Particle Guidance: non-I.I.D. Diverse Sampling with Diffusion Models

Gabriele Corso[*1], **Yilun Xu**[1], **Valentin de Bortoli**[2], **Regina Barzilay**[1], **Tommi Jaakkola**[1]
[1]CSAIL, Massachusetts Institute of Technology, [2]ENS, PSL University

Abstract

In light of the widespread success of generative models, a significant amount of research has gone into speeding up their sampling time. However, generative models are often sampled multiple times to obtain a diverse set incurring in a cost that is orthogonal to sampling time. We tackle the question of how to improve diversity and sample efficiency by moving beyond the common assumption of independent samples. For this we propose *particle guidance*, an extension of diffusion-based generative sampling where a joint-particle time-evolving potential enforces diversity. We analyze theoretically the joint distribution that particle guidance generates, its implications on the choice of potential, and the connections with methods in other disciplines. Empirically, we test the framework both in the setting of conditional image generation, where we are able to increase diversity without affecting quality, and molecular conformer generation, where we reduce the state-of-the-art median error by 13% on average.

1 Introduction

Deep generative modeling has become pervasive in many computational tasks across computer vision, natural language processing, physical sciences, and beyond. In many applications, these models are used to take a number of representative samples of some distribution of interest. Although independent samples drawn from a distribution will perfectly represent it in the limit of infinite samples, for a finite number, this may not be the optimal strategy. Therefore, while deep learning methods have so far largely focused on the task of taking independent identically distributed (I.I.D.) samples from some distribution, this paper examines how one can use deep generative models to take a finite number of samples that can better represent the distribution of interest.

Towards the goal of better *finite-samples* generative models, we propose a general framework for sampling sets of particles using a diffusion model. This framework, which we call *particle guidance*, is based on the use of a time-dependent permutation-invariant potential to guide the inference process. For example, when, to optimize coverage, this potential is set to be the sum of pairwise inverse similarity kernels, the resulting reverse diffusion process combines the usual element-wise score vectors with repulsive terms among particles, inducing the generation of a more diverse set of samples that better covers the distribution of interest (see Fig. 1).

The theoretical analysis of the framework leads us to two key results. On one hand, we obtain an expression for the joint marginal distribution of the sampled process when using any arbitrary guidance potential. On the other, we derive a simple objective one can use to train a model to learn a time-evolving potential that exactly samples from a joint distribution of interest. Further, we also demonstrate the relations of particle guidance to techniques for non-I.I.D. sampling developed in other fields and natural processes and discuss its advantages.

Empirically, we demonstrate the effectiveness of the method in both synthetic experiments and two of the most successful applications of diffusion models: text-to-image generation and molecular

[*]Correspondence to gcorso@mit.edu

Neural Information Processing Systems (NeurIPS) 2023 Workshop on Diffusion Models.

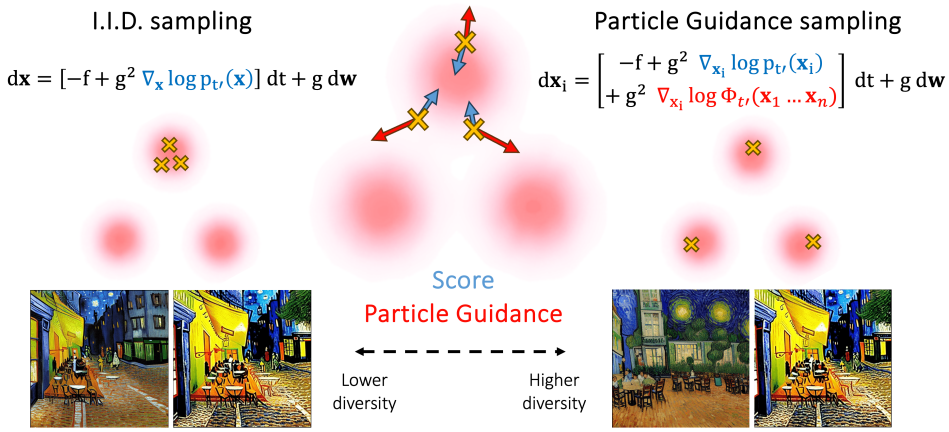


Figure 1: Comparison of I.I.D. and particle guidance sampling. The center figure represents each step, with the distribution in pink and the samples as yellow crosses, where particle guidance uses not only the score (in blue) but also the guidance from the repulsion of the particles (red), leading it to discover different modes (right-hand samples vs those on the left). At the bottom, Van Gogh cafe images samples generated with Stable Diffusion with and without particle guidance.

conformer generation. In the former, we show that particle guidance can improve the diversity of the samples generated with Stable Diffusion while maintaining a quality comparable to that of I.I.D. sampling. For molecular conformer generation, applied to the state-of-the-art method Torsional Diffusion, particle guidance is able to simultaneously improve precision and coverage, reducing their median error by respectively 19% and 8%. In all settings, we also study the critical effect that different potentials can have on the diversity and sample quality.

2 Particle Guidance

Sampling Sets of Particles Our goal is to define a sampling process that promotes the diversity of a finite number of samples while retaining the advantages and flexibility that characterize diffusion models. Let $p(\mathbf{x})$ be some probability distribution of interest and $\nabla_{\mathbf{x}} \log p_t(\mathbf{x})$ be the score that we have learned to reverse the diffusion process $d\mathbf{x} = f(\mathbf{x}, t)dt + g(t)d\mathbf{w}$. Similarly to how classifier guidance is applied, we modify the reverse diffusion process by adding the gradient of a potential. However, we are now sampling together a whole set of particles $\mathbf{x}_1$, ..., $\mathbf{x}_n$, and the potential $\log \Phi_t$ is not only a function of the current point but a permutation invariant function of the whole set:

$$d\mathbf{x}_i = \Bigg[-f(\mathbf{x}_i, t') + g^2(t') \Big(\nabla_{\mathbf{x}_i} \log p_{t'}(\mathbf{x}_i) + \nabla_{\mathbf{x}_i} \log \Phi_{t'}(\mathbf{x}_1, ..., \mathbf{x}_n) \Big) \Bigg] dt + g(t')d\mathbf{w}. \quad (1)$$

where the points are initially sampled I.I.D. from a prior distribution p_T. We call this idea *particle guidance*. This framework allows one to impose different properties, such as diversity, on the set of particles being sampled without the need to retrain a new score model operating directly on the space of sets.

To promote diversity and sample efficiency, in our experiments, we choose the potential $\log \Phi_t$ to be the negative of the sum of a pairwise similarity kernel k between each pair of particles $\log \Phi_t(\mathbf{x}_1, ...\mathbf{x}_n) = -\frac{\alpha_t}{2} \sum_{i,j} k_t(\mathbf{x}_i, \mathbf{x}_j)$ obtaining:

$$d\mathbf{x}_i = \Bigg[-f(\mathbf{x}_i, t') + g^2(t') \Big(\nabla_{\mathbf{x}_i} \log p_{t'}(\mathbf{x}_i) - \alpha_{t'} \nabla_{\mathbf{x}_i} \sum_{j=1}^{n} k_{t'}(\mathbf{x}_i, \mathbf{x}_j) \Big) \Bigg] dt + g(t')d\mathbf{w} \quad (2)$$

Intuitively, this will push our different samples to be dissimilar from one another while at the same time matching our distribution, improving sample efficiency. Critically, this does not come at a significant additional runtime as, in most domains, the cost of running the pairwise similarity kernels is very small compared to the execution of the large score network architecture. Moreover, it allows the use of domain-specific similarity kernels and does not require training any additional classifier or score model.

Theoretical analysis To understand the effect that *particle guidance* has beyond a simple intuition, we study the joint distribution of sets of particles coming out of the proposed reverse diffusion process. However, unlike methods related to energy-based models (see coupled replicas, metadynamics, SVGD in Section B) analyzing the effect of the addition of a time-evolving potential $\log \Phi_t$ in the reverse diffusion is non-trivial.

While the score component in *particle guidance* is the score of the sequence of probability distributions $\tilde{p}_t(\mathbf{x}_1, \ldots, \mathbf{x}_n) = \Phi_t(\mathbf{x}_1, \ldots, \mathbf{x}_n) \prod_{i=1}^{n} p_t(\mathbf{x}_i)$, we are not necessarily sampling exactly $\tilde{p}_0$ because, for an arbitrary time-evolving potential Φ_t, this sequence of marginals does not correspond to a diffusion process. One strategy used by other works in similar situations [Du et al., 2023] relies on taking, after every step or at the end, a number of Langevin steps to reequilibrate and move the distribution back towards $\tilde{p}_t$. This, however, increases significantly the runtime cost and is technically correct only in the limit of infinite steps leaving uncertainty in the real likelihood of our samples. Instead, in Theorem 1 in Appendix C.1, we use the Feynman-Kac theorem to derive a formula for the exact reweighting that particle guidance has on a distribution.

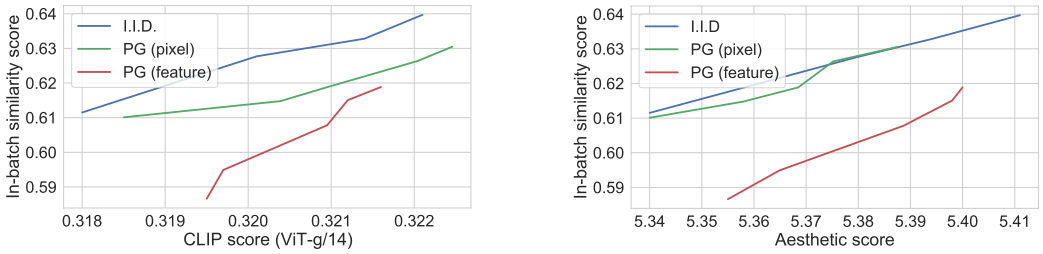
Preserving Invariances The objects that we learn to sample from with generative models often present invariances such as the permutation of the atoms in a molecule or the roto-translation of a conformer. For diffusion models, to obtain a distribution that is invariant to the action of some group G such as that of rotations or permutations, it suffices to have an invariant prior and build a score model that is G-equivariant [Xu et al., 2021]. Similarly, we are interested in distributions that are invariant to the action of G on any of the set elements (see Section 3.2), we show that a sufficient condition for this invariance to be maintained is that the time-evolving potential Φ_t is itself invariant to G-transformations of any of its inputs (see Proposition 1 in Appendix C.3).

3 Experiments

3.1 Text-to-image generation

In practice, most prevalent text-to-image diffusion models, such as Stable Diffusion [Rombach et al., 2021] or Midjourney, generally constrain the output budget to four images per given prompt. Ideally, this set of four images should yield a diverse batch of samples for user selection. However, the currently predominant method of classifier-free guidance [Ho, 2022] tends to push the mini-batch samples towards a typical mode to enhance fidelity, at the expense of diversity.

To mitigate this, we apply the proposed particle guidance to text-to-image generation. Stable Diffusion v1.5 [2] serves as our testbed, having been pre-trained on LAION-5B [Schuhmann et al., 2022] with a resolution of 512×512. In line with [Xu et al., 2023a], we use the validation set in COCO 2014 [Lin et al., 2014] for evaluation, and the CLIP [Hessel et al., 2021]/Aesthetic score [Team, 2022] to assess the text-image alignment/visual quality, respectively. To evaluate the diversity within each batch of generated images corresponding to a given prompt, we introduce the *in-batch similarity score*. This metric represents the average pairwise cosine similarity of features within an image batch, utilizing DINO [Caron et al., 2021] as the feature extractor. For particle guidance, we implement the RBF kernel on the (down-sampled) pixel space in Stable Diffusion model as well as the feature space provided by DINO. Please refer to Appendix F.1 for more experimental details.



(a) In-batch similarity versus CLIP score

(b) In-batch similarity versus Aesthetic score

Figure 2: In-batch similiarity score versus **(a)** CLIP ViT-g/14 score and **(b)** Aesthetic score for text-to-image generation at 512×512 resolution, using Stable Diffusion v1.5 with a varying guidance scale from 6 to 10.

[2] https://huggingface.co/runwayml/stable-diffusion-v1-5

Table 1: Quality of generated conformer ensembles for the GEOM-DRUGS test set in terms of Coverage (%) and Average Minimum RMSD (Å).

	Recall				Precision			
	Coverage ↑		AMR ↓		Coverage ↑		AMR ↓	
Method	Mean	Med	Mean	Med	Mean	Med	Mean	Med
RDKit ETKDG	38.4	28.6	1.058	1.002	40.9	30.8	0.995	0.895
OMEGA	53.4	54.6	0.841	0.762	40.5	33.3	0.946	0.854
GeoMol	44.6	41.4	0.875	0.834	43.0	36.4	0.928	0.841
GeoDiff	42.1	37.8	0.835	0.809	24.9	14.5	1.136	1.090
Torsional Diffusion	72.7	80.0	0.582	0.565	55.2	56.9	0.778	0.729
TD w/ particle guidance	**77.0**	**82.6**	**0.543**	**0.520**	**68.9**	**78.1**	**0.656**	**0.594**

As shown in Fig. 2(a) and Fig. 2(b), particle guidance (PG) consistently obtains a better (lower) in-batch similarity score in most cases, given the same CLIP/Aesthetic score, with a classifier-free guidance scale ranging from 6 to 10. Conversely, we observe that while in-batch similarity score of I.I.D. sampling improves with the reduced classifier-free guidance scale, particle guidance continues to surpass I.I.D. sampling in terms of CLIP/Aesthetic score given the same in-batch similarity. When the repulsive force is applied in the feature space, particle guidance notably attains a lower in-batch similarity score compared to I.I.D. sampling or to the approach in the original downsampled pixel space. This suggests that utilizing a semantically meaningful feature space is more appropriate for determining distances between images.

3.2 Molecular conformer generation

Molecular conformer generation is a key task in computational chemistry that consists of finding the set of different conformations that a molecule most likely takes in 3D space. Critically it is often important to find all or most of the low-energy conformers as each can determine a different behavior.

Over the past few years, molecular conformer generation has been extensively studied by the machine learning community, with well-established benchmarks [Axelrod & Gomez-Bombarelli, 2022] and several generative models designed specifically for this task [Ganea et al., 2021; Xu et al., 2021; Jing et al., 2022]. However, all these methods are based on generating a large number of I.I.D. samples. Therefore, we take the state-of-the-art conformer generation model, *torsional diffusion*, and, without retraining the model itself, we show that we can obtain significant improvements in both coverage and precision via particle guidance.

Torsional diffusion [Jing et al., 2022] defines the diffusion process over the manifold defined by changes in torsion angles from some initial conformer because of the relative rigidity of the remaining degrees of freedom. Given this observation, we also define the guidance kernel on this manifold as an RBF kernel over the dihedral angle differences in such a way that preserves atom permutation invariance (formalized in Appendix E).

Table 1 shows that by tuning the different parameters that derive from applying particle guidance to torsional diffusion (see Appendix E) we are able to balance diversity and coverage with the precision being able to obtain, without retraining the model, significantly improved results on both metrics with 8% and 19% simultaneous reductions respectively in recall and precision median AMR.

4 Conclusion

In this paper, we have analyzed how one can improve the sample efficiency of generative models by moving beyond I.I.D. sampling and enforcing diversity, a critical challenge in many real applications that has been largely unexplored. Our proposed framework, particle guidance, steers the sampling process of diffusion models toward more diverse sets of samples via the definition of a time-evolving joint potential. We have studied the theoretical properties of the framework such as the joint distribution it converges to and shown how it has strong parallels with techniques developed in other disciplines for energy-based methods. Finally, we evaluated its performance in two important applications of diffusion models text-to-image generation and molecular conformer generation, and showed how in both cases it is able to push the Pareto frontier of sample diversity vs quality.

References

Simon Axelrod and Rafael Gomez-Bombarelli. Geom, energy-annotated molecular conformations for property prediction and molecular generation. *Scientific Data*, 2022.

Alessandro Barducci, Giovanni Bussi, and Michele Parrinello. Well-tempered metadynamics: a smoothly converging and tunable free-energy method. *Physical review letters*, 2008.

Joe Benton, Yuyang Shi, Valentin De Bortoli, George Deligiannidis, and Arnaud Doucet. From denoising diffusions to denoising markov models. *arXiv preprint arXiv:2211.03595*, 2022.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2021.

Wei-Cheng Chang, Chun-Liang Li, Youssef Mroueh, and Yiming Yang. Kernel stein generative modeling. *arXiv preprint arXiv:2007.03074*, 2020.

Sinho Chewi, Chen Lu, Kwangjun Ahn, Xiang Cheng, Thibaut Le Gouic, and Philippe Rigollet. Optimal dimension dependence of the metropolis-adjusted langevin algorithm. *ArXiv*, abs/2012.12810, 2020.

Gabriele Corso. Modeling molecular structures with intrinsic diffusion models. *arXiv preprint arXiv:2302.12255*, 2023.

Francesco D'Angelo and Vincent Fortuin. Annealed stein variational gradient descent. *arXiv preprint arXiv:2101.09815*, 2021.

Yilun Du, Conor Durkan, Robin Strudel, Joshua B Tenenbaum, Sander Dieleman, Rob Fergus, Jascha Sohl-Dickstein, Arnaud Doucet, and Will Sussman Grathwohl. Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and mcmc. In *International Conference on Machine Learning*. PMLR, 2023.

Octavian Ganea, Lagnajit Pattanaik, Connor Coley, Regina Barzilay, Klavs Jensen, William Green, and Tommi Jaakkola. Geomol: Torsional geometric generation of molecular 3d conformer ensembles. *Advances in Neural Information Processing Systems*, 2021.

Paul CD Hawkins and Anthony Nicholls. Conformer generation with omega: learning from the data set and the analysis of failures. *Journal of chemical information and modeling*, 2012.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Joseph Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *Conference on Empirical Methods in Natural Language Processing*, 2021.

Jonathan Ho. Classifier-free diffusion guidance. *ArXiv*, abs/2207.12598, 2022.

Gerhard Hummer and Jürgen Köfinger. Bayesian ensemble refinement by replica simulations and reweighting. *The Journal of chemical physics*, 2015.

John Ingraham, Max Baranov, Zak Costello, Vincent Frappier, Ahmed Ismail, Shan Tie, Wujie Wang, Vincent Xue, Fritz Obermeyer, Andrew Beam, et al. Illuminating protein space with a programmable generative model. *BioRxiv*, 2022.

Bowen Jing, Gabriele Corso, Jeffrey Chang, Regina Barzilay, and Tommi Jaakkola. Torsional diffusion for molecular conformer generation. *arXiv preprint arXiv:2206.01729*, 2022.

Ioannis Karatzas and Steven Shreve. *Brownian motion and stochastic calculus*. Springer Science & Business Media, 1991.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in Neural Information Processing Systems*, 2022.

Alessandro Laio and Michele Parrinello. Escaping free-energy minima. *Proceedings of the national academy of sciences*, 2002.

Greg Landrum et al. Rdkit: A software suite for cheminformatics, computational chemistry, and predictive modeling, 2013.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014.

Qiang Liu and Dilin Wang. Stein variational gradient descent: A general purpose bayesian inference algorithm. *Advances in neural information processing systems*, 29, 2016.

Amey P Pasarkar, Gianluca M Bencomo, Simon Olsson, and Adji Bousso Dieng. Vendi sampling for molecular simulations: Diversity as a force for faster convergence and better exploration. *The Journal of Chemical Physics*, 159(14), 2023.

Philipp Pracht, Fabian Bohle, and Stefan Grimme. Automated exploration of the low-energy chemical space with fast quantum chemical methods. *Physical Chemistry Chemical Physics*, 2020.

Robin Rombach, A. Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L. Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, Seyedeh Sara Mahdavi, Raphael Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. *ArXiv*, abs/2205.11487, 2022.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. Laion-5b: An open large-scale dataset for training next generation image-text models. *ArXiv*, abs/2210.08402, 2022.

Chence Shi, Shitong Luo, Minkai Xu, and Jian Tang. Learning gradient fields for molecular conformation generation. In *International Conference on Machine Learning*, 2021.

Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Diffusion art or digital forgery? investigating data replication in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

LAION-AI Team. Laion-aesthetics predictor v2. `https://github.com/christophschuhmann/improved-aesthetic-predictor`, 2022.

Oleg Trott and Arthur J Olson. Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of computational chemistry*, 2010.

Li Kevin Wenliang and Heishiro Kanagawa. Blindness of score-based methods to isolated components and mixing proportions. 2020.

Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. Geodiff: A geometric diffusion model for molecular conformation generation. In *International Conference on Learning Representations*, 2021.

Yilun Xu, Ziming Liu, Max Tegmark, and Tommi Jaakkola. Poisson flow generative models. *Advances in Neural Information Processing Systems*, 2022.

Yilun Xu, Mingyang Deng, Xiang Cheng, Yonglong Tian, Ziming Liu, and Tommi Jaakkola. Restart sampling for improving generative processes. *arXiv preprint arXiv:2306.14878*, 2023a.

Yilun Xu, Ziming Liu, Yonglong Tian, Shangyuan Tong, Max Tegmark, and T. Jaakkola. Pfgm++: Unlocking the potential of physics-inspired generative models. In *International Conference on Machine Learning*, 2023b.

Jason Yim, Brian L Trippe, Valentin De Bortoli, Emile Mathieu, Arnaud Doucet, Regina Barzilay, and Tommi Jaakkola. Se(3) diffusion model with application to protein backbone generation. *arXiv preprint*, 2023.

Jianyi Zhang, Ruiyi Zhang, Lawrence Carin, and Changyou Chen. Stochastic particle-optimization sampling and the non-asymptotic convergence theory. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 2020.

Jingwei Zhuo, Chang Liu, Jiaxin Shi, Jun Zhu, Ning Chen, and Bo Zhang. Message passing stein variational gradient descent. In *International Conference on Machine Learning*. PMLR, 2018.

A Extension: Controlling the Joint Marginal

As we saw in the experiments in Section 3, defining particle guidance with some simple kernels as in Eq. 2 leads to significant sample efficiency improvements with no additional training required and little inference overhead. However, in some domains, particularly in scientific applications, researchers need to control the distribution that they are sampling so that, for example, importance weights can be applied and free energy differences computed. While Theorem 1 allows us to theoretically analyze properties of the distribution, the marginal distribution itself remains largely intractable.

We therefore derive an alternative strategy for choosing the potential in particle guidance: instead of defining the time-evolving potential first and then studying the joint marginal distribution it induces, one can also define the joint distribution of interest and then learn a time-evolving potential that generates it. This is partially similar to learning noise-dependent classifiers in classifier guidance. To achieve this, we mandate that the generation process of particle guidance in Eq. 1 adheres to the sequence of marginals $\hat{p}_t(\mathbf{x}_1^t, ..., \mathbf{x}_n^t) = \Phi_t(\mathbf{x}_1^t, ..., \mathbf{x}_n^t) \prod_{i=1}^n p_t(\mathbf{x}_i^0)$ where Φ_0 is defined by the user based on the desired final joint distribution. We can now learn $\Phi_t^\theta(\mathbf{x}_t)$ to satisfy this evolution. Under mind assumptions, using Doob h-theory (derivation in Appendix C.2), we show that we can learn the $\Phi_t^\theta(\mathbf{x}_t)$ by the following objective:

$$\theta^* = \arg\min_\theta \mathbb{E}_{\mathbf{x}_1^0,...,\mathbf{x}_n^0 \sim p_0} \, \mathbb{E}_{\mathbf{x}_i^t \sim p_{t|0}(\cdot|\mathbf{x}_i^0)} [\|\Phi_0(\mathbf{x}_1^0, ..., \mathbf{x}_n^0) - \Phi_t^\theta(\mathbf{x}_1^t, ..., \mathbf{x}_n^t)\|^2]$$

where $p_{t|0}$ is the Gaussian perturbation kernel in diffusion models. Importantly, here the initial $\mathbf{x}_i^0$ are sampled independently from the data distribution, and p_{0t} refers to the independent diffusion process so this training scheme can be easily executed in parallel to learning the score of p_t.

B Connections with Existing Methods

As discussed in the introduction, other fields have developed methods to improve the tradeoff between sampling cost and coverage of the distribution of interest. In this section, we will briefly introduce four methods (coupled replicas, metadynamics, SVGD and electrostatics) and draw connections with *particle guidance*.

B.1 Coupled Replicas and Metadynamics

In many domains linked to biochemistry and material science, researchers study the properties of the physical systems by collecting several samples from their Boltzmann distributions using molecular dynamics or other enhanced sampling methods. Motivated by the significant cost that sampling each individual structure requires, researchers have developed a range of techniques to go beyond I.I.D. sampling and improve sample efficiency. The most popular of these techniques are parallel sampling with coupled replicas and sequential sampling with metadynamics.

As the name suggests, replica methods involve directly taking n samples of a system with the different sampling processes, replicas, occurring in parallel. In particular, coupled replica methods [Hummer & Köfinger, 2015; Pasarkar et al., 2023], like *particle guidance*, create a dependency between the replicas by adding an extra potential Φ to the energy function to enforce diversity or better match experimental observables. This results in energy-based sampling procedures that target:

$$\tilde{p}(\mathbf{x}_1, \ldots, \mathbf{x}_n) = \Phi(\mathbf{x}_1, \ldots, \mathbf{x}_n) \prod_{i=1}^n p(\mathbf{x}_i).$$

Metadynamics [Laio & Parrinello, 2002; Barducci et al., 2008] was also developed to more efficiently sample the Boltzmann distribution of a given system. Unlike replica methods and our approach, metadynamics is a sequential sampling technique where new samples are taken based on previously taken ones to ensure diversity, typically across certain collective variables of interest $s(\mathbf{x})$. In its original formulation, the Hamiltonian at the kth is augmented with a potential as:

$$\tilde{H}_k = H - \omega \sum_{j<k} \exp\left(- \frac{\|s(\mathbf{x}) - s(\mathbf{x}_j^0)\|^2}{2\sigma^2} \right)$$

where H is the original Hamiltonian, $\mathbf{x}_j^0$ are the previously sampled elements and ω and σ parameters set a priori. Once we take the gradient and perform Langevin dynamics to sample, we obtain dynamics that, with the exception of the fixed Hamiltonian, resemble those of *particle guidance* in Eq. 2 where

$$\nabla_{\mathbf{x}_i} \log \Phi_t(\mathbf{x}_1, \cdots, \mathbf{x}_n) \leftarrow \nabla_{\mathbf{x}_i} \omega \sum_{j<i} \exp\left(- \frac{\|s(\mathbf{x}_i) - s(\mathbf{x}_j^0)\|^2}{2\sigma^2} \right).$$

Although they differ in their parallel or sequential approach, both coupled replicas and metadynamics can be broadly classified as energy-based generative models. As seen here, energy-based models offer a simple way of controlling the joint distribution one converges to by simply adding a potential to the energy function. On the other hand, however, the methods typically employ an MCMC sampling procedure, which lacks the critical *finite-time sampling* property of diffusion models and significantly struggles to cover complex probability distributions such as those of larger molecules and biomolecular complexes. Additionally, the MCMC typically necessitates a substantial number of steps, generally proportional to a polynomial of the data dimension [Chewi et al., 2020]. With particle guidance, we instead aim to achieve both properties (controllable diversity and finite time sampling) at the same time. We can simulate the associated SDE/ODE with a total number of steps that is independent of the data dimension.

B.2 SVGD

Stein Variational Gradient Descent (SVGD) [Liu & Wang, 2016] is a well-established method in the variational inference community to iteratively transport a set of particles to match a target distribution. Given a set of initial particles $\{\mathbf{x}_1^0 \dots \mathbf{x}_n^0\}$, it updates them at every iteration as:

$$\mathbf{x}_i^{\ell-1} \leftarrow \mathbf{x}_i^\ell + \epsilon_\ell \psi(\mathbf{x}_i^\ell) \quad \text{where} \quad \psi(\mathbf{x}) = \frac{1}{n-1} \sum_{j=1}^{n} [k(\mathbf{x}_j^\ell, \mathbf{x}) \nabla_{\mathbf{x}_j^\ell} \log p(\mathbf{x}_j^\ell) + \nabla_{\mathbf{x}_j^\ell} k(\mathbf{x}_j^\ell, \mathbf{x})] \quad (3)$$

where k is some (similarity) kernel and ϵ_ℓ the step size. Although SVGD was developed with the intent of sampling a set of particles that approximate some distribution p without the direct goal of obtaining diverse samples, SVGD and our method have a close relation.

This relation between our method and SVGD can be best illustrated under specific choices for drift and potential under which the probability flow ODE discretization of *particle guidance* can be approximated as (derivation in Appendix C.4):

$$\mathbf{x}_i^{t+\Delta t} \approx \mathbf{x}_i^t + \epsilon_t(\mathbf{x}_i)\psi_t(\mathbf{x}_i^t) \quad \text{where} \quad \psi(\mathbf{x}) = \frac{1}{n-1} \sum_{j=1}^{n} [k_t(\mathbf{x}_j^t, \mathbf{x}) \nabla_{\mathbf{x}} \log p_t(\mathbf{x}) + \nabla_{\mathbf{x}_j^t} k_t(\mathbf{x}_j^t, \mathbf{x})] \quad (4)$$

Comparing this with Eq. 3, we can see a clear relation in the form of the two methods, with some key distinctions. Apart from the different constants, the two methods use different terms for the total score component. Interestingly both methods use smoothed-out scores, however, on the one hand, particle guidance uses the *diffused* score at the specific particle $\mathbf{x}_i$, $\nabla_{\mathbf{x}_i} \log p_t(\mathbf{x}_i)$, while on the other, SVGD smoothes it out by taking a weighted average of the score of nearby particles weighted by the similarity kernel $(\sum_j k(\mathbf{x}_i, \mathbf{x}_j) \nabla_{\mathbf{x}_j} \log p(\mathbf{x}_j))/(\sum_j k(\mathbf{x}_i, \mathbf{x}_j))$.

The reliance of SVGD on other particles for the "smoothing of the score", however, causes two related problems, firstly, it does not have the *finite-time sampling* guarantee that the time evolution of diffusion models provides and, secondly, it suffers from the collapse to few local modes near the initialization and cannot discover isolated modes in data distribution [Wenliang & Kanagawa, 2020]. This challenge has been theoretically [Zhuo et al., 2018] and empirically [Zhang et al., 2020] studied with several works proposing practical solutions. In particular, relevant works use an annealing schedule to enhance exploration [D'Angelo & Fortuin, 2021] or use score matching to obtain a noise-conditioned kernel for SVGD [Chang et al., 2020]. Additionally, we empirically observe that the score smoothing in SVGD results in blurry samples in image generation.

B.3 Electrostatics

Recent works [Xu et al., 2022, 2023b] have shown promise in devising novel generative models inspired by the evolution of point charges in high-dimensional electric fields defined by the data distribution. It becomes natural therefore to ask whether particle guidance could be seen as describing

the evolution of point charges when these are put in the same electric field such that they are not only attracted by the data distribution but also repel one another. One can show that this evolution can indeed be seen as the combination of Poisson Flow Generative Models with particle guidance, where the similarity kernel is the extension of Green's function in $N+1$-dimensional space, *i.e.*, $k(x, y) \propto 1/||x - y||^{N-1}$. We defer more details to Appendix C.5.

C Derivations

C.1 Joint Distribution under Particle Guidance

Theorem 1. *Under integrability assumptions, sampling $\mathbf{x}_1^T, ..., \mathbf{x}_n^T$ from p_T and following the particle guidance reverse diffusion process, we obtain samples from the following joint probability distribution at time $t = 0$:*

$$\hat{p}_0(\mathbf{x}_1, \ldots, \mathbf{x}_n) = \mathbb{E}[Z \exp[-\int_0^T g(t)^2 \{\langle \nabla \log \Phi_t(\mathbf{X}_t), \nabla \log \hat{p}_t(\mathbf{X}_t)\rangle + \Delta \log \Phi_t(\mathbf{X}_t)\} \mathrm{d}t]], \quad (5)$$

with Z (explicit in the appendix) such that

$$\prod_{i=1}^N p_0(\mathbf{x}_i) = \mathbb{E}[Z], \quad (6)$$

$(\mathbf{X}_t)_{t\in[0,T]}$ is a stochastic process driven by the equation

$$\mathrm{d}\mathbf{X}_t = \{f(\mathbf{X}_t, t) - g(t)^2 \nabla \log p_t(\mathbf{X}_t)\}\mathrm{d}t + g(t)\mathrm{d}\mathbf{w}, \qquad \mathbf{X}_0 = \{\mathbf{x}_i\}_{i=1}^N. \quad (7)$$

Hence the density $\hat{p}_0$ can be understood as a reweighting of the random variable Z.

Proof. First we restate the Feynman-Kac theorem. Let $u: [0, T] \times \mathbb{R}^d$ such that for any $t \in [0, T]$ and $x \in \mathbb{R}^d$ we have

$$\partial_t u(t, x) + \langle b(t, x), \nabla u(t, x)\rangle + (1/2)\langle \Sigma(t, x), \nabla^2 u(t, x)\rangle - V(t, x)u(t, x) + f(t, x) = 0, \quad (8)$$

with $u(T, x) = \Phi(T, x)$. Then, under integrability and regularity assumptions, see Karatzas & Shreve [1991] for instance, we have

$$u(0, x) = \mathbb{E}[\int_0^T \exp[-\int_0^r V(\tau, \mathbf{X}_\tau)\mathrm{d}\tau] f(r, \mathbf{X}_r)\mathrm{d}r + \exp[-\int_0^T V(\tau, \mathbf{X}_\tau)\mathrm{d}\tau]\Phi(T, \mathbf{X}_T) \mid \mathbf{X}_0 = x], \quad (9)$$

with $u(T, x) = \Phi(T, x)$ and $\mathrm{d}\mathbf{X}_t = b(t, \mathbf{X}_t)\mathrm{d}t + \Sigma(t, \mathbf{X}_t)\mathrm{d}\mathbf{B}_t$. In the rest of this section, we derive the specific case of Theorem 1.

We recall that the generative model with particle guidance is given by $(\hat{p}_t)_{t\in[0,T]}$ and is associated with the generative model

$$\mathrm{d}\hat{\mathbf{Y}}_t = \{-f(\hat{\mathbf{Y}}_t, T - t) + g(T - t)^2(s_\theta(\hat{\mathbf{Y}}_t, T - t) + \nabla \log \Phi_{T-t}(\hat{\mathbf{Y}}_t))\}\mathrm{d}t + g(T - t)\mathrm{d}\mathbf{w}. \quad (10)$$

We also recall that the generative model without particle guidance is given by $(q_t)_{t\in[0,T]}$ and is associated with the generative model

$$\mathrm{d}\mathbf{Y}_t = \{-f(\mathbf{Y}_t, T - t) + g(T - t)^2 s_\theta(\mathbf{Y}_t, T - t)\}\mathrm{d}t + g(T - t)\mathrm{d}\mathbf{w}. \quad (11)$$

Using the Fokker-Planck equation associated with equation 11 we have for any $x \in (\mathbb{R}^d)^N$

$$\partial_t q_t(x) + \mathrm{div}(\{-f(T - t, \cdot) + g(T - t)^2 s_\theta(T - t, \cdot)\}q_t)(x) - (g(T - t)^2/2)\Delta q_t(x) = 0. \quad (12)$$

This can also be rewritten as

$$\partial_t q_t(x) + \langle -f(T - t, x) + g(T - t)^2 s_\theta(x, T - t), \nabla q_t(x)\rangle - (g(T - t)^2/2)\Delta q_t(x) \quad (13)$$

$$+ \mathrm{div}(\{-f(\cdot, T - t) + g(T - t)^2 s_\theta(T - t, \cdot)\})(x)q_t(x) = 0 \quad (14)$$

Denoting $u_t = q_{T-t}$ we have

$$\partial_t u_t(x) + \langle f(x, t) - g(t)^2 s_\theta(x, t), \nabla u_t(x)\rangle + (g(t)^2/2)\Delta u_t(x) \quad (15)$$

$$- \mathrm{div}(\{-f(t, \cdot) + g(t)^2 s_\theta(\cdot, t)\})(x)u_t(x) = 0. \quad (16)$$

Note that since $u_t = q_{T-t}$, we have that $u_t = p_t$ with the conventions from 2. Now combining this result with equation 8 and equation 9 with $V(t, x) = \mathrm{div}(\{-f(\cdot, t) + g(t)^2 s_\theta(t, \cdot)\})(x)$ and $f = 0$ we have that

$$u_0(x) = \mathbb{E}[Z], \quad (17)$$

with

$$Z = \exp[-\int_0^T V(\tau, \mathbf{X}_\tau)\mathrm{d}\tau]p_0(\mathbf{X}_T), \tag{18}$$

and

$$\mathrm{d}\mathbf{X}_t = \{f(t, \mathbf{X}_t) - g(t)^2 s_\theta(\mathbf{X}_t, t)\}\mathrm{d}t + g(t)\mathrm{d}\mathbf{w}. \tag{19}$$

with $\mathbf{X}_0 = x$. We now consider a similar analysis in the case of the generative with particle guidance. Using the Fokker-Planck equation associated with equation 10 we have for any $x \in (\mathbb{R}^d)^N$

$$\partial_t \tilde{q}_t(x) + \mathrm{div}(\{-f(\cdot, T-t) + g(T-t)^2(s_\theta(\cdot, T-t) + \nabla \log \Phi_{T-t})\}\tilde{q}_t)(x) - (g(T-t)^2/2)\Delta\tilde{q}_t(x) = 0. \tag{20}$$

This can also be rewritten as

$$\partial_t \tilde{q}_t(x) + \langle -f(x, T-t) + g(T-t)^2 s_\theta(x, T-t), \nabla\tilde{q}_t(x)\rangle - (g(T-t)^2/2)\Delta\tilde{q}_t(x) \tag{21}$$

$$+ \mathrm{div}(\{-f(\cdot, T-t) + g(T-t)^2 s_\theta(\cdot, T-t)\})(x)\tilde{q}_t(x) \tag{22}$$

$$+ g(T-t)^2(\langle \log \Phi_{T-t}(x), \nabla \log \tilde{q}_t(x)\rangle + \Delta \log \Phi_{T-t}(x))\tilde{q}_t(x) = 0. \tag{23}$$

Denoting $\hat{u}_t = \tilde{q}_{T-t}$ we have

$$\partial_t \hat{u}_t(x) + \langle f(t, x) - g(t)^2 s_\theta(x, t), \nabla\hat{u}_t(x)\rangle + (g(t)^2/2)\Delta\hat{u}_t(x) \tag{24}$$

$$- \mathrm{div}(\{-f(\cdot, t) + g(t)^2 s_\theta(t, \cdot)\})(x)\hat{u}_t(x) \tag{25}$$

$$- g(t)^2(\langle \nabla \log \Phi_t(x), \nabla \log \tilde{q}_{T-t}(x)\rangle + \Delta \log \Phi_t(x))\hat{u}_t(x) = 0. \tag{26}$$

Following the convetion of 2, we have that $\hat{u}_t = \hat{p}_0$. Now combining this result with equation 8 and equation 9 with $\hat{V}(t, x) = \mathrm{div}(\{-f(\cdot, t) + g(t)^2 s_\theta(\cdot, t)\})(x) + g(t)^2(\langle \nabla \log \Phi_t(x), \nabla \log \tilde{p}_{T-t}(x)\rangle + \Delta \log \Phi_t(x))$ and $f = 0$ we have that

$$\hat{u}_0(x) = \mathbb{E}[\hat{Z}], \tag{27}$$

with

$$\hat{Z} = \exp[-\int_0^T \hat{V}(\tau, \mathbf{X}_\tau)\mathrm{d}\tau]p_0(\mathbf{X}_T), \tag{28}$$

and

$$\mathrm{d}\mathbf{X}_t = \{f(\mathbf{X}_t, t) - g(t)^2 s_\theta(\mathbf{X}_t, t)\}\mathrm{d}t + g(t)\mathrm{d}\mathbf{w}. \tag{29}$$

again with $\mathbf{X}_0 = x$. We conclude the proof upon noting that

$$\hat{Z} = Z \exp[-\int_0^T g(t)^2(\langle \nabla \log \Phi_t(\mathbf{X}_t), \nabla \log \hat{u}_t(\mathbf{X}_t)\rangle + \Delta \log \Phi_t(\mathbf{X}_t))dt]. \tag{30}$$

□

Riemannian Manifolds. Note that our theoretical insights can also be extended to the manifold framework. This is a direct consequence of the fact that the Feynman-Kac theorem can be extended to the manifold setting, see for instance Benton et al. [2022].

C.2 Sampling predefined joint distribution

For ease of derivation via the Doob h-transform, we temporarily reverse the time from t to $T - t$. Here, p_T is treated as the data distribution, and Φ_T is regarded as the potential, as specified by users. We now consider another model. Namely, we are looking for a generative model $\hat{p}_t$ with $t \in [0, T]$ such that for any $t \in [0, T]$ we have $\hat{p}_t = p_t\Phi_t$ with Φ_T given by the user. In layman's terms, this means that we are considering a *factorized* model for all times t with the additional requirement that at the final time T, the model is given by $p_T = \hat{p}_T\Phi_T$ with Φ_T known. This is to be compared with Theorem 1. Indeed in Theorem 1 while the update on the generative dynamics is explicit (particle guidance term), the update on the density is not. In what follows, we are going to see, using tools from Doob h-transform theory, that we can obtain an expression for the update of the drift in the generative process when considering models of the form $\hat{p}_t = p_t\Phi_t$.

More precisely, we consider the following model. Let $\hat{p}_T = p_T$ and for any $s, t \in [0, T]$ with $s < t$ and $\mathbf{x}_s^{1:n} = \{\mathbf{x}_t^i\}_{i=1}^n \in (\mathbb{R}^d)^n$ and $\mathbf{x}_t^{1:n} = \{\mathbf{x}_t^i\}_{i=1}^n \in (\mathbb{R}^d)^n$ we define

$$\hat{p}_{t|s}(\mathbf{x}_t^{1:n}|\mathbf{x}_s^{1:n}) = p_{t|s}(\mathbf{x}_t^{1:n}|\mathbf{x}_s^{1:n})\Phi_t(\mathbf{x}_t^{1:n})/\Phi_s(\mathbf{x}_s^{1:n}), \tag{31}$$

with Φ_t which satisfies for any $\mathbf{x}_t^{1:n} \in (\mathbb{R}^d)^n$

$$\partial_t \Phi_t(\mathbf{x}_t^{1:n}) + \langle -f_{T-t}(\mathbf{x}_t^{1:n}) + g(T-t)^2 \nabla \log p_t(\mathbf{x}_t^{1:n}), \nabla\Phi_t(\mathbf{x}_t^{1:n})\rangle + (g(T-t)^2/2)\Delta\Phi_t(\mathbf{x}_t^{1:n}) = 0, \tag{32}$$

with Φ_T given. Note that equation 32 expresses that Φ_t satisfies the backward Kolmogorov equation. Under mild assumptions, using Doob h-theory, we get that there exists $(\hat{\mathbf{X}}_t)_{t\in[0,T]}$ such that for any $t \in [0,T]$ we have $\hat{\mathbf{Y}}_t \sim \hat{p}_t$ and for any $t \in [0,T]$

$$\mathrm{d}\hat{\mathbf{Y}}_t = \{-f_{T-t}(\hat{\mathbf{Y}}_t) + g(T-t)^2[\nabla \log p_t(\hat{\mathbf{Y}}_t) + \nabla \log \Phi_t(\hat{\mathbf{Y}}_t)]\}\mathrm{d}t + g(T-t)\mathrm{d}\mathbf{w}. \tag{33}$$

The main difficulty is to compute Φ_t for any $t \in [0,T]$. Under mild assumptions, solutions to the backward Kolmogorov equation 32 are for any $t \in [0,T]$ by

$$\Phi_t(\mathbf{x}_t^{1:n}) = \mathbb{E}[\Phi_T(\mathbf{Y}_T)|\mathbf{Y}_t = \mathbf{x}_t^{1:n}] = \int \Phi_T(\mathbf{Y}_T = \mathbf{x}_T^{1:n}) p_{T|t}(\mathbf{x}_T^{1:n}|\mathbf{x}_t^{1:n}) d\mathbf{x}_T^{1:n}, \tag{34}$$

where we have

$$\mathrm{d}\mathbf{Y}_t = \{-f_{T-t}(\mathbf{Y}_t) + g(T-t)^2 \nabla \log p_t(\mathbf{Y}_t)\}\mathrm{d}t + g(T-t)\mathrm{d}\mathbf{w}. \tag{35}$$

This means that $(\mathbf{Y}_t)_{t\in[0,T]}$ is given by the original generative model, with time-dependent marginals p_t. The expression equation 34, suggests to parameterize Φ_t by Φ_t^θ and to consider the loss function

$$\ell_t(\theta) = \mathbb{E}_{\mathbf{Y}_T}\mathbb{E}_{\mathbf{Y}_t\sim p_{t|T}(\cdot|\mathbf{Y}_T)}[\|\Phi_T(\mathbf{Y}_T) - \Phi_t^\theta(\mathbf{Y}_t)\|^2]. \tag{36}$$

Then, we can define a global loss function $\mathcal{L}(\theta) = \int_0^T \lambda(t)\ell_t(\theta)\mathrm{d}t$ where λ_t is some weight. One problem with this original loss function is that it requires sampling and integrating with respect to $\mathbf{Y}_t$ which requires sampling from the generative model.

Recall that we reverse the time from t to $T-t$ at the beginning. Reverse back to the original convention in the main text, Eq. (37) can be expressed as

$$\ell_t(\theta) = \mathbb{E}_{\mathbf{X}_0\sim p_0}\mathbb{E}_{\mathbf{X}_t\sim p_{t|0}(\cdot|\mathbf{X}_T)}[\|\Phi_0(\mathbf{X}_0) - \Phi_t^\theta(\mathbf{X}_t)\|^2]. \tag{37}$$

C.3 Invariance of Particle Guidance

Proposition 1. *Let G be the group of rotations or permutations of a set of vectors. Assuming that $p_T(\mathbf{x})$ is a G-invariant distribution, the learned score $s(\mathbf{x},t)$ and $f(\mathbf{x},t)$ are G-equivariant and the potential $\log \Phi_t(\mathbf{x}_1 \dots \mathbf{x}_n)$ is G-invariant to a transformation of any of its inputs, then the resulting distribution we sample from will also be G-invariant to a transformation of any of the elements of the set.*

Note that in this section we will derive this specific formulation for the group of rotations or permutations and the Brownian motion in Euclidean space. For a more general statement on Lie groups G and Brownian motions associated with a given metric, one could generalize the result from Yim et al. [2023] Proposition F.2.

Proof. For simplicity, we will consider Euler discretization steps going with time from T to 0 (as used in our experiments), however, the proposition applies in the continuous setting too:

$$p_\theta(\mathbf{x}_i^{(t-1)}|\mathbf{x}_{1:n}^{(t)}) = p_{\mathbf{z}}(\mathbf{x}_i^{(t-1)} - \mathbf{x}_i^{(t)} + f(\mathbf{x}_i^{(t)},t) - g^2(\mathbf{s}_\theta(\mathbf{x}_i^{(t)},t) + \nabla_{\mathbf{x}_i^{(t)}} \log \Phi_t(\mathbf{x}_{1:n}^{(t)})))$$

where $\mathbf{z} \sim N(0, g^2 I)$. Without loss of generality since the whole method is invariant to permutations of the particles, consider $\mathbf{x}_n$ to be the particle to which we apply T_g the transformation of an arbitrary group element g.

Since by assumption $\log \Phi_t(\mathbf{x}_{1:n}^{(t)}) = \log \Phi_t(\mathbf{x}_{1:n-1}^{(t)}, T_g(\mathbf{x}_n^{(t)}))$ we have $p_\theta(\mathbf{x}_i^{(t-1)}|\mathbf{x}_{1:n}^{(t)}) = p_\theta(\mathbf{x}_i^{(t-1)}|\mathbf{x}_{1:n-1}^{(t)}, T_g(\mathbf{x}_n^{(t)}))$.

On the other hand, since $\log \Phi_t(\mathbf{x}_{1:n}^{(t)})$ is invariant to G transformations of $\mathbf{x}_n^{(t)}$, its gradient w.r.t. the same variable will be G-equivariant. Therefore:

$$\begin{aligned}
&p_\theta(T_g(\mathbf{x}_n^{(t-1)})|\mathbf{x}_{1:n-1}^{(t)}, T_g(\mathbf{x}_n^{(t)})) = \\
&= p_{\mathbf{z}}(T_g(\mathbf{x}_n^{(t-1)}) - T_g(\mathbf{x}_n^{(t)}) + f(T_g(\mathbf{x}_n^{(t)}),t) - g^2(\mathbf{s}_\theta(T_g(\mathbf{x}_n^{(t)}),t) + \nabla_{\mathbf{x}_n^{(t)}} \log \Phi_t(\mathbf{x}_{1:n-1}^{(t)}, T_g(\mathbf{x}_n^{(t)})))) \\
&= p_{\mathbf{z}}(T_g(\mathbf{x}_n^{(t-1)}) - T_g(\mathbf{x}_n^{(t)}) + T_g(f(\mathbf{x}_n^{(t)},t)) - g^2(T_g(\mathbf{s}_\theta(\mathbf{x}_n^{(t)},t)) + T_g(\nabla_{\mathbf{x}_n^{(t)}} \log \Phi_t(\mathbf{x}_{1:n}^{(t)})))) \\
&= p_{\mathbf{z}}(T_g(\mathbf{x}_n^{(t-1)} - \mathbf{x}_n^{(t)} + f(\mathbf{x}_n^{(t)},t) - g^2(\mathbf{s}_\theta(\mathbf{x}_n^{(t)},t) + \nabla_{\mathbf{x}_n^{(t)}} \log \Phi_t(\mathbf{x}_{1:n}^{(t)})))) = p_\theta(\mathbf{x}_n^{(t-1)}|\mathbf{x}_{1:n}^{(t)})
\end{aligned}$$

where between lines 2 and 3 we have used the equivariance assumptions and in the latter two the properties of elements of G.

Putting these together, we follow a similar derivation of Proposition 1 from Xu et al. [2021]:

$$
\begin{aligned}
&p_\theta(\mathbf{x}_{1:n-1}^{(0)}, T_g(\mathbf{x}_n^{(0)})) = \\
&= \int p(\mathbf{x}_{1:n-1}^{(T)}, T_g(\mathbf{x}_n^{(T)})) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{1:n-1}^{(t-1)}, T_g(\mathbf{x}_n^{(t-1)}) | \mathbf{x}_{1:n-1}^{(t)}, T_g(\mathbf{x}_n^{(t)})) = \\
&= \int \left(\prod_{i<n} p(\mathbf{x}_i^{(T)}) \right) \left(\prod_{t=1}^{T} \prod_{i<n} p_\theta(\mathbf{x}_i^{(t-1)} | \mathbf{x}_{1:n-1}^{(t)}, T_g(\mathbf{x}_n^{(t)})) \right) \cdot \\
&\qquad \cdot \left(p(T_g(\mathbf{x}_n^{(T)})) \prod_{t=1}^{T} p_\theta(T_g(\mathbf{x}_n^{(t-1)}) | \mathbf{x}_{1:n-1}^{(t)}, T_g(\mathbf{x}_n^{(t)})) \right) = \\
&= \int \left(\prod_{i<n} p(\mathbf{x}_i^{(T)}) \right) \left(\prod_{t=1}^{T} \prod_{i<n} p_\theta(\mathbf{x}_i^{(t-1)} | \mathbf{x}_{1:n}^{(t)}) \right) \left(p(\mathbf{x}_n^{(T)}) \prod_{t=1}^{T} p_\theta(\mathbf{x}_n^{(t-1)} | \mathbf{x}_{1:n}^{(t)}) \right) = p_\theta(\mathbf{x}_{1:n}^{(0)})
\end{aligned}
$$

□

C.4 Particle Guidance as SVGD

In this section, we derive the approximation of Eq. 4 starting from the probability flow ODE equivalent of Eq. 2 under the assumptions of no drift $f(x,t) = 0$ and using the following form for $\Phi_t(x_1, ..., x_n) = (\sum_{i,j} k_t(x_i, x_j))^{-\frac{n-1}{2}}$ where k_t is a similarity kernel based on the Euclidean distance (e.g. RBF kernel).

$$
dx_i = \left[f(x_i, t) - \frac{1}{2} g^2(t) \left(\nabla_{x_i} \log p_t(x_i) + \nabla_{x_i} \log \left(\sum_{ij} k_t(x_i, x_j) \right)^{-\frac{n-1}{2}} \right) \right] dt
$$

$$
= -\frac{1}{2} g^2(t) dt \left(\nabla_{x_i} \log p_t(x_i) - \frac{\frac{1}{2} \nabla_{x_i} \sum_{ij} k_t(x_i, x_j)}{\frac{1}{n-1} \sum_{ij} k_t(x_i, x_j)} \right)
$$

Now we can simplify the numerator using the fact that k_t is symmetric and approximate the denominator assuming that different particles will have similar average distances to other particles:

$$
\approx -\frac{1}{2} g^2(t) dt \left(\nabla_{x_i} \log p_t(x_i) - \frac{\nabla_{x_i} \sum_j k_t(x_i, x_j)}{\sum_j k_t(x_i, x_j)} \right)
$$

$$
= -\frac{g^2(t) dt}{2\, S(x_i)} \left(\sum_j k_t(x_i, x_j) \nabla_{x_i} \log p_t(x_i) - \nabla_{x_i} k_t(x_i, x_j) \right)
$$

where $S(x_i) = \sum_j k_t(x_i, x_j)$. Now we can use the fact that $\nabla_{x_i} k_t(x_i, x_j) = -\nabla_{x_j} k_t(x_i, x_j)$ because the kernel only depends on the Euclidean distance between the two points:

$$
= -\frac{n\, g^2(t) dt}{2\, S(x_i)} \left(\frac{1}{n-1} \sum_j k_t(x_i, x_j) \nabla_{x_i} \log p_t(x_i) + \nabla_{x_j} k_t(x_i, x_j) \right)
$$

Letting $\epsilon_t(x_i) = \frac{n\, g^2(t) \Delta t}{2\, S(x_i)}$, we obtain Eq. 4:

$$
x_i^{t-\Delta t} \approx x_i^t + \epsilon_t(x_i) \psi_t(x_i^t) \quad \text{where} \quad \psi(x) = \frac{1}{n-1} \sum_{j=1}^{n} [k_t(x_j^t, x) \nabla_x \log p_t(x) + \nabla_{x_j^t} k_t(x_j^t, x)]
$$

C.5 Particle Guidance in Poisson Flow Generative Models

In this section, we consider the more general Poisson Flow Generative Models++ [Xu et al., 2023b] framework in which the N-dimensional data distribution is embedded into $N+D$-dimensional space, where D is a positive integer ($D = 1/D \to \infty$ recover PFGM [Xu et al., 2022]/diffusion models). The data distribution is interpreted as a positive charge distribution. Each particle independently follows the electric field generated by the N-dimensional data distribution $p(x)$ embedded in a $N+D$-dimensional space. One can similarly do particle guidance in the PFGM++ scenarios, treating the group of particles as negative charges, not only attracted by the data distribution but also exerting the mutually repulsive force. Formally, for the augmented data the ODE in PFGM++ (Eq.4 in Xu et al. [2023b]) is

$$\frac{dx}{dr} = \frac{E(x,r)_x}{E(x,r)_r}$$

where E_x and E_r are the electric fields for different coordinates:

$$E(x,r)_x = \frac{1}{S_{N+D-1}(1)} \int \frac{x-y}{(\|x-y\|^2 + r^2)^{\frac{N+D}{2}}} p(y)dy$$

$$E(x,r)_r = \frac{1}{S_{N+D-1}(1)} \int \frac{r}{(\|x-y\|^2 + r^2)^{\frac{N+D}{2}}} p(y)dy$$

Note that when $r = \sigma\sqrt{D}, D \to \infty$, the ODE is $\frac{dx}{dr} = \frac{E(x,r)_x}{E(x,r)_r} = -\frac{\sigma}{\sqrt{D}} \nabla_x \log p_\sigma(x)$ and the framework degenerates to diffusion models.

Now if we consider the repulsive forces among a set of (uniformly weighted) particles with the same anchor variables r, $\{(x_i, r)\}_{i=1}^n$, only the electric field in the x coordinate changes (the component in the r coordinate is zero). Denote the new electric field in x component as $\hat{E}_x$:

$$\hat{E}(x_i,r)_x = \underbrace{E(x_i,r)_x}_{\text{attractive force by data}} + \underbrace{\frac{1}{S_{N+D-1}(1)} \frac{1}{n-1} \sum_{j \neq i} \frac{x_j - x_i}{(\|x_j - x_i\|^2)^{\frac{N+D}{2}}}}_{\text{repulsive force between particles}}$$

The corresponding new ODE for the i-th particle is

$$\begin{aligned}
\frac{dx_i}{dr} &= \frac{\hat{E}(x_i,r)_x}{E(x_i,r)_r} \\
&= \frac{E(x_i,r)_x}{E(x_i,r)_r} + \frac{\frac{1}{S_{N+D-1}(1)} \frac{1}{n-1} \sum_{j \neq i} \frac{x_j - x_i}{(\|x_j - x_i\|^2)^{\frac{N+D}{2}}}}{E(x_i,r)_r} \\
&= \underbrace{\frac{E(x_i,r)_x}{E(x_i,r)_r}}_{\text{predicted by pre-trained models}} + \underbrace{\frac{\frac{1}{n-1} \sum_{j \neq i} \frac{x_j - x_i}{(\|x_j - x_i\|^2)^{\frac{N+D}{2}}}}{\int \frac{r}{(\|x-y\|^2 + r^2)^{\frac{N+D}{2}}} p(y)dy}}_{\text{particle guidance}} \\
&= \frac{E(x_i,r)_x}{E(x_i,r)_r} + \frac{\frac{1}{n-1} \sum_{j \neq i} \frac{x_j - x_i}{\|x_j - x_i\|^{N+D}}}{\frac{S_{N+D-1}}{r^{D-1} S_{D-1}} p_r(x_i)}
\end{aligned}$$

where p_r is the intermidate distribution, and S_n is the surface area of n-sphere. Clearly, the direction of the guidance term can be regarded as the sum of the gradient of $N+D$-dimensional Green's function $G(x,y) \propto 1/||x-y||^{N+D-2}$, up to some scaling factors:

$$\nabla_{x_i} G(x_i, x_j) = \frac{x_i - x_j}{\|x_j - x_i\|^{N+D}}$$

C.6 Combinatorial Analysis of Synthetic Experiments

Proposition 2. *Let us have a random variable taking a value equiprobably between N distinct bins. The expectation of the proportion of bins discovered (i.e. sampled at least once) after N samples is* $1 - (\frac{N-1}{N})^N$ *which tends to* $1 - 1/e$ *as* N *tends to infinity.*

Proof. Let n_i be the number of samples in the i^{it} bin. The proportion of discovered bins is equal to:

$$\frac{1}{N} E[\sum_{i=1}^{N} I_{n_i>0}] = \frac{1}{N} \sum_{i=1}^{N} E[I_{n_i>0}] = P(n_i > 0) = 1 - P(n_i = 0) = 1 - (\frac{N-1}{N})^N$$

In the limit of $N \to \infty$:

$$\lim_{N\to\infty} 1 - (\frac{N-1}{N})^N = 1 - y = 1 - \frac{1}{e}$$

since (using L'Hôpital's rule):

$$\log y = \lim_{N\to\infty} N \log(\frac{N-1}{N}) = \lim_{N\to\infty} \frac{\log(\frac{N-1}{N})}{1/N} = \lim_{N\to\infty} \frac{1/N^2}{-1/N^2} = -1$$

□

Therefore for $N = 10$ we would expect $10 * (1 - 0.9^{10}) \approx 6.51$, which corresponds to what is observed in the empirical results of Section D.

Proposition 3. ***(Coupon collector's problem)*** *Let us have a random variable taking a value equiprobably between N distinct bins. The expectation of the number of samples required to discover all the bins is* $N\, H_N$*, where* H_N *is the Nth harmonic number, which is* $\Theta(N \log N)$ *as* N *tends to infinity.*

Proof. Len $L_{i|j}$ be the number of samples it takes to go from j to i bins discovered. We are therefore interested in $E[L_{N|0}]$.

$$E[L_{j|j-1}] = \frac{N-(j-1)}{N} * 1 + \frac{j-1}{N}[E[L_{j|j-1}] + 1] \implies E[L_{j|j-1}] = \frac{N}{N-(j-1)}$$

Therefore:

$$E[L_{N|0}] = E[\sum_{j=1}^{N} L_{j|j-1}] = \sum_{j=1}^{N} E[L_{j|j-1}] = N \sum_{j=1}^{N} \frac{1}{N-(j-1)} = N \sum_{j=1}^{N} \frac{1}{j} = N\, H_N$$

Since H_N is $\Theta(\log N)$, then $E[L_{N|0}]$ is $\Theta(N \log N)$. □

For $N = 10$, $E[L_{10|0}] \approx 29.29$.

D Synthetic experiments

To show visually the properties of particle guidance and its effect on sample efficiency, we use a two-dimensional Gaussian mixture model. In particular, we consider a mixture of $N = 10$ identical Gaussian distributions whose centers are equally spaced over the unit circle and whose variance is 0.005. These Gaussians form a set of approximately disjoint equal bins. As we are interested in inference, no model is trained and the true score of the distribution is given as an oracle.

As expected if one runs normal I.I.D. diffusion, the sample falls in one bin at random. Taking ten samples, as shown in Fig. 3, some of them will fall in the same bin and some bins will be left unfound. The empirical experiments confirm the combinatorial analysis (see Appendix C.6) which shows that the expected number of bins discovered with $N = 10$ samples is only 6.5 and it takes on average more than 29 samples to discover all the bins.

In many settings this behavior is suboptimal, and we would want our model to discover all the modes of the distribution with as few samples as possible. Using the straightforward application of particle

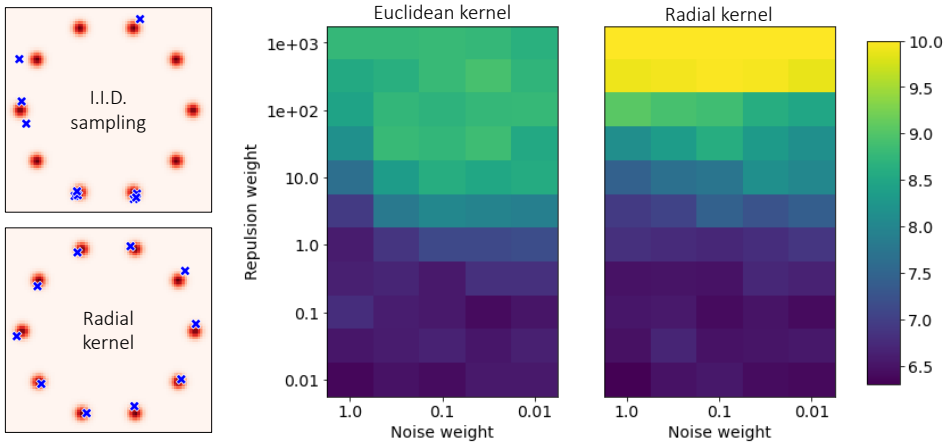


Figure 3: Left: plot of random samples (in blue) of the two-dimensional Gaussian mixture distribution (density depicted in red). I.I.D. samples often recover the same modes, while particle guidance with a radial kernel captures all modes. Right: average number of modes recovered with 10 samples as a function of the weight given by the diffusion noising terms and the repulsion weight when using an RBF kernel with Euclidean and radial distances respectively. As expected with little weight to the repulsion terms we obtain approximately 6.5 modes recovered in line with the I.I.D. diffusion performance. Further increasing the repulsion weight on the Euclidean creates instability.

guidance with a simple RBF kernel based on the squared Euclidean distance, we are able to encourage diversity obtaining, on average, the discovery of nearly 9 bins (see Fig. 3).

Intrinsic diffusion models [Corso, 2023] have shown significant improvements when diffusion models operate on the submanifold where the data lies. Similarly, here building into the kernel the degrees of freedom over which the diversity lies helps the particle guidance to effectively distribute the samples over the distribution. We know that the different modes are distributed in a radial fashion, and thus we build an RBF kernel based on the angle difference w.r.t. the origin. Using this lower-dimensional kernel enables us to consistently discover all modes of the distribution. This submanifold observation aligns well with the practice of methods such as metadynamics where the kernels are defined over some lower-dimensional collective variables of interest.

E Molecular Conformer Generation Experiments

E.1 Dataset, Metrics and Baselines

Dataset We evaluate the method for the task of molecular conformer generation using the data from GEOM [Axelrod & Gomez-Bombarelli, 2022], a collection of datasets that has become the standard benchmark for this task in the machine learning community. In particular, we focus on GEOM-DRUGS, the largest, most pharmaceutically relevant and widely used dataset in GEOM which consists of 304k drug-like molecules. For each of these molecules, an ensemble of conformers was generated with metadynamics in CREST [Pracht et al., 2020], a procedure that gives accurate structures but is prohibitive in high-throughput applications, costing an average of 90 core-hours per molecule. To be able to use existing pretrained models we rely on the experimental setup and splits introduced by Ganea et al. [2021] and used by several papers afterward. As we do not retrain the score model, we do not use the training set, instead, we finetune the inference parameters for *particle guidance* and the other ablation experiments on a random subset of 200 molecules out of 30433 from the validation set.

Evaluation metrics To evaluate conformer generation methods we want to test the ability of a method to generate a set of conformers that are both individually good poses (precision) and as a set cover the distribution of true conformers (recall). For this we employ the same evaluation setup and metrics used by several papers in the field starting from Ganea et al. [2021]. In this setup, methods are asked to generate twice as many conformers as in the original ensemble and then the so-called Average Minimum RMSD (AMR) and Coverage (COV) are measured for precision (P) and recall (R).

For $K = 2L$ let $\{C_l^*\}_{l\in[1,L]}$ and $\{C_k\}_{k\in[1,K]}$ be respectively the sets of ground truth and generated conformers:

$$\begin{aligned} \text{COV-R} &:= \frac{1}{L}\left|\{l \in [1..L] : \exists k \in [1..K], \text{RMSD}(C_k, C_l^*) < \delta\right| \\ \text{AMR-R} &:= \frac{1}{L}\sum_{l\in[1..L]} \min_{k\in[1..K]} \text{RMSD}(C_k, C_l^*) \end{aligned} \tag{38}$$

where δ is the coverage threshold (set to 0.75Å for the GEOM-DRUGS experiments). Swapping ground truth and generated conformers in the equations above we obtain the precision metrics.

Baselines As baselines we report the performances of previous methods as measured by Ganea et al. [2021] and Jing et al. [2022]. These are the cheminformatics methods RDKit ETKDG [Landrum et al., 2013] and OMEGA [Trott & Olson, 2010; Hawkins & Nicholls, 2012] and the machine learning models GeoMol [Ganea et al., 2021], GeoDiff [Xu et al., 2021], CGCF [Shi et al., 2021], and Torsional Diffusion [Jing et al., 2022].

E.2 Particle Guidance Setup

Reverse diffusion As discussed in Section 3.2, we applied particle guidance to torsional diffusion, as this is currently considered to be state-of-the-art and it uses, like most ML-based methods before, I.I.D. sampling during inference. We define the particle guidance kernel to operate directly on the implicit hypertorus manifold where torsional diffusion defines the diffusion process, this, at the same time, makes the kernel lower dimensional and it involves a minor modification to the existing inference procedure. The reverse diffusion process that we apply is:

$$\begin{aligned} d\boldsymbol{\tau}_i = &\underbrace{\frac{1}{2}\, g^2(T-t)\, \mathbf{s}(\boldsymbol{\tau}_i, L, T-t)\, dt}_{\text{diffusion ODE}} + \underbrace{\beta_{T-t}\left(\frac{1}{2}\, g^2(t)\, \mathbf{s}(\boldsymbol{\tau}_i, L, T-t)\, dt + g(T-t)\, d\mathbf{w}\right)}_{\text{Langevin diffusion SDE}} \\ &+ \underbrace{\frac{\gamma_{T-t}}{2}\, g^2(T-t)\, \nabla_{\boldsymbol{\tau}_i} \log \Phi_{T-t}(\boldsymbol{\tau}_1, ..., \boldsymbol{\tau}_n) dt}_{\text{particle guidance}} \end{aligned}$$

where we follow the idea from Karras et al. [2022] of dividing the different components of the reverse diffusion and tuning their individual parameters. The potential was chosen to be:

$$\log \Phi_t(\boldsymbol{\tau}_1, ...\boldsymbol{\tau}_n) = -\frac{\alpha_t}{2n}\sum_{i,j} k_t(\boldsymbol{\tau}_i, \boldsymbol{\tau}_j) \quad \text{where} \quad k_t(\boldsymbol{\tau}_i, \boldsymbol{\tau}_j) = \exp(-\frac{||\boldsymbol{\tau}_i - \boldsymbol{\tau}_j||^2}{h_t}) \tag{39}$$

where the difference of each torsion angle is computed to be in $(-\pi, \pi]$. α_t, β_t, γ_t and h_t are inference hyperparameters that are 'logarithmically interpolated' between two end values chosen with hyperparameter tuning (using $T = 1$), e.g. $\alpha_t = \exp(t\log(\alpha_1) + (1-t)\log(\alpha_0))$.

Permutation invariant kernel Since the kernel operates on the torsion angle differences it is naturally invariant to SE(3) transformations, i.e. translations or rotations, of the conformers in space. Moreover, as illustrated in Fig. 2 of Jing et al. [2022], while exact torsion angle values depend on arbitrary choices of neighbors or orientation (to compute the dihedral angle) differences in torsion angles are invariant to these choices. However, one transformation that the kernel in Equation 39 is not invariant to are permutations of the atoms in the molecule. Many of these permutations lead to isomorphic molecular graphs where however each of the torsion angles may now refer to a different dihedral. To maximize the sample efficiency we make the kernel invariant to these by taking the minimum over the values of the kernel under all such permutations:

$$k_t'(\boldsymbol{\tau}_i, \boldsymbol{\tau}_j) = \min_{\pi\in\Pi} k_t(\boldsymbol{\tau}_i, P_\pi \boldsymbol{\tau}_j)$$

where Π is the set of all permutations that keep the graph isomorphic (but do change the torsion angles assignment) and P_π is the permutation matrix corresponding to some permutation π. In practice, these isomorphisms can be precomputed efficiently, however, to limit the overhead from applying the kernel multiple times, whenever there are more than 32 isomorphic graphs leading to a change in dihedral assignments we subsample these to only keep 32.

E.3 Full Results

We provide in Table 2 again the results reported in Table 1 with the additions of other baselines and ablation experiments. In particular, as ablations, on top of running non-invariant particle guidance i.e. without the minimization over the permutations described in the previous section, we also test low-temperature sampling, another variation of the inference-time procedure that has been proposed for diffusion models that we applied as described below.

Low-temperature sampling. Low-temperature sampling of some distribution $p(\mathbf{x})$ with temperature $\lambda^{-1} < 1$ consists of sampling the distribution $p_\lambda(\mathbf{x}) \propto p(\mathbf{x})^\lambda$. This helps mitigate the overdispersion problem by concentrating more on high-likelihood modes and trading off sample diversity for quality. Exact low-temperature sampling is intractable for diffusion models, however, various approximation schemes exist. We use an adaptation of Hybrid Langevin-Reverse Time SDE proposed by Ingraham et al. [2022]:

$$d\boldsymbol{\tau} = -\left(\lambda_t + \frac{\lambda\,\psi}{2}\right) \mathbf{s}_{\theta,G}(C,t)\; g^2(t)\; dt + \sqrt{1+\psi}\; g(t)\; d\mathbf{w} \quad \text{with } \lambda_t = \frac{\sigma_d + \sigma_t}{\sigma_d + \sigma_t/\lambda}$$

where λ (the inverse temperature), ψ and σ_d are parameters that can be tuned.

Table 2: Quality of generated conformer ensembles for the GEOM-DRUGS test set in terms of Coverage (%) and Average Minimum RMSD (Å). Minimizing recall and precision refers to the hyperparameter choices that minimize the respective median AMR on the validation set.

	Recall				Precision			
	Coverage ↑		AMR ↓		Coverage ↑		AMR ↓	
Method	Mean	Med	Mean	Med	Mean	Med	Mean	Med
RDKit ETKDG	38.4	28.6	1.058	1.002	40.9	30.8	0.995	0.895
OMEGA	53.4	54.6	0.841	0.762	40.5	33.3	0.946	0.854
CGCF	7.6	0.0	1.247	1.225	3.4	0.0	1.837	1.830
GeoMol	44.6	41.4	0.875	0.834	43.0	36.4	0.928	0.841
GeoDiff	42.1	37.8	0.835	0.809	24.9	14.5	1.136	1.090
Torsional Diffusion	72.7	80.0	0.582	0.565	55.2	56.9	0.778	0.729
TD w/ low temperature								
- minimizing recall	73.3	77.7	0.570	0.551	66.4	73.8	0.671	0.613
- minimizing precision	68.0	69.6	0.617	0.604	72.4	81.3	0.607	0.548
TD w/ non-invariant PG								
- minimizing recall	75.8	81.5	**0.542**	**0.520**	66.2	72.4	0.668	0.607
- minimizing precision	58.9	56.8	0.730	0.746	**76.8**	**88.8**	**0.555**	**0.488**
TD w/ invariant PG								
- minimizing recall	**77.0**	**82.6**	0.543	**0.520**	68.9	78.1	0.656	0.594
- minimizing precision	72.5	75.1	0.575	0.578	72.3	83.9	0.617	0.523

F Experimental Details on Stable Diffusion

F.1 Setup

In this section, we detail the experimental setup on Stable Diffusion model. We replace the score function ($\nabla_{\mathbf{x}_i} \log p_{t'}(\mathbf{x}_i)$) in the original particle guidance formula (Eq. (2)) with the classifier-free guidance [Ho, 2022] as follows:

$$\tilde{s}(\mathbf{x}_i, c, t') = w\nabla_{\mathbf{x}_i} \log p_{t'}(\mathbf{x}_i, c) + (1-w)\nabla_{\mathbf{x}_i} \log p_{t'}(\mathbf{x}_i)$$

where c symbolizes the text condition, $w \in \mathbb{R}^+$ is the guidance scale, and $\nabla_{\mathbf{x}_i} \log p_{t'}(\mathbf{x}_i, c)/\log p_{t'}(\mathbf{x}_i)$ is the conditional/unconditional scores, respectively. As probability ODE with classifier-free guidance is the prevailing method employed in text-to-image models [Saharia et al., 2022], we subsititue the reverse-time SDE in Eq. (2) with the margianlly

equivalent ODE. Assuming that $f(\mathbf{x}_i, t') = 0$, the new backward ODE with particle guidance is

$$d\mathbf{x}_i = \left[\frac{1}{2}g^2(t')\left(\tilde{s}(\mathbf{x}_i, c, t') - \alpha_{t'}\nabla_{\mathbf{x}_i}\sum_{j=1}^{n} k_{t'}(\mathbf{x}_i, \mathbf{x}_j)\right)\right]dt$$

Following SVGD [Liu & Wang, 2016], we employ RBF kernel $k_t(\boldsymbol{\tau}_i, \boldsymbol{\tau}_j) = \exp(-\frac{||\boldsymbol{\tau}_i - \boldsymbol{\tau}_j||^2}{h_t})$ with $h_t = m_t^2/\log n$, where m_t is the median of particle distances. We implement the kernel both in the original (dowmsampled) pixel space or the feature space of DINO-VIT-b/8 [Caron et al., 2021]. Defining the DINO feature extractor as g_{DINO}, the formulation becomes:

$$d\mathbf{x}_i = \left[\frac{1}{2}g^2(t')\left(\tilde{s}(\mathbf{x}_i, c, t') - \alpha_{t'}\nabla_{\mathbf{x}_i^0}\sum_{j=1}^{n} k_{t'}\left(g_{\text{DINO}}(\mathbf{x}_i^0), g_{\text{DINO}}(\mathbf{x}_j^0)\right)\right)\right]dt$$

where we set the input to the DINO feature extractor g_{DINO} to the $\mathbf{x}_0$-prediction of at $\mathbf{x}_i$: $\mathbf{x}_i^0 = \mathbf{x}_i + \sigma(t')^2\tilde{s}(\mathbf{x}_i, c, t')$, as $\mathbf{x}_0$-prediction lies in the data manifold rather than noisy images. $\sigma(t)$ is the standard deviation of Gaussian perturbatio kernel given time t in diffusion models. The gradient can be calculated by forward-mode auto-diff. We hypothesize that defining Euclidean distance in the feature space is markedly more natual and effective compared to the pixel space, allowing the repulsion in the semantic meaning. Our experimental resutls in Fig. 3.1 corroborate the hypothesis.

We randomly sample 500 prompts from the COCO validation set [Lin et al., 2014]. For each prompts, we generate a batch of four images. To get the average CLIP score/Aesthetic score versus in-batch similarity score curve, for I.I.D. sampling, we use $w \in \{6, 7.5, 8.5, 9\}$; for particle guidance, we use a set of larger guidance scales: $w \in \{7.5, 8, 9, 9.5, 10\}$. We set the hyper-parameter $\alpha_{t'}$ to $8\sigma(t)$ in particle guidance (feature) and $30\sigma(t)^2$ in particle guidance (pixel). We use Euler solver with 30 NFE in all the experiments.

F.2 In-batch similarity score

We propose in-batch similarity score to capture the diversity of a small set of samples $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ given a prompt c:

$$\text{In-batch similarity score}(\mathbf{x}_1, \ldots, \mathbf{x}_n) = \frac{1}{n(n-1)}\sum_{i \neq j}\frac{g_{\text{DINO}}(\mathbf{x}_i)^T g_{\text{DINO}}(\mathbf{x}_j)}{||g_{\text{DINO}}(\mathbf{x}_i)||_2 ||g_{\text{DINO}}(\mathbf{x}_j)||_2}$$

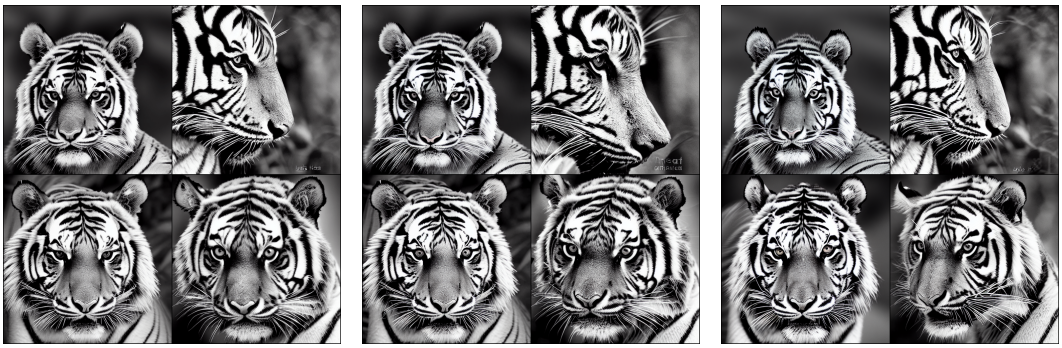
To save memory, we use the DINO-VIT-s/8 [Caron et al., 2021] as the feature extractor.

G Extended Image Samples

In Fig. 4-Fig. 8, we visualize samples generated by the I.I.D. sampling process, particle guidance in the pixel space and particle guidance in the DINO feature space, on four different prompts. For Fig. 4-Fig. 6, we select the prompts in Somepalli et al. [2023], with which Stable Diffusion model is shown to replicate content directly from the LAION dataset. We also include the the generated samples of SVGD-guidance, in which we replace the particle guidance term with SVGD formula (Eq. (3)). In Fig. 9, we observe that SVGD generally leads to blurry images when increasing the coefficient α_t. This is preditable as the guidance term in SVGD involves a weighted sum of scores of nearby samples, which will steer the samples toward the mean of nearby samples.

(a) I.I.D. (b) particle guidance (pixel) (c) particle guidance (feature)

Figure 4: Text prompt: Captain Marvel Exclusive Ccxp Poster Released Online By Marvel



(a) I.I.D. (b) particle guidance (pixel) (c) particle guidance (feature)

Figure 5: Text prompt: Portrait of Tiger in black and white by Lukas Holas

(a) I.I.D. (b) particle guidance (pixel) (c) particle guidance (feature)

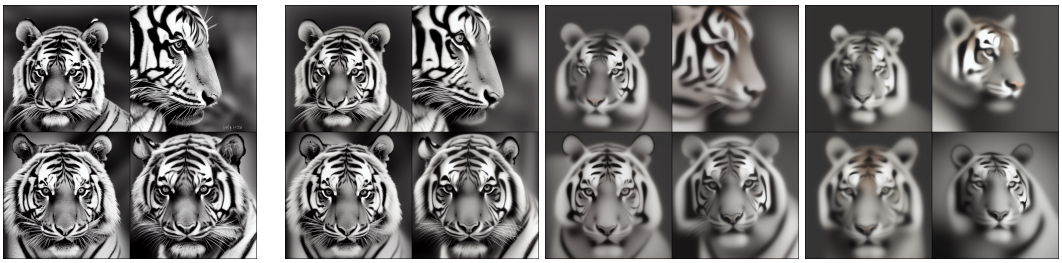
Figure 6: Text prompt: VAN GOGH CAFE TERASSE copy.jpg

(a) I.I.D. (b) particle guidance (pixel) (c) particle guidance (feature)

Figure 7: Text prompt: A transparent sculpture of a duck made out of glass

(a) I.I.D. (b) particle guidance (pixel) (c) particle guidance (feature)

Figure 8: Text prompt: A unicorn in a snowy forest



(a) $\alpha_t = 0.1$ (b) $\alpha_t = 0.3$ (c) $\alpha_t = 1$ (d) $\alpha_t = 2$

Figure 9: SVGD guidance, with varying α_t