (CoSCoDiff without consensus splitting), and task-specific baselines. Shared hyperparameters are matched across methods within each task, including between the P and PD variants. We also show that existing objective-guidance methods can be plugged into CoSCoDiff without modification. Ablation studies are provided in Appendix D.

Toy Ring Experiment. To isolate the core mechanism of CoSCoDiff from task-specific complexities, we construct a 2D toy problem using a pre-trained diffusion model concentrated on the unit circle $x^2 + y^2 = 1$. For the clean sample $x_0 = (x, y)$, we define

$$h(x_0) = \|x_0 - (0, -1)\|_2, \qquad \mathcal{C} = \{(x, y) \mid y \geq 0.2\}.$$

The objective target $(0, -1)$ lies outside the feasible region, creating an explicit conflict between objective optimization and constraint satisfaction. Under the unit-circle prior and the constraint $y \geq 0.2$, the objective is minimized at $\mathcal{X}^\star = \{(\pm\sqrt{0.96}, 0.2)\}$. We measure proximity to these constrained optima by $d_{\text{opt}} = \mathbb{E}_x[\min_{x^\star \in \mathcal{X}^\star} \|x - x^\star\|_2]$. Fig. 1 shows that CoSCoDiff-PD concentrates feasible samples near the constrained optima while staying on the unit-circle prior. Table 1 confirms this. All methods satisfy the constraint, and CoSCoDiff-PD achieves the smallest d_{opt}.

Trajectory Planning. The 2D trajectory planning aims to minimize path length while avoiding collisions with obstacles. Diffusion models provide trajectory priors for motion planning (Power et al., 2023; Christopher et al., 2024). Following the experimental setting of Christopher et al. (2024), we use an unconditional diffusion model trained on demonstration trajectories generated by classical planners in EnvSimple2D (Carvalho et al., 2023). A clean trajectory is represented by its position waypoints, $x_0 = (p_0, p_1, \ldots, p_N)$, $p_i \in \mathbb{R}^2$, with fixed start and goal positions. We minimize the total path length subject to the collision-free constraint

Table 1: Quantitative results on the toy experiment. All methods achieve 100% feasibility.

Method		$d_{\text{opt}} \downarrow$
Obj. + Post-hoc		0.4973 ± 0.0132
PDM	Native	0.6147 ± 0.0182
	+ Obj.	0.5622 ± 0.0174
Naive Iterate		0.4979 ± 0.0168
CoSCoDiff	-P	0.4928 ± 0.0166
	-PD	$\mathbf{0.2130 \pm 0.0093}$

$$h(x_0) = \lambda_{\text{len}} \sum_{i=0}^{N-1} \|p_{i+1} - p_i\|_2^2 + \lambda_{\text{sm}} \sum_{i=1}^{N-1} \left(\sqrt{\|p_{i+1} - 2p_i + p_{i-1}\|_2^2 + \epsilon^2} - \epsilon \right), \tag{17}$$
$$\mathcal{C} = \{x_0 : [p_i, p_{i+1}] \cap \mathcal{O} = \varnothing,\ \forall i\}.$$

where $\mathcal{O}$ denotes the obstacle region. The first term penalizes squared distances between consecutive waypoints, while the second encourages trajectory smoothness. We evaluate CoSCoDiff using primal–dual coordination against diffusion-based baselines in the in-distribution environment EnvSimple2D and the out-of-distribution environment EnvDense2D. Feasibility and path length results are reported in Table 2. Detailed experimental settings are provided in the Appendix C.2. Table 2 shows that both CoSCoDiff variants achieve full feasibility on EnvSimple2D, with CoSCoDiff-P attaining the lowest mean path length. On EnvDense2D, CoSCoDiff-P achieves the highest observed feasibility and the lowest mean feasible path length. Since feasible subsets differ across methods, these conditional length averages should be interpreted alongside feasibility.

Table 2: Trajectory planning results (Mean $\pm$ standard deviation). Path length is averaged over feasible trajectories only. A dash indicates no feasible trajectory.

Method		EnvSimple2D		EnvDense2D	
		Feas. (%) $\uparrow$	Length $\downarrow$	Feas. (%) $\uparrow$	Length $\downarrow$
MPD		8.00 ± 3.80	2.4397 ± 0.1406	0.00 ± 0.00	–
Joint Penalty		71.33 ± 7.67	2.9185 ± 0.1730	0.00 ± 0.00	–
PDM	Native	78.00 ± 6.06	3.1112 ± 0.1373	10.00 ± 7.07	8.6561 ± 4.3597
	+ Obj.	79.33 ± 6.41	1.8200 ± 0.0488	72.00 ± 10.95	2.9886 ± 0.3328
Naive Iterate		$\mathbf{100.00 \pm 0.00}$	1.6586 ± 0.0142	82.00 ± 8.37	2.0557 ± 0.0780
CoSCoDiff	-P	$\mathbf{100.00 \pm 0.00}$	$\mathbf{1.6248 \pm 0.0050}$	$\mathbf{84.00 \pm 5.48}$	$\mathbf{2.0319 \pm 0.1028}$
	-PD	$\mathbf{100.00 \pm 0.00}$	1.6736 ± 0.0124	76.00 ± 11.40	2.2183 ± 0.1312

Physics-informed Video Frame Generation. The physics-informed video frame generation aims to generate a sequence depicting a target ball-sport category while ensuring that the ball follows a physically consistent trajectory under gravity. We use an unconditional noise-conditional score network (NCSN) (Song & Ermon, 2020) as the frame-level image prior, together with a pretrained two-head classifier for scene and ball semantics. A clean video sequence is represented as $x_0 = (I_0, \ldots, I_{T-1})$, with $I_k \in \mathbb{R}^{H \times W \times 3}$. Given a target category $y^\star$, let V_{s} and V_{b} denote the scene and ball views used by the classifier, and let $q_k = q(I_k) \in \mathbb{R}^2$ denote the extracted ball center at time t_k. We define the objective combining semantic and temporal consistency, together with the physical feasibility constraint, as

$$h(x_0; y^\star) = -\frac{1}{T} \sum_{k=0}^{T-1} \sum_{r \in \{\text{s},\text{b}\}} \lambda_r \log p_\phi^r(y^\star \mid V_r(I_k)) + \frac{\lambda_{\text{temp}}}{T-1} \sum_{k=0}^{T-2} \|M_{k,k+1}^{\text{bg}} \odot (I_{k+1} - I_k)\|_F^2,$$
$$\mathcal{C} = \left\{x_0 : \exists\, a, v \in \mathbb{R}^2,\ q_k = a + v t_k + \tfrac{1}{2} g_{\text{img}} t_k^2,\ \forall k\right\}, \tag{18}$$

Table 3: Physics-informed video frame generation results (mean ± standard deviation) on football.

Method		Physical trajectory error		Soft-objective performance	
		SSE (px^2) ↓	RMSE (px) ↓	Semantic cost ↓	Background inconsistency (10^{-3}) ↓
Augmented-DPS + Post-hoc		38.5680 ± 28.3040	2.1860 ± 1.0470	$\mathbf{0.0417 \pm 0.0036}$	9.2560 ± 0.3297
Naive Iterate		12.6650 ± 3.4470	1.3360 ± 0.1900	0.0521 ± 0.0028	2.1597 ± 0.0413
CoSCoDiff	-P	10.3480 ± 2.0490	1.2120 ± 0.1220	0.0504 ± 0.0020	2.1305 ± 0.0260
	-PD	$\mathbf{10.0330 \pm 1.6350}$	$\mathbf{1.1940 \pm 0.1000}$	0.0502 ± 0.0018	$\mathbf{2.1299 \pm 0.0256}$

where $g_{\text{img}} \in \mathbb{R}^2$ denotes the prescribed gravitational acceleration in image coordinates, and $M^{\text{bg}}_{k,k+1}$ masks out the ball regions in adjacent frames. The first objective term promotes scene- and object-level semantic consistency, while the second encourages background temporal consistency. The constraint $\mathcal{C}$ enforces a ballistic trajectory across frames.

In practice, physical consistency is evaluated using the extracted ball positions and a numerical tolerance. The projection operator adjusts the ball trajectory toward this constraint set. We instantiate the objective guidance using a augmented-DPS (Chung et al., 2022) and compare CoSCoDiff against augmented-DPS with a single post-hoc feasibility correction and naive iterative refinement. All methods share the same classifier, semantic-guidance configuration, and physical feasibility operator. Detailed experimental settings are provided in the Appendix C.3.

Figure 2 shows qualitative results across four ball-sport categories. Table 3 further shows that CoSCoDiff-PD achieves the lowest trajectory error and background inconsistency. Semantic cost is computed from the frozen guidance classifier using scene and ball negative log-probabilities, while background inconsistency measures RGB deviation from the temporal median outside detected ball regions; both therefore evaluate task-specific objectives rather than independent perceptual quality. Although Augmented-DPS + Post-hoc attains a lower semantic cost, its projected samples exhibit incomplete ball boundaries and residual artifacts. PDM variants fail to complete sampling after losing valid ball localization in the tested configurations. A FreeDOM-guided (Yu et al., 2023) example in Appendix C.3 further demonstrates compatibility with alternative objective guidance.

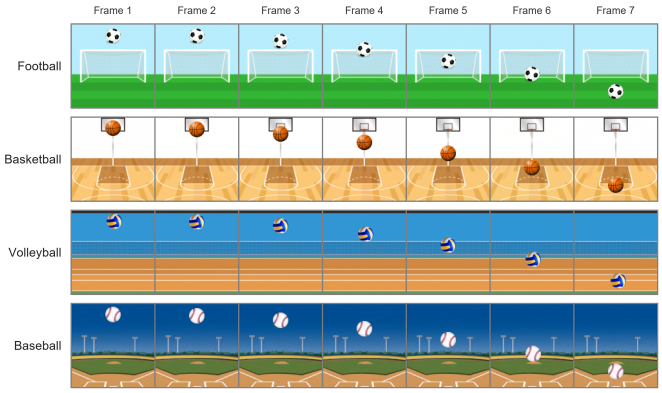

Figure 2: CoSCoDiff-PD on four ball-sport categories.

Text-Conditioned Human Motion Generation. The text-conditioned human motion generation aims to generate motions consistent with a textual description while following a prescribed root trajectory. We use a pre-trained GMD model (Karunratanakul et al., 2023) on HumanML3D as the motion prior. A clean sample is represented as $x_0 = (m_1, \ldots, m_T)$ with $m_t \in \mathbb{R}^{263}$. Given a text prompt c, let E_{motion} and E_{text} denote the frozen motion and text encoders of the T2M evaluator, and let $r_t(x_0) \in \mathbb{R}^2$ and $\bar{r}_t$ denote the generated and prescribed root positions on the ground plane, respectively. We define the semantic objective and feasibility constraint as

Table 4: Text-conditioned human motion generation with $\varepsilon_m = 0.02\,\text{m}$ on the 100-case benchmark.

Method		T2M Dist. ↓	Feas. (%) ↑
GMD	Native	4.018 ± 1.363	0.00
	+ Post-hoc	4.368 ± 0.886	100.00
ADMMDiff	Native	5.326 ± 1.671	0.00
	+ Post-hoc	5.522 ± 1.715	100.00
LGD-MC	Native	3.845 ± 1.508	0.00
	+ Post-hoc	4.871 ± 1.415	100.00
Naive Iterate		1.727 ± 0.625	100.00
CoSCoDiff	-P	$\mathbf{1.722 \pm 0.618}$	100.00
	-PD	1.972 ± 1.061	100.00

$$h(x_0; c) = \|E_{\text{motion}}(x_0) - E_{\text{text}}(c)\|_2^2, \qquad \mathcal{C}_\varepsilon = \{x_0 : \|r_t(x_0) - \bar{r}_t\|_2 \le \varepsilon,\ \forall t\}, \tag{19}$$

where $\varepsilon = 0.02\,\text{m}$. Generated motions are converted to the evaluator's input representation before computing the text–motion distance.

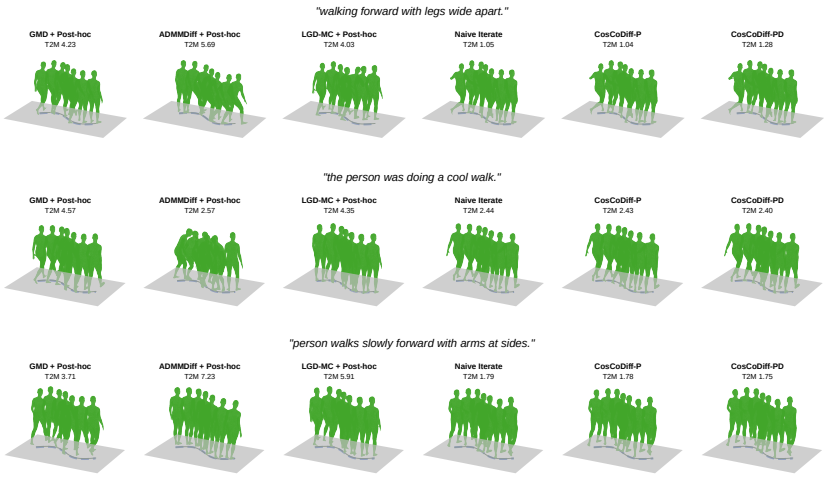

Figure 3: Text-conditioned motion comparisons. All outputs satisfy the $2\,\mathrm{cm}$ constraint. Blue dashes and dark green denote target and generated root trajectories. T2M distance is lower-is-better.

We compare CoSCoDiff-P and CoSCoDiff-PD with GMD, ADMMDiff (Zhang et al., 2025), LGD-MC (Song et al., 2023), and Naive Iterate. The main table reports native T2M distance and final root-trajectory error. For GMD, ADMMDiff, and LGD-MC, we also report post-hoc projection variants using the same root-trajectory constraint operator. Additional comparisons, including full-trajectory inpainting for GMD and T2M-guided variants of ADMMDiff, are provided in the Appendix C.4.

Table 4 shows that CosCoDiff-P achieves 100% feasibility and the lowest mean T2M distance, substantially outperforming post-hoc baselines on T2M. Figure 3 shows selected paired examples across three prompts, with CosCoDiff achieving the lowest T2M distance in each row. Beyond the default objective guidance realization, Table 5 shows that integrating LGD-MC into CosCoDiff also achieves full feasibility with lower mean T2M distance than native LGD-MC and its post-hoc variant, demonstrating compatibility with LGD-MC.

Table 5: LGD-MC compatibility with $\varepsilon_m = 0.02\,\mathrm{m}$.

Method	T2M Dist. ↓	Feas. (%) ↑
LGD-MC	3.845 ± 1.508	0
LGD-MC + Post-hoc	4.871 ± 1.415	**100**
CoSCoDiff-P + LGD-MC	$\mathbf{3.401 \pm 1.343}$	**100**

Ablation Ablations confirm the effectiveness of the key components of CoSCoDiff. On the toy ring problem, SDEdit restarts preserve feasibility corrections across rollouts, while combining them with PD coordination yields the strongest concentration near the constrained optima. Across tasks, both P and PD provide effective coordination, with PD benefiting more from accurate feasibility updates and P remaining more robust to approximate operators. Sensitivity studies further show how ρ, the restart level, and the number of outer iterations affect the coordination behavior. Detailed results are provided in Appendix D.

7 Conclusion

We proposed CoSCoDiff, a training-free consensus-splitting framework that separates task objectives from feasibility constraints and coordinates them during diffusion sampling. Through terminal-state splitting and Tweedie-based sequential rollout, CoSCoDiff propagates objective and feasibility information throughout denoising while preserving explicit constraint enforcement. Experiments across four heterogeneous tasks demonstrate effective objective–constraint coordination without retraining the underlying diffusion models. Current limitations include the cost of multiple iterations and sensitivity to approximate feasibility operators. Future work will study more efficient coordination and better robust performance across iterations under approximate updates.

AI USE STATEMENT

We used generative AI tools to improve the language and readability of selected passages in the manuscript. The authors reviewed and edited the AI-assisted text and take full responsibility for the final content of this work.

ETHICS STATEMENT

This work studies controllable generation and planning using pretrained diffusion models. Potential concerns include biases inherited from these models and misuse of generated visual or human-motion content. Satisfaction of specified computational constraints should not be interpreted as a guarantee of real-world safety. Applications in physical or safety-critical settings require independent validation.

REPRODUCIBILITY STATEMENT

The sampling procedure is described in Sec. 5, with additional implementation details in Appendices A.1 and A.2. The submission includes source code for the general sampling framework and scripts for the toy experiment. Task-specific interfaces and scripts for the remaining main experiments are not included in the current code supplement and will be publicly released after publication.

REFERENCES

Arpit Bansal, Hong-Min Chu, Avi Schwarzschild, Soumyadip Sengupta, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Universal guidance for diffusion models, 2023. URL https://arxiv.org/abs/2302.07121.

João Carvalho, An T. Le, Mark Baierl, Dorothea Koert, and Jan Peters. Motion planning diffusion: Learning and planning of robot motions with diffusion models. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1916–1923, 2023. doi: 10.1109/IROS55552.2023.10342382.

Stanley H. Chan, Xiran Wang, and Omar A. Elgendy. Plug-and-play admm for image restoration: Fixed-point convergence and applications. *IEEE Transactions on Computational Imaging*, 3(1): 84–98, 2017. doi: 10.1109/TCI.2016.2629286.

Jinho Chang, Jaemin Kim, and Jong Chul YE. Training-free reward-guided image editing via trajectory optimal control. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 26375–26396, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/2c3b636b64ca1dfdae3e096e4deeaa42-Paper-Conference.pdf.

Jacob K Christopher, Stephen Baek, and Ferdinando Fioretto. Constrained synthesis with projected diffusion models, 2024. URL https://arxiv.org/abs/2402.03559.

Hyungjin Chung, Jeongsol Kim, Michael T Mccann, Marc L Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*, 2022.

Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021. URL https://arxiv.org/abs/2105.05233.

Chenhe Du, Xuanyu Tian, Qing Wu, Muyu Liu, Jingyi Yu, Hongjiang Wei, and Yuyao Zhang. Plug-and-play diffusion meets admm: Dual-variable coupling for robust medical image reconstruction. *arXiv preprint arXiv:2602.23214*, 2026.

Nic Fishman, Leo Klarner, Emile Mathieu, Michael Hutchinson, and Valentin De Bortoli. Metropolis sampling for constrained diffusion models. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 62296–62331. Curran Associates, Inc., 2023. doi: 10.52202/075280-2721.

Nic Fishman, Leo Klarner, Valentin De Bortoli, Emile Mathieu, and Michael Hutchinson. Diffusion models for constrained domains, 2024. URL https://arxiv.org/abs/2304.05364.

Daniel Gabay and Bertrand Mercier. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & mathematics with applications*, 2(1): 17–40, 1976.

Paolo Giaretta, Zeyang Li, and Navid Azizan. Direct: Safe diffusion-based planning via receding-horizon denoising. *arXiv preprint arXiv:2606.15359*, 2026.

Glowinski, R. and Marroco, A. Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires. *R.A.I.R.O. Analyse Numérique*, 9:41–76, 1975. doi: 10.1051/m2an/197509R200411. URL https://doi.org/10.1051/m2an/197509R200411.

Yingqing Guo, Hui Yuan, Yukang Yang, Minshuo Chen, and Mengdi Wang. Gradient guidance for diffusion models: An optimization perspective. *Advances in neural information processing systems*, 37:90736–90770, 2024.

Yutong He, Naoki Murata, Chieh-Hsin Lai, Yuhta Takida, Toshimitsu Uesaka, Dongjun Kim, Wei-Hsiang Liao, Yuki Mitsufuji, J. Zico Kolter, Ruslan Salakhutdinov, and Stefano Ermon. Manifold preserving guided diffusion, 2023. URL https://arxiv.org/abs/2311.16424.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022. URL https://arxiv.org/abs/2207.12598.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in neural information processing systems*, 35:8633–8646, 2022.

Lianghua Huang, Di Chen, Yu Liu, Yujun Shen, Deli Zhao, and Jingren Zhou. Composer: Creative and controllable image synthesis with composable conditions. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 13753–13773. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/huang23b.html.

Michael Janner, Yilun Du, Joshua Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 9902–9915. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/janner22a.html.

Ioannis Karatzas and Steven E. Shreve. *Brownian Motion and Stochastic Calculus*, volume 113 of *Graduate Texts in Mathematics*. Springer, New York, NY, 2 edition, 1998. ISBN 978-0-387-97655-6. doi: 10.1007/978-1-4612-0949-2.

Korrawe Karunratanakul, Konpat Preechakul, Supasorn Suwajanakorn, and Siyu Tang. Guided motion diffusion for controllable human motion synthesis. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2151–2162. IEEE, 2023.

Shervin Khalafi, Dongsheng Ding, and Alejandro Ribeiro. Constrained diffusion models via dual training. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 26543–26576. Curran Associates, Inc., 2024. doi: 10.52202/079017-0836.

Sunwoo Kim, Minkyu Kim, and Dongmin Park. Test-time alignment of diffusion models without reward over-optimization. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 87238–87272, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/d9042abf40782fbce28901c1c9c0e8d8-Paper-Conference.pdf.

Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22511–22521, June 2023.

Guan-Horng Liu, Tianrong Chen, Evangelos Theodorou, and Molei Tao. Mirror diffusion models for constrained and watermarked generation. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 42898–42917. Curran Associates, Inc., 2023a. doi: 10.52202/075280-1858.

Qingyang Liu, Junqi You, Jianting Wang, Xinhao Tao, Bo Zhang, and Li Niu. Shadow generation for composite image using diffusion model. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8121–8130. IEEE, 2024.

Xingchao Liu, Lemeng Wu, Mao Ye, and qiang liu. Learning diffusion bridges on constrained domains. In *The Eleventh International Conference on Learning Representations*, 2023b. URL https://openreview.net/forum?id=WH1yCa0TbB.

Aaron Lou and Stefano Ermon. Reflected diffusion models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 22675–22701. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/lou23a.html.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations, 2022. URL https://arxiv.org/abs/2108.01073.

Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models, 2023. URL https://arxiv.org/abs/2302.08453.

Thomas Power, Rana Soltani-Zarrin, Soshi Iba, and Dmitry Berenson. Sampling constrained trajectories using composable diffusion models. In *Iros 2023 workshop on differentiable probabilistic robotics: Emerging perspectives on robot learning*, 2023.

Omer Rochman-Sharabi and Gilles Louppe. Predict-project-renoise: Sampling diffusion models under hard constraints. *arXiv preprint arXiv:2601.21033*, 2026.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 10674–10685. ieee, 2022a.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022b. URL https://arxiv.org/abs/2112.10752.

Yifei Shen, Xinyang Jiang, Yezhen Wang, Yifan Yang, Dongqi Han, and Dongsheng Li. Understanding and improving training-free loss-based diffusion guidance, 2024. URL https://arxiv.org/abs/2403.12404.

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 Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 32483–32498. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/song23k.html.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution, 2020. URL https://arxiv.org/abs/1907.05600.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2021. URL https://arxiv.org/abs/2011.13456.

Joseph L Watson, David Juergens, Nathaniel R Bennett, Brian L Trippe, Jason Yim, Helen E Eisenach, Woody Ahern, Andrew J Borst, Robert J Ragotte, Lukas F Milles, et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976):1089–1100, 2023.

Suttisak Wizadwongsa and Supasorn Suwajanakorn. Accelerating guided diffusion sampling with splitting numerical methods. *arXiv preprint arXiv:2301.11558*, 2023.

Haotian Ye, Haowei Lin, Jiaqi Han, Minkai Xu, Sheng Liu, Yitao Liang, Jianzhu Ma, James Zou, and Stefano Ermon. Tfg: Unified training-free guidance for diffusion models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 22370–22417. Curran Associates, Inc., 2024. doi: 10.52202/079017-0704. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/2818054fc6de6dacdda0f142a3475933-Paper-Conference.pdf.

Jiwen Yu, Yinhuai Wang, Chen Zhao, Bernard Ghanem, and Jian Zhang. Freedom: Training-free energy-guided conditional diffusion model, 2023. URL https://arxiv.org/abs/2303.09833.

Stefano Zampini, Jacob K. Christopher, Luca Oneto, Davide Anguita, and Ferdinando Fioretto. Training-free constrained generation with stable diffusion models, 2025. URL https://arxiv.org/abs/2502.05625.

Jichen Zhang, Liqun Zhao, Antonis Papachristodoulou, and Jack Umenberger. Constrained diffusers for safe planning and control. *Advances in Neural Information Processing Systems*, 38:34965–34998, 2026.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3836–3847, October 2023.

Youyuan Zhang, Zehua Liu, Zenan Li, Zhaoyu Li, James J Clark, and Xujie Si. Decoupling training-free guided diffusion by admm. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23292–23302. IEEE, 2025.

Yuanzhi Zhu, Kai Zhang, Jingyun Liang, Jiezhang Cao, Bihan Wen, Radu Timofte, and Luc Van Gool. Denoising diffusion models for plug-and-play image restoration. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1219–1229. IEEE, 2023.

A Implementation Details

A.1 Objective Guidance

CoSCoDiff represents the task-objective guidance in the general form

$$a_t(x)\,\widehat{\nabla_x h}(x), \tag{20}$$

where $\widehat{\nabla_x h}(x)$ denotes an available estimate of the objective gradient and $a_t(x)$ controls its state- and timestep-dependent magnitude. This representation accommodates different objective realizations without modifying the consensus splitting procedure.

Direct-gradient objectives. When the task objective $h(x)$ is explicitly defined and differentiable, its gradient can be obtained analytically or through automatic differentiation:

$$\widehat{\nabla_x h}(x) = \nabla_x h(x). \tag{21}$$

This setting applies, for example, to differentiable geometric or trajectory-level objectives such as path length and smoothness costs.

Scorer-based objectives. For semantic or perceptual objectives defined through a pretrained scorer S_ϕ, we construct

$$h(x) = \ell(S_\phi(x)), \tag{22}$$

where ℓ converts the scorer output into a minimization objective. The corresponding guidance can be obtained by back-propagating through the scorer or by using existing diffusion-guidance estimators, such as DPS-style guidance based on denoised clean-state predictions (Chung et al., 2022) or Monte Carlo loss-guidance estimators (Song et al., 2023). From the perspective of CoSCoDiff, these methods simply provide different realizations of $\widehat{\nabla_x h}(x)$.

The scaling function $a_t(x)$ further controls the guidance strength. The choice

$$a_t(x) = 1 \tag{23}$$

recovers the raw gradient, while

$$a_t(x) = \frac{\eta_t}{\|\widehat{\nabla_x h}(x)\|_2 + \varepsilon_g} \tag{24}$$

yields normalized guidance, where $\eta_t > 0$ controls the guidance magnitude and $\varepsilon_g > 0$ ensures numerical stability. Other task-specific scaling rules can be incorporated in the same form.

A.2 Constraint Feasibility Operators

The hard constraint is specified by a feasible set $\mathcal{C}$, while its algorithmic realization is abstracted by the feasibility operator $\mathcal{P}_\mathcal{C}$. Given an input v, the operator returns a sample satisfying, or moving toward satisfaction of, the task-specific constraint.

Exact projection. When the Euclidean projection onto $\mathcal{C}$ is tractable,

$$\mathcal{P}_\mathcal{C}(v) = \Pi_\mathcal{C}(v) \in \arg\min_{x \in \mathcal{C}} \frac{1}{2}\|x - v\|_2^2. \tag{25}$$

This recovers the exact feasibility update associated with the indicator function $I_\mathcal{C}$. For example, the half-plane constraint in the toy experiment admits a closed-form Euclidean projection.

Iterative or domain-specific correction. For complex constraints where the exact projection is unavailable or computationally expensive, $\mathcal{P}_\mathcal{C}$ can instead be implemented by an iterative numerical solver or a domain-specific feasibility correction. Such procedures may internally exploit constraint gradients when available, but CoSCoDiff does not require the constraint itself to provide a differentiable guidance loss. Examples include SDF-based geometric correction in trajectory planning and task-specific feasibility correction in high-dimensional generation problems.

Importantly, the feasibility operator is applied to the terminal clean sample rather than directly supplying an additional constraint gradient to every diffusion step. Its output $\tilde{x}_0^k$ enters the consensus term

$$\frac{\rho}{2}\left\|\hat{x}_0(x_t, t) - \tilde{x}_0^k - z_0^k\right\|_2^2, \tag{26}$$

allowing the resulting feasibility information to influence intermediate noisy states through Tweedie prediction even when $\mathcal{P}_\mathcal{C}$ itself is non-differentiable.

B PRIMAL AND PRIMAL–DUAL COORDINATION

The two CoSCoDiff variants defined in Eq.10 differ in whether consensus discrepancies are retained across outer iterations. COSCODIFF-P discards the historical residual and coordinates each rollout only with the most recent feasible iterate, whereas COSCODIFF-PD accumulates the residual through a scaled dual variable and therefore retains historical consensus information.

To characterize this difference, let $r^{k+1} = \tilde{x}_0^{k+1} - x_0^{k+1}$ denote the terminal consensus residual. For COSCODIFF-PD, the dual variable satisfies $z_0^K = z_0^0 + \sum_{k=0}^{K-1} r^{k+1}$. Hence, primal–dual coordination retains a history of the disagreement between objective-guided diffusion and feasibility correction. When these residuals provide persistent and informative consensus signals, such memory can improve coordination across iterations. However, the same accumulation can also retain errors introduced by approximate or stochastic updates.

This issue is particularly relevant in diffusion sampling. Unlike the idealized splitting setting, the primal subproblem in CoSCoDiff is approximately solved through sequential rollout and Tweedie prediction, while $\mathcal{P}_{\mathcal{C}}$ may correspond to an iterative or domain-specific feasibility operator rather than an exact proximal projection. SDEdit warm restarts further introduce stochastic variation between outer iterations. Therefore, the classical convergence behavior of exact ADMM does not directly transfer to the practical COSCODIFF-PD iteration. The accumulated dual feedback may encode both useful consensus information and approximation or operator mismatch.

In contrast, COSCODIFF-P removes historical residual memory and retains only the current quadratic attraction toward the latest feasible iterate. We therefore regard primal-only and primal–dual coordination as two complementary realizations of the same consensus-splitting framework, rather than assuming that dual accumulation is uniformly beneficial. We report both variants across our benchmarks and empirically analyze their behavior under different feasibility operators.

C DETAILED EXPERIMENTAL SETTINGS

C.1 TOY RING EXPERIMENT

Implementation and Evaluation. We use a frozen score network pretrained by denoising score matching on samples drawn uniformly from the unit circle. The network parameterizes a radial score using a multilayer perceptron that takes the radius and log noise level as inputs and predicts the radial score magnitude. It has three hidden layers of width 64 with SiLU activations. The main quantitative evaluation uses an NVIDIA GeForce RTX 4060 Laptop GPU. All methods share the same pretrained model and evaluation seeds. We report mean $\pm$ standard deviation across 20 seeds, with 512 samples per seed. Evaluation measures feasibility, objective value, and distance to the constrained optimum set $\mathcal{X}^\star = \{(\pm\sqrt{0.96}, 0.2)\}$.

Objective guidance. We use the raw-gradient interface for $h(v) = \|v - (0, -1)\|_2$, evaluating its analytic gradient at the clean-sample estimate. The objective gradient is used without normalization, with $a_t = 1$. The resulting guidance is norm-clipped for numerical stability.

Constraint Operator. We use the exact Euclidean projection onto the half-space, $\mathcal{P}_{\mathcal{C}}(x, y) = (x, \max\{y, 0.2\})$. This closed-form operator requires no iterative solver or constraint surrogate and guarantees feasibility of the corrected samples.

Hyperparameters. We use 10 geometrically spaced noise levels from 10 to 0.005, with 10 Langevin updates per visited level. Both CoSCoDiff variants use $(\rho, K, \tau_{\mathrm{idx}}) = (5, 15, 6)$ and an objective weight of 1. Iterative methods return the final corrected samples. Hyperparameter sensitivity is examined in Appendix D.

C.2 TRAJECTORY PLANNING

Implementation and Evaluation. We evaluate on EnvSimple2D and EnvDense2D, representing in-distribution and out-of-distribution planning, respectively. Both environments use the same

frozen diffusion model pretrained on EnvSimple2D. The benchmarks contain 30 and 10 fixed start–goal conditions, respectively, selected by difficulty-stratified sampling. All methods share these conditions and sampling seeds, with five repeated runs. Primary experiments use an NVIDIA L40S GPU. Final feasibility is assessed using a common segment–obstacle collision evaluator. We report mean $\pm$ standard deviation across repeated runs, with geometric path length averaged over feasible trajectories within each run.

Objective guidance. The objective combines a squared waypoint-displacement term with a smoothness penalty on second-order waypoint differences. We use its gradient to guide diffusion toward shorter, smoother trajectories. CoSCoDiff additionally incorporates the consensus penalty: CoSCoDiff-P uses no dual accumulation, whereas CoSCoDiff-PD maintains dual memory. Naive Iterate removes both the consensus penalty and dual accumulation.

Constraint Operator. For Naive Iterate and CoSCoDiff, we use SDF-based geometric corrections while keeping endpoints fixed. Waypoints and segment midpoints are checked during sampling. A weak correction is applied after each noise level, followed by a stronger correction at the end of each rollout. On EnvDense2D, both PDM variants also use SDF-based correction. We additionally apply a common terminal SDF correction to the final outputs of Naive Iterate, both CoSCoDiff variants, and both PDM variants, using denser segment checks. These finite-budget corrections are approximate and do not guarantee collision-free trajectories.

Hyperparameters. We use 10 noise levels with 100 Langevin updates per level. Iterative methods perform $K = 15$ outer iterations and return the final corrected trajectory. Both CoSCoDiff variants use $(\rho, \tau_{\mathrm{idx}}) = (20, 3)$ on EnvSimple2D and $(10, 5)$ on EnvDense2D. The objective weights are $\lambda_{\mathrm{len}} = 1000$ and $\lambda_{\mathrm{sm}} = 5$. The additional terminal correction on EnvDense2D uses at most 500 iterations, nine interior points per segment, and a clearance margin of 0.005.

C.3 Physics-informed Video Frame Generation

Implementation and evaluation. We consider a controlled synthetic sports-video benchmark with four ball-sport categories, where the generated sequence should match a target category while depicting ballistic ball motion. Quantitative comparisons focus on football, while qualitative examples cover all four categories.

The diffusion prior is pretrained on NVIDIA A100 GPUs using a foreground-weighted denoising score-matching loss. Following the region-weighted loss design of Liu et al. (2024), we assign larger weights to the ball region to reduce background dominance. Semantic classifiers operate on full frames and ball-centered crops, with hard-negative training and an auxiliary objectness objective using negative examples. The diffusion prior and classifiers are frozen during generation. Sampling and evaluation are conducted on NVIDIA RTX A5000 GPUs.

Semantic consistency is evaluated at both the scene and object levels. Physical consistency is assessed by independently re-estimating ball centers from the generated frames and measuring their residual from a ballistic trajectory with prescribed gravity. Detection validity is reported alongside this residual. The football evaluation reference contains seven frames from a single template video; reference-based feature distances are therefore interpreted as template-similarity diagnostics, not distribution-level measures of generative quality.

Objective guidance. We use differentiable semantic scorers operating on full frames and ball-centered crops. Their target-category log probabilities provide scene-level and object-level guidance, respectively. A temporal consistency term penalizes changes between adjacent frames within their shared background region, excluding the moving ball.

Our Augmented-DPS baseline adapts DPS-style guidance to this combined objective, with noise-dependent per-frame gradient normalization and magnitude clipping. Gradients are evaluated with respect to a detached, range-clipped clean-frame estimate, without backpropagating through the score network. The baseline performs a single guided rollout without consensus coupling, dual updates, or outer SDEdit restarts. Augmented-DPS + Post-hoc additionally applies the same constraint operator once to the resulting sequence. Naive Iterate and CoSCoDiff use the same objective-guidance implementation within their outer iterations.

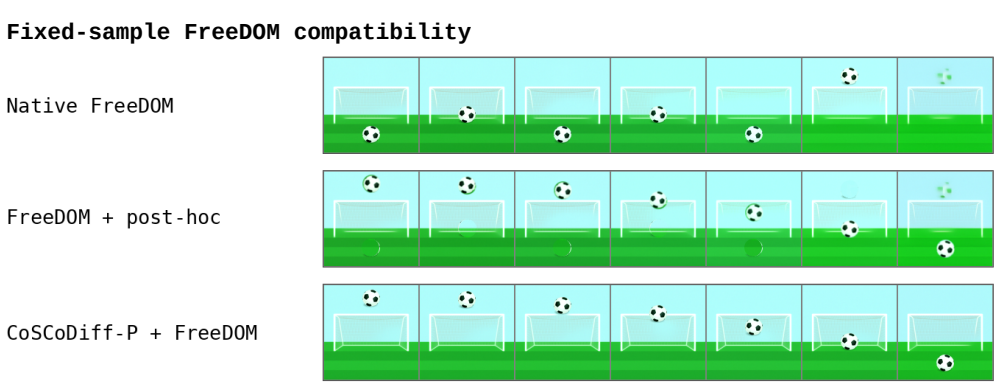

Figure 4: Fixed-sample compatibility comparison with FreeDOM. Each row shows the same seven-frame sample processed using native FreeDOM, FreeDOM with a single post-hoc projection, or CoSCoDiff-P with FreeDOM guidance, respectively. All methods share the identical initial FreeDOM tensor, and the constrained variants use the same physical feasibility operator.

Constraint operator. We estimate ball centers from the generated frames and fit a ballistic trajectory by optimizing its initial position and velocity while keeping image-space gravity fixed. A damped Gauss–Newton procedure adjusts spatial warps to align the detected motion with this trajectory. The operator transforms the generated image content and provides an approximate feasibility correction. Physical consistency is therefore evaluated on the corrected frames rather than inferred from the fitted target trajectory.

Hyperparameters. We use $K = 3$ outer updates with SDEdit restart index $\tau_{\text{idx}} = 5$ and 25 Langevin steps per noise level. The consensus weight is $\rho = 0.3$ with noise-level scaling; CoSCoDiff-PD uses a dual-update step size of 0.25. Refinement uses semantic guidance scale 0.2, preconditioning scale 0.5, and control clipping fraction 0.25. The temporal-consistency weight is 10, and the Gauss–Newton damping coefficient is 10^{-3}.

Compatibility with FreeDOM. We further evaluate whether CoSCoDiff can incorporate FreeDOM (Yu et al., 2023) through the same objective-guidance interface. Figure 4 presents a fixed-sample comparison among native FreeDOM, FreeDOM followed by a single post-hoc projection, and CoSCoDiff-P with FreeDOM guidance. All three variants start from exactly the same FreeDOM-generated video tensor, while the two constrained variants use the same physical feasibility operator and projection parameters. Native FreeDOM leaves the final frame without a recognizable ball. The post-hoc correction recovers the missing ball but introduces visible ghosting and duplicated ball content. By integrating the same FreeDOM guidance with feasibility correction over $K = 3$ iterations, CoSCoDiff-P produces recognizable footballs in all seven frames while maintaining a coherent ballistic trajectory; its final trajectory has 100% valid detections and an independently evaluated q_{eval} RMSE of 1.432 pixels. This fixed-sample experiment demonstrates compatibility with an alternative objective-guidance method and the benefit of iterative coordination over a single terminal correction; it is not intended as evidence of robustness across random initializations.

C.4 Text-Conditioned Human Motion Generation

Implementation and Evaluation. We use a frozen GMD model pretrained on HumanML3D, with motions represented by 263-dimensional features over 120 frames. Experiments are conducted on NVIDIA RTX A5000 GPUs. The benchmark combines three prescribed root trajectories (straight, arc, and S-curve) with five text conditions: wide walking, cool walking, sad walking, walking with arms at sides, and drunk walking. Each text–trajectory condition uses ten seeds for the arc and five seeds for each remaining trajectory, yielding 100 samples across 15 conditions. Methods share the same prompt, trajectory, and seed assignments. We report native T2M distance as mean $\pm$ standard deviation across all 100 samples, without filtering by feasibility. Native T2M evaluation converts generated motions through the skeleton representation into the evaluator's input features. A sample

is feasible when every frame satisfies the $0.02\,\mathrm{m}$ root-position tolerance, allowing a numerical tolerance of $10^{-5}\,\mathrm{m}$. Qualitative examples use matched seeds across methods and are selected based on paired T2M improvements.

Objective guidance. The semantic objective is the Euclidean distance between frozen T2M motion and text embeddings, evaluated on clean-motion estimates. This external objective supplements GMD's native text conditioning. CoSCoDiff additionally applies a root-space consensus penalty: CoSCoDiff-P uses no dual accumulation, whereas CoSCoDiff-PD maintains full-state dual memory. Naive Iterate uses semantic guidance and repeated feasibility correction without the consensus penalty or dual accumulation. For the compatibility experiment, we replace the point-estimate semantic guidance with an LGD-MC estimator. Native LGD-MC instead uses soft trajectory guidance together with GMD's native text conditioning.

Constraint Operator. We correct the physical root position at each frame by projecting it onto the radius-ε_m disk centered at the prescribed position. The corrected motion is re-encoded to maintain consistency between its trajectory and motion features, followed by a feasibility check. Although the root-space disk projection is exact, the complete correction and re-encoding procedure is not claimed to be a Euclidean projection in the full 263-dimensional motion representation. Naive Iterate and both CoSCoDiff variants apply this operator after each rollout and return the final corrected motion. Post-hoc baselines apply the same operator once to the saved native output, without regenerating the motion.

Hyperparameters. The main iterative comparisons use $K = 5$ outer iterations, $\rho = 3584$, and GMD text-conditioning scale 2.5. We retain the native VP diffusion schedule and use SDEdit restart index $\tau_{\mathrm{idx}} = 989$. Under our reverse-time indexing convention, $t_r = 999 - \tau_{\mathrm{idx}} = 10$ on the 1000-step schedule; larger indices therefore correspond to less restart noise. The initial and restart semantic-guidance scales are 1 and 524288, respectively, and the consensus-guidance budget ratio is 0.02. ADMMDiff uses its released trajectory-guided sampler with guidance scale 100. Native LGD-MC uses ten Monte Carlo samples, trajectory-loss scale 0.0025, and 100 DDIM steps with $\eta = 0$. The separate LGD-MC compatibility experiment uses $K = 4$, ten Monte Carlo samples, semantic-guidance scale 524288, and the same ρ, restart index, consensus budget, and feasibility tolerance.

D Ablation Study

We use the toy ring problem to examine the effects of SDEdit restarts and the hyperparameters ρ, K, and τ_{idx}. This setting provides an analytic optimum set,

$$\mathcal{X}^\star = \left\{(-\sqrt{0.96}, 0.2), (\sqrt{0.96}, 0.2)\right\},$$

allowing us to measure $d_{\mathrm{opt}} = \mathbb{E}_x[\min_{x^\star \in \mathcal{X}^\star} \|x - x^\star\|_2]$. This metric evaluates proximity to solutions that jointly respect the ring prior, the constraint, and the objective.

D.1 Effect of SDEdit Restarts

We compare four configurations: Naive Iterate and CoSCoDiff-PD, each with and without SDEdit. Without SDEdit, every outer iteration starts from fresh Gaussian noise and follows the full denoising schedule. CoSCoDiff-PD still retains its consensus anchor and dual state in this variant. With SDEdit, subsequent rollouts instead start from a re-noised version of the preceding corrected sample. All configurations share the same first rollout. Figure 5 shows the outputs of Step 1 and Step 2 at $K = 1, 5, 10, 15$.

Without SDEdit, Naive Iterate repeatedly produces objective-guided samples whose correction places substantial mass inside the ring along the constraint boundary. PD coordination changes this behavior, but restarting from fresh noise limits the preservation of corrected sample structure across iterations. With SDEdit, both methods better retain the feasible portion of the ring. Combining SDEdit with PD coordination further concentrates samples near the two constrained optima. These qualitative observations indicate that sample reuse and coordination play complementary roles. The

comparison uses the same outer-iteration budget; the full restarts without SDEdit visit more noise levels.

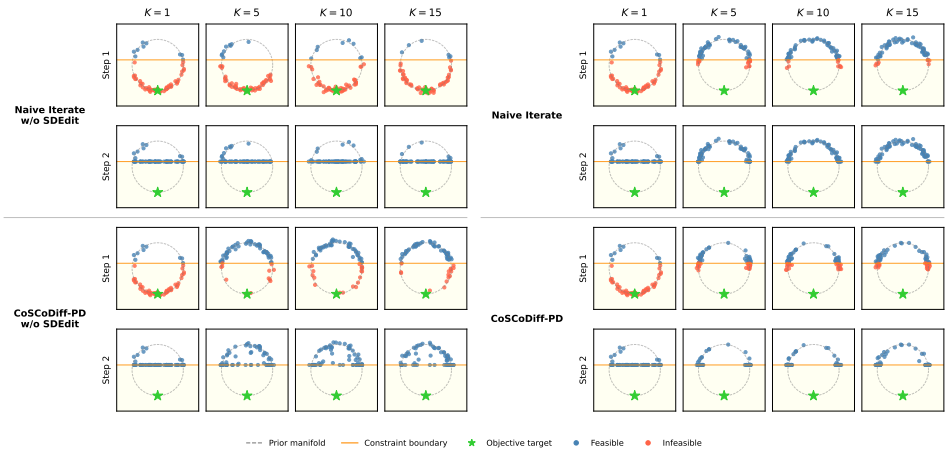

Figure 5: SDEdit ablation on the toy ring problem. Each method block shows guided-rollout outputs (Step 1) and corrected samples (Step 2) at $K = 1, 5, 10, 15$.

D.2 Primal and Primal–Dual Coordination

We use the P/PD comparisons already reported in the main experiments, including Tables 1 and 2. Both variants retain the consensus penalty. CoSCoDiff-P discards dual memory, whereas CoSCoDiff-PD accumulates disagreement between the guided and corrected samples.

On the toy ring problem, the half-space projection is exact, and PD achieves a smaller distance to the constrained optimum set than P under the default settings. This supports the usefulness of residual history in this controlled setting. For tasks with surrogate-based or approximate feasibility corrections, the residual can also reflect correction error. Accumulating such residuals may bias subsequent guidance and feasibility updates, making primal-only coordination competitive.

This provides a possible explanation for the task-dependent P/PD behavior. The main comparisons do not isolate projection accuracy as its cause, and exact projection alone does not guarantee that PD will outperform P. We therefore present P and PD as complementary coordination variants.

D.3 Hyperparameter Sensitivity

We vary one hyperparameter at a time around $(\rho, K, \tau_{\text{idx}}) = (5, 15, 6)$ for both P and PD. The sweeps use the same 20 seeds and 512 samples per seed, and evaluate the final corrected samples. Figure 6 reports the mean and standard deviation of d_{opt} across seeds. All evaluated configurations satisfy the toy hard constraint after correction.

Iteration budget K. We evaluate $K \in \{1, 5, 10, 15, 20, 30\}$. P and PD coincide at the first iteration, before dual history can affect a subsequent rollout. Both improve from $K = 1$ to $K = 5$, but larger budgets progressively increase d_{opt} in this sweep. For example, PD achieves 0.1680 at $K = 5$, compared with 0.2130 at $K = 15$ and 0.2782 at $K = 30$. Thus, additional stochastic refinement does not imply monotonic improvement, and K is treated as a finite refinement budget.

SDEdit restart index τ_{idx}. We evaluate $\tau_{\text{idx}} \in \{2, 4, 6, 8\}$. Noise levels are ordered from high to low, so larger indices correspond to weaker re-noising and shorter denoising rollouts. PD performs substantially better at indices 6 and 8 than at 2 and 4, consistent with the benefit of preserving more of the corrected sample structure. P responds non-monotonically to the restart index. These results show that restart strength interacts with the coordination variant.

Consensus weight ρ. We sweep ρ from 0 to 20, with finer spacing near zero. P shows a modest decrease in d_{opt} as ρ increases, whereas PD performs better with smaller weights in this experiment. At the zero-penalty endpoint, P recovers Naive Iterate. The implemented PD variant still retains its dual update and dual-shifted feasibility correction when $\rho = 0$; this endpoint therefore remains distinct from naive iteration. The different trends highlight the interaction between the consensus penalty and dual memory.

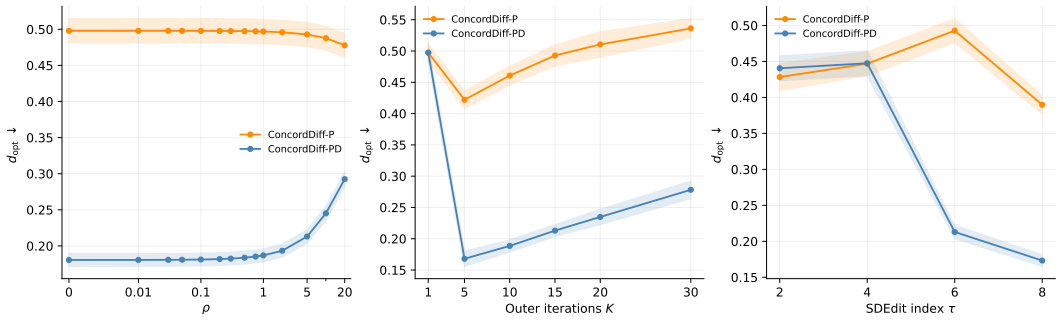

Figure 6: Toy hyperparameter sensitivity for P and PD. Curves show mean d_{opt}; shaded regions indicate standard deviation across seeds. One parameter varies at a time.