BNEM: A Boltzmann Sampler Based on Bootstrapped Noised Energy Matching

RuiKang OuYang[*]
University of Cambridge
ro352@cam.ac.uk

Bo Qiang[*]
University of Washington
bqiang@uw.edu

José Miguel Hernández-Lobato
University of Cambridge
jmh233@cam.ac.uk

Abstract

Developing an efficient sampler capable of generating independent and identically distributed (IID) samples from a Boltzmann distribution is a crucial challenge in scientific research, *e.g.* molecular dynamics. In this work, we intend to learn neural samplers given energy functions instead of data sampled from the Boltzmann distribution. By learning the energies of the noised data, we propose a diffusion-based sampler, NOISED ENERGY MATCHING, which theoretically has lower variance and more complexity compared to related works. Furthermore, a novel bootstrapping technique is applied to NEM to balance between bias and variance. We evaluate NEM and BNEM on a 2-dimensional 40 Gaussian Mixture Model (GMM) and a 4-particle double-well potential (DW-4). The experimental results demonstrate that BNEM can achieve state-of-the-art performance while being more robust.

1 Introduction

Given energy functions $\mathcal{E}(x)$, samplers for unnormalized target distributions *i.e.* Boltzmann distribution $\mu_{\text{target}} \propto \exp(-\mathcal{E}(x))$ are fundamental in probabilistic modeling and physical systems simulation. For example, predicting the folding of proteins could be formalized as sampling from a Boltzmann distribution (Śledź and Caflisch, 2018), where the energies are defined by inter-atomic forces (Case et al., 2021). Efficient samplers for many-particle systems could have the potential to speed up drug discovery (Zheng et al., 2024) and material design (Komanduri et al., 2000).

However, most related works face challenges with scalability to high dimensions and are time-consuming. To solve these problems, a previous work(Akhound-Sadegh et al., 2024) proposes Iterated Denoising Energy Matching (iDEM), which is not only computationally tractable but also guarantees good coverage of all modes. iDEM proposes a bi-level training scheme that iteratively generates samples from the learned sampler and performs score matching to the Monte Carlo estimated target. While Woo and Ahn (2024) proposes a variant, iEFM, that targets the MC estimated vector fields in a Flow Matching fashion, which we found can be linked through Tweedie's formula (Efron, 2011) shown in Appendix G. Nevertheless, both iDEM and iEFM require large numbers of MC samplings for score estimation to minimize the variance at large time steps, which holds disadvantages for complicated energy functions.

In this work, we propose an energy-based variant of DEM, NOISED ENERGY MATCHING (NEM), which targets the MC estimated noised energies instead of scores. Despite a need to differentiate the neural network when simulating the diffusion sampler, NEM is theoretically found to be targeting less noisy objectives compared with iDEM. Our method demonstrates better performance on a 2-dimensional 40 Gaussian Mixture Model (GMM) and a 4-particle double-well potential (DW-4) (Klein et al., 2023b). To further reduce the variance, BOOTSTRAP NEM (BNEM) is proposed in our

[*]Equal Contribution

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

work, which estimates high noise-level energies by bootstrapping from slightly lower noise-level estimators. We theoretically found that BNEM trades bias to the variance of training target, which empirically leads to state-of-the-art performance.

2 Related works

Conventional methods for many-body system simulation are based on molecular dynamics (MD) Leimkuhler and Matthews (2012) techniques which require long simulation times. Others leverage Monte Carlo techniques *e.g.* Annealed Importance Sampling (AIS) (Lyman and Zuckerman, 2007), which are computationally expensive. To speed up this process, a promising remedy is training a deep generative model, *i.e.* a neural sampler. Unlike data-driven tasks, simply minimizing the reverse Kullback-Leibler (KL) divergence for samplers can lead to mode-seeking behavior. Noé et al. (2019) address this problem by using the Boltzmann Generator, which minimizes the combination of forward and reverse KL divergence.

Inspired by the rapid development of deep generative models, *e.g.* diffusion models (Song and Ermon, 2019; Ho et al., 2020), pseudo samples could be generated from an arbitrary prior distribution. Then, we can train the neural samplers by matching these sample trajectories, as in Path Integral Sampler (PIS)(Zhang and Chen, 2022) and Denoising Diffusion Sampler (DDS) (Vargas et al., 2023). Midgley et al. (2023) further deploy a replay buffer for the trajectories while proposing an α-divergence as the objective to avoid mode-seeking. However, these methods require simulation during training. DDS and PIS require the diffusion trajectories to minimize the KL divergence between prior distribution and target distribution over them, while FAB requires simulating the AIS to correct the weights. This poses challenges for scaling up to higher dimensional tasks.

3 Methods

3.1 Denoising diffusion based Boltzmann sampler

In this work, we consider training an energy-based diffusion sampler corresponding to a variance exploding (VE) noising process defined by $dx_t = g(t)dw_t$, where $t \in [0, 1]$, $g(t)$ is a function of time and w_t is a Brownian motion. Then the corresponding reverse SDE with Brownian motion $\bar{w}_t$ is $dx_t = -g^2(t)\nabla \log p_t(x_t)d_t + g(t)d\bar{w}_t$, where p_t is the marginal distribution of the diffusion process that starts at $p_0 = \mu_{target}$. Given access to the system energy $\mathcal{E}(x)$ and the perturbation kernel $q_t(x_t|x_0) = \mathcal{N}(x_t; x_0, \sigma_t^2)$, where $\exp(-\mathcal{E}(x)) \propto p_0(x)$, one can obtain the marginal distribution p_t

$$p_t(x_t) \propto \int \exp(-\mathcal{E}(x_0))\mathcal{N}(x_t; x_0, \sigma_t^2 I)\mathrm{d}x_0 = \mathbb{E}_{\mathcal{N}(x;x_t,\sigma_t^2 I)}[\exp(-\mathcal{E}(x))] \tag{1}$$

iDEM is proposed to train a score network s_θ to match an MC score estimator S_K

$$S_K(x_t, t) := \nabla \log \frac{1}{K}\sum_{i=1}^{K} \exp(-\mathcal{E}(x_{0|t}^{(i)})), \quad x_{0|t}^{(i)} \sim \mathcal{N}(x; x_t, \sigma_t^2 I) \tag{2}$$

Instead of the score network, here we consider training an energy network, $E_\theta(x, t)$, to approximate the following noised energy, *i.e.* the energy of noised data

$$\mathcal{E}_t(x) := -\log \mathbb{E}_{\mathcal{N}(x;x_t,\sigma_t^2 I)}[\exp(-\mathcal{E}(x))] \tag{3}$$

where $\exp(-\mathcal{E}_t(x_t)) \propto p_t(x_t)$. Then the score of marginal distribution at t can be approximated by differentiating the energy network w.r.t its input x, i.e. $s_\theta(x, t) = -\nabla_x E_\theta(x, t)$. We train NEM with a bi-level scheme by iterating: (a) an outer-loop that simulates the energy-based diffusion sampler and updates the replay buffer; (b) a simulation-free inner-loop that matches the noised energies of samples in the replay buffer. Detailed implementation is discussed in Appendix A.

3.2 Noised Energy Matching

NEM targets a novel MC energy estimator that approximates the noised energies

$$E_K(x_t, t) : = -\log \frac{1}{K}\sum_{i=1}^{K} \exp(-\mathcal{E}(x_{0|t}^{(i)})), \quad x_{0|t}^{(i)} \sim \mathcal{N}(x; x_t, \sigma_t^2 I) \tag{4}$$

where Equation 4 can be implemented by the LogSumExp trick for stability. We characterize the bias of E_K with the following proposition:

Proposition 1 *If* $\exp(-\mathcal{E}(x_{0|t}^{(i)}))$ *is sub-Gaussian, then there exists a constant* $\tilde{c}(x_t)$ *such that with probability* $1-\delta$ *over* $x_{0|t}^{(i)} \sim \mathcal{N}(x_t, \sigma_t^2)$*, we have*

$$\|E_K(x_t,t) - \mathcal{E}_t(x_t)\| \leq \frac{\tilde{c}(x_t)\sqrt{\log(1/\delta)}}{\sqrt{K}} \tag{5}$$

with $c(x_t)/\tilde{c}(x_t) = 2(1+\|\nabla\mathcal{E}_t(x_t)\|)$*, where*

$$\|S_K(x_t,t) - S_t(x_t)\| \leq \frac{c(x_t)\sqrt{\log(1/\delta)}}{\sqrt{K}}. \tag{6}$$

Proposition 1 shows that the training target of NEM has a smaller bound of bias compared with the one of DEM, especially on regions with steep gradient, *i.e.* large $\|\nabla\mathcal{E}_t(x))\|$. The complete proof is given in Appendix B. Besides, we characterize the variance of S_K and E_K as follows:

Proposition 2 *If* $\exp(-\mathcal{E}(x_{0|t}^{(i)}))$ *is sub-Gaussian and* $\|\nabla\exp(-\mathcal{E}(x_{0|t}^{(i)}))\|$ *is bounded, the total variance of the MC score estimator* S_K *is consistently larger than that of the MC energy estimator* E_K *in regions associated with low energies, with*

$$\frac{\textbf{\textit{tr}}\left(Cov[S_K(x_t,t)]\right)}{Var[E_K(x_t,t)]} = 4(1+\|\nabla\mathcal{E}_t(x_t)\|)^2. \tag{7}$$

In regions associated with high energies, $Var[E_K(x_t,t)] < Var[S_K(x_t,t)]$ *holds when the target energy* $\mathcal{E}(x_t)$ *is positively related to at least one element of the score* $\nabla\mathcal{E}(x_t)$*.*

It demonstrates that the MC energy estimator can provide a less noisy training signal than the score one, showcasing the theoretical advantage of NEM compared with DEM. The complete proof is provided in Appendix C.

3.3 Bootstrap NEM: an improvement with bootstrapped energy estimation

Bootstrap NEM, or BNEM, targets a novel MC energy estimator at a high noise level that is bootstrapped from the learned energies at a slightly lower noise level. Intuitively, the variance of E_K exploded at a high noise level as a result of the VE noising process; while we can estimate these energies from the ones at a low noise level rather than the system energy to reduce variance. Suppose a low-level noise-convolved energy is well learned, say $\mathcal{E}_s$, we can construct a bootstrap energy estimator at higher noise level t by

$$E_K(x_t,t,s;\phi) := -\log\frac{1}{K}\sum_{i=1}^{K}\exp(-E_\phi(x_{s|t}^{(i)},s)), \quad x_{s|t}^{(i)} \sim \mathcal{N}\left(x; x_s, (\sigma_t^2-\sigma_s^2)I\right) \tag{8}$$

where $s<t$ and ϕ a well learned neural network for any $u\in[0,s]$. We show that this bootstrap energy estimator is trading variance to bias, in terms of training target, characterized in the following proposition:

Proposition 3 *Given a bootstrap trajectory* $\{s_i\}_{i=0}^n$ *such that* $\sigma_{s_i}^2 - \sigma_{s_{i-1}}^2 \leq \kappa$*, where* $s_0=0$*,* $s_n=1$ *and* $\kappa>0$ *is a small number, suppose* E_θ *is incrementally optimized from* 0 *to* 1 *as follows: if* $t\in[s_i,s_{i+1}]$*, then* E_θ *targets an energy estimator bootstrapped from* $s\in[s_{i-1},s_i]$ *using Eq. 8. Let* $v_{yz}(x_z) = Var_{\mathcal{N}(x_z,(\sigma_z^2-\sigma_y^2)I)}[\exp(-\mathcal{E}(x))]$ *and* $m_z(x_z) = \exp(-\mathcal{E}_z(x_z))$*, where* $0\leq y<z\leq 1$ *and* $\mathcal{E}_z$ *is the ground truth noised energy at time* z*. The variance of the bootstrap energy estimator is given by*

$$Var[E_K(x_t,t,s;\theta)] = \frac{v_{st}(x_t)}{v_{0t}(x_t)}Var[E_K(x_t,t)] \tag{9}$$

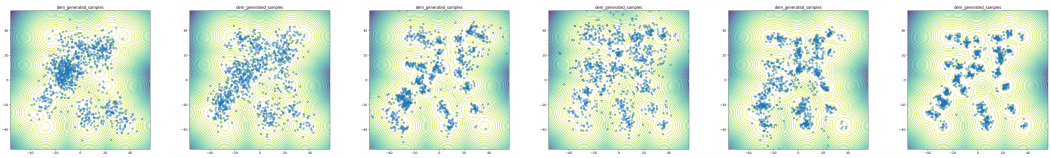
and the bias of $E_K(x_t,t,s;\theta)$ *is given by*

$$\frac{v_{0t}(x_t)}{2m_t^2(x_t)K^{n+1}} + \sum_{j=1}^{n}\frac{v_{0s_j}(x_t)}{2m_{s_j}^2(x_t)K^j}. \tag{10}$$

A detailed discussion and proof are given in Appendix D. Proposition 3 demonstrates that the bootstrap energy estimator, which estimates the noised energies by sampling in a smaller ball, can reduce the variance of the training target while this new target can introduce accumulated bias.

Table 1: Neural sampler performance comparison for GMM-40 and DW-4 energy function. we measured the performance using data Wasserstein-2 distance (x-$\mathcal{W}_2$), Energy Wasserstein-2 distance ($\mathcal{E}$-$\mathcal{W}_2$), and Total Variation (TV). * indicates divergent training. **Bold** indicates the best values and underline indicates the second best ones.

Energy →	**GMM-40** ($d = 2$)			**DW-4** ($d = 8$)		
Sampler ↓	**x-$\mathcal{W}_2$↓**	**$\mathcal{E}$-$\mathcal{W}_2$↓**	**TV↓**	**x-$\mathcal{W}_2$↓**	**$\mathcal{E}$-$\mathcal{W}_2$↓**	**TV↓**
DDS	11.69	86.69	0.944	0.701	109.8	0.429
PIS	5.806	76.35	0.940	*	*	*
FAB	<u>3.828</u>	<u>64.23</u>	<u>0.824</u>	0.614	211.5	0.359
iDEM	8.512	562.7	0.909	0.532	2.109	0.161
NEM (ours)	5.192	85.05	0.906	<u>0.489</u>	<u>0.999</u>	<u>0.145</u>
BNEM (ours)	**3.652**	**2.973**	**0.830**	**0.467**	**0.458**	**0.134**



(a) DDS (b) PIS (c) FAB (d) iDEM (e) NEM (f) BNEM

Figure 1: Sampled points from samplers applied to GMM-40 potentials, with the ground truth represented by contour lines.

4 Experiments

We evaluate our methods and baseline models on 2 potentials, the Gaussian mixture model (GMM) and the 4-particle double-well (DW-4) potential. We provide the full description of the experimental setting in Appendix H.

Baseline. We compare NEM and BNEM to following recent works: Denoising Diffusion Sampler (DDS)(Vargas et al., 2023), Path Integral Sampler (PIS)(Zhang and Chen, 2022), Flow Annealed Bootstrap (FAB)(Midgley et al., 2023) and Iterated Denoising Energy Matching (iDEM)(Akhound-Sadegh et al., 2024). For a fair comparison, we set the number of integration steps to 100 and the number of MC samples to 100. For baselines, we train all samplers using an NVIDIA-A100 GPU.

Architecture. We implement the same network architecture (MLP for GMM and EGNN for DW-4) for all baselines. To ensure a similar number of parameters for each sampler, if the score network is parameterized by $s_\theta(x,t) = f_\theta(x,t)$, the energy network is set to be $E_\theta(x,t) = \sum f_\theta(x,t) + c$ with a learnable scalar c. Furthermore, this setting ensures SE(3) invariance for the energy network.

Metrics. We use x-$\mathcal{W}_2$, $\mathcal{E}-\mathcal{W}_2$, and TV as metrics. To compute $\mathcal{W}_2$ and TV, we use pre-generated samples as datasets: (a) For GMM, we sample from the ground truth distribution; (b) For DW-4, we use samples from Klein et al. (2023b). .

Main results. We report x-$\mathcal{W}_2$, $\mathcal{E}-\mathcal{W}_2$ and TV for GMM and DW-4 potentials in Table 1. It shows that when the reverse SDE integration step is limited to 100, NEM can outperform iDEM on all metrics in each task, showcasing its robustness. Furthermore, BNEM can outperform NEM by targeting the bootstrap energy estimator, achieving state-of-the-art performance. In addition, the visualized results for GMM are provided in Figure 1, which shows that NEM can generate samples more concentrated to the GMM modes than iDEM and BNEM can further boost this concentration.

5 Conclusion

In this work, we introduce NEM and BNEM, neural samplers for Boltzmann distribution and equilibrium systems like many-body systems. NEM uses a novel Monte Carlo energy estimator with reduced bias and variance. BNEM builds on NEM, employing an energy estimator bootstrapped from lower noise-level data, theoretically trading bias for variance. Empirically, BNEM achieves state-of-the-art results on the GMM and DW4 benchmarks. Future work will focus on scaling these methods to higher-dimensional tasks like Lennard-Jones potential and alanine dipeptide.

Acknowledgements

José Miguel Hernández-Lobato acknowledges support from a Turing AI Fellowship under grant EP/V023756/1. RKOY acknowledges resources provided by the Cambridge Service for Data Driven Discovery (CSD3) operated by the University of Cambridge Research Computing Service (www.csd3.cam.ac.uk), provided by Dell EMC and Intel using Tier-2 funding from the Engineering and Physical Sciences Research Council (capital grant EP/T022159/1), and DiRAC funding from the Science and Technology Facilities Council (www.dirac.ac.uk).

References

Akhound-Sadegh, T., Rector-Brooks, J., Bose, A. J., Mittal, S., Lemos, P., Liu, C.-H., Sendera, M., Ravanbakhsh, S., Gidel, G., Bengio, Y., Malkin, N., and Tong, A. (2024). Iterated denoising energy matching for sampling from boltzmann densities.

Case, D. A., Aktulga, H. M., Belfon, K., Ben-Shalom, I., Brozell, S. R., Cerutti, D. S., Cheatham III, T. E., Cruzeiro, V. W. D., Darden, T. A., Duke, R. E., et al. (2021). *Amber 2021*. University of California, San Francisco.

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

Flamary, R., Courty, N., Gramfort, A., Alaya, M. Z., Boisbunon, A., Chambon, S., Chapel, L., Corenflos, A., Fatras, K., Fournier, N., Gautheron, L., Gayraud, N. T., Janati, H., Rakotomamonjy, A., Redko, I., Rolet, A., Schutz, A., Seguy, V., Sutherland, D. J., Tavenard, R., Tong, A., and Vayer, T. (2021). Pot: Python optimal transport. *Journal of Machine Learning Research*, 22(78):1–8.

Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851.

Karras, T., Aittala, M., Aila, T., and Laine, S. (2022). Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577.

Klein, L., Krämer, A., and Noe, F. (2023a). Equivariant flow matching. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S., editors, *Advances in Neural Information Processing Systems*, volume 36, pages 59886–59910. Curran Associates, Inc.

Klein, L., Krämer, A., and Noé, F. (2023b). Equivariant flow matching.

Köhler, J., Klein, L., and Noe, F. (2020). Equivariant flows: Exact likelihood generative learning for symmetric densities. In III, H. D. and Singh, A., editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5361–5370. PMLR.

Komanduri, R., Chandrasekaran, N., and Raff, L. (2000). Md simulation of nanometric cutting of single crystal aluminum–effect of crystal orientation and direction of cutting. *Wear*, 242(1-2):60–88.

Leimkuhler, B. and Matthews, C. (2012). Rational construction of stochastic numerical methods for molecular sampling. *Applied Mathematics Research eXpress*.

Lyman, E. and Zuckerman, D. M. (2007). Annealed importance sampling of peptides. *The Journal of chemical physics*, 127(6).

Midgley, L. I., Stimper, V., Simm, G. N. C., Schölkopf, B., and Hernández-Lobato, J. M. (2023). Flow annealed importance sampling bootstrap.

Moore, J. H., Cole, D. J., and Csanyi, G. (2024). Computing hydration free energies of small molecules with first principles accuracy.

Noé, F., Olsson, S., Köhler, J., and Wu, H. (2019). Boltzmann generators – sampling equilibrium states of many-body systems with deep learning.

Owen, A. B. (2023). *Practical Quasi-Monte Carlo Integration*. `https://artowen.su.domains/mc/practicalqmc.pdf`.

Śledź, P. and Caflisch, A. (2018). Protein structure-based drug design: from docking to molecular dynamics. *Current opinion in structural biology*, 48:93–102.

Song, Y. and Ermon, S. (2019). Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. (2020). Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*.

Vargas, F., Grathwohl, W., and Doucet, A. (2023). Denoising diffusion samplers.

Woo, D. and Ahn, S. (2024). Iterated energy-based flow matching for sampling from boltzmann densities.

Zhang, Q. and Chen, Y. (2022). Path integral sampler: a stochastic control approach for sampling.

Zheng, S., He, J., Liu, C., Shi, Y., Lu, Z., Feng, W., Ju, F., Wang, J., Zhu, J., Min, Y., et al. (2024). Predicting equilibrium distributions for molecular systems with deep learning. *Nature Machine Intelligence*, pages 1–10.

A Training Details

Algorithm 1 describes NEM in detail. The key difference in training the BNEM is only in the inner-loop.

To train the BNEM, it's crucial that the bootstrap energy estimation at t can be accurate only when the noised energy at s is well-learned. Therefore, we need to sequentially learn energy at small t using the original MC energy estimator then refines the estimator using bootstrap for large t, which is inefficient. To be more efficient, we adapt a rejection training scheme: (a) given s and t, we first compute the loss of targeting the MC energy estimator (4), l_s and l_t; (b) These losses indicate how does the energy network fit the noised energies at different times, and therefore compute $\alpha = l_s/l_t$ as an indicator; (c) with probability α, we accept targeting a energy estimator bootstrapped from s and otherwise we stick to target the original MC estimator. We provide a full description of the inner-loop of BNEM training in Algorithm 2.

B Proof of Proposition 1

Proposition 1 *If* $\exp(-\mathcal{E}(x_{0|t}^{(i)}))$ *is sub-Gaussian, then there exists a constant* $\tilde{c}(x_t)$ *such that with probability* $1-\delta$ *over* $x_{0|t}^{(i)} \sim \mathcal{N}(x_t, \sigma_t^2)$*, we have*

$$\|E_K(x_t, t) - \mathcal{E}_t(x_t)\| \leq \frac{\tilde{c}(x_t)\sqrt{\log(1/\delta)}}{\sqrt{K}} \tag{11}$$

with $c(x_t)/\tilde{c}(x_t) = 2(1 + \|\nabla \mathcal{E}_t(x_t)\|)$*, where*

$$\|S_K(x_t, t) - S_t(x_t)\| \leq \frac{c(x_t)\sqrt{\log(1/\delta)}}{\sqrt{K}}. \tag{12}$$

Algorithm 1 Iterated training for Noised Energy Matching

Require: Network E_θ, Batch size b, Noise schedule σ_t^2, Base distribution p_1, Num. integration steps L, Replay buffer $\mathcal{B}$, Max Buffer Size $|\mathcal{B}|$, Num. MC samples K

while Outer-Loop **do**
  $\{x_1\}_{i=1}^b \sim p_1(x_1)$
  $\{x_0\}_{i=1}^b \leftarrow \text{sde.int}(\{x_1\}_{i=1}^b, -\nabla E_\theta, L)$ ▷ Simulate the reverse SDE for sampling
  $\mathcal{B} = (\mathcal{B} \cup \{x_0\}_{i=1}^b)$ ▷ Update Buffer $\mathcal{B}$
  while Inner-Loop **do**
    $x_0 \leftarrow \mathcal{B}.\text{sample}()$ ▷ Uniform sampling from $\mathcal{B}$
    $t \sim \mathcal{U}(0,1), x_t \sim \mathcal{N}(x_0, \sigma_t^2)$
    $\mathcal{L}_{\text{NEM}}(x_t, t) = \|E_K(x_t, t) - E_\theta(x_t, t)\|^2$
    $\theta \leftarrow \text{Update}(\theta, \nabla_\theta \mathcal{L}_{\text{NEM}})$
  end while
end while

Ensure: s_θ

Algorithm 2 Inner-loop of Bootstrap Noised Energy Matching training

Require: Network E_θ, Batch size b, Noise schedule σ_t^2, Replay buffer $\mathcal{B}$, Num. MC samples K

while Inner-Loop **do**
  $x_0 \leftarrow \mathcal{B}.\text{sample}()$ ▷ Uniform sampling from $\mathcal{B}$
  $t \sim \mathcal{U}(0,1), x_t \sim \mathcal{N}(x_0, \sigma_t^2)$
  $n \leftarrow \arg\{i : t \in [t_i, t_{i+1}]\}$ ▷ Identify the time split range of t
  $s \sim \mathcal{U}(t_{n-1}, t_n), x_s \sim \mathcal{N}(x_0, \sigma_s^2)$
  $l_s(x_s) \leftarrow \|E_K(x_s, s) - E_\theta(x_s, s)\|^2/\sigma_s^2$
  $l_t(x_t) \leftarrow \|E_K(x_t, t) - E_\theta(x_t, t)\|^2/\sigma_t^2$
  $\alpha \leftarrow \min(1, l_t(x_t)/l_s(x_s))$
  with probability α,
    $\mathcal{L}_{\text{BNEM}}(x_t, t) = \|E_K(x_t, t, s; \text{StopGrad}(\theta)) - E_\theta(x_t, t)\|^2$
  Otherwise, ▷ Use MC estimator if the model is not well trained
    $\mathcal{L}_{\text{BNEM}}(x_t, t) = \|E_K(x_t, t) - E_\theta(x_t, t)\|^2$
  $\theta \leftarrow \text{Update}(\theta, \nabla_\theta \mathcal{L}_{\text{BNEM}})$
end while

Ensure: E_θ

Proof. We first introduce the error bound of the MC score estimator S_K, where $S_K = \nabla E_K$, proposed by Akhound-Sadegh et al. (2024) as follows

$$\|S_K(x_t, t) - S(x_t, t)\| \leq \frac{2C\sqrt{\log(\frac{2}{\delta})}(1 + \|\nabla \mathcal{E}_t(x_t)\|)\exp(\mathcal{E}_t(x_t))}{\sqrt{K}} \tag{13}$$

which assumes that $\exp(-\mathcal{E}(x_{0|t}^{(i)}))$ is sub-Gaussian. Let's define the following variables

$$m_t(x_t) = \exp(-\mathcal{E}_t(x_t)) \tag{14}$$

$$v_{st}(x_t) = \text{Var}_{\mathcal{N}(x_t, (\sigma_t^2 - \sigma_s^2)I)}[\exp(-\mathcal{E}(x))] \tag{15}$$

By the sub-Gaussianess assumption, it's easy to show that the constant term C in Equation 13 is $C = \sqrt{2v_{0t}(x_t)}$. Notice that E_K is a logarithm of an unbiased estimator . By the sub-Gaussian assumption, one can derive that E_K is also sub-Gaussian. Furthermore, it's mean and variance can be derived by employing a first-order Taylor expansion:

$$\mathbb{E}[E_K(x_t, t)] \approx \mathcal{E}_t(x_t) + \frac{v_{0t}(x_t)}{2m_t^2(x_t)K} \tag{16}$$

$$\text{Var}[E_K(x_t, t)] \approx \frac{v_{0t}(x_t)}{m_t^2(x_t)K} \tag{17}$$

And one can obtain its concentration inequality by incorporating the sub-Gaussianess

$$\|E_K(x_t,t) - \mathbb{E}[E_K(x_t,t)]\| \leq \sqrt{2\frac{v_{0t}(x_t)}{m_{0t}^2(x_t)K}\log\frac{2}{\delta}} \tag{18}$$

By using the above Inequality 18 and the triangle inequality

$$\begin{aligned}
\|E_K(x_t,t) - \mathcal{E}_t(x_t)\| &\leq \|\log E_K(x_t,t) - \mathbb{E}[E_K(x_t,t)]\| + \|\mathbb{E}[E_K(x_t,t)] - \mathcal{E}_t(x_t)\| && (19)\\
&= \|\log E_K(x_t,t) - \mathbb{E}[E_K(x_t,t)]\| + \frac{v_{0t}(x_t)}{2m_t^2(x_t)K} && (20)\\
&\leq \sqrt{2\frac{v_{0t}(x_t)}{m_t^2(x_t)K}\log\frac{2}{\delta}} + \frac{v_{0t}(x_t)}{2m_t^2(x_t)K} && (21)\\
&= \frac{C\sqrt{\log\frac{2}{\delta}}\exp(\mathcal{E}_t(x_t))}{\sqrt{K}} + O(1/K) && (22)\\
&= \frac{c(x_t)}{2(1+\|\nabla\mathcal{E}_t(x_t)\|)}\frac{\sqrt{\log\frac{1}{\delta}}}{\sqrt{K}} && (23)
\end{aligned}$$

Therefore, we have $c(x_t) = 2(1+\|\nabla\mathcal{E}_t(x_t)\|)\tilde{c}(x_t)$. It demonstrates a less biased estimator, which, what's more, doesn't require a sub-Gaussianess assumption over $\|\nabla\mathcal{E}(x_{0|t}^{(i)})\|$. □

C Proof of Proposition 2

Proposition 2 *If $\exp(-\mathcal{E}(x_{0|t}^{(i)}))$ is sub-Gaussian and $\|\nabla\exp(-\mathcal{E}(x_{0|t}^{(i)}))\|$ is bounded, the total variance of the MC score estimator S_K is consistently larger than that of the MC energy estimator E_K in regions associated with high energies, with*

$$\frac{\textbf{\textit{tr}}\left(Cov[S_K(x_t,t)]\right)}{Var[E_K(x_t,t)]} = 4(1+\|\nabla\mathcal{E}_t(x_t)\|)^2. \tag{24}$$

In regions associated with high energies, $Var[E_K(x_t,t)] < Var[S_K(x_t,t)]$ holds when the target energy $\mathcal{E}(x_t)$ is positively related to at least one element of the score $\nabla\mathcal{E}(x_t)$.

Proof. We split the proof into two parts: low-energy region and high-energy one. The proof in the low-energy region requires only the aforementioned sub-Gaussianess and bounded assumptions, while the one in the high-energy region requires an additional constraint which will be clarified later. Review that S_K can be expressed as an importance-weighted estimator as follows:

$$S_K(x_t,t) = \frac{\frac{1}{K}\sum_{i=1}^K \nabla\exp(-\mathcal{E}(x_{0|t}^{(i)}))}{\frac{1}{K}\sum_{i=1}^K \exp(-\mathcal{E}(x_{0|t}^{(i)}))} \tag{25}$$

Let $\|\nabla\exp(-\mathcal{E}(x_{0|t}^{(i)}))\| \leq M$, where $M > 0$. Since a bounded variable is sub-Gaussian, this assumption resembles a sub-Gaussianess assumption of $\|\nabla\exp(-\mathcal{E}(x_{0|t}^{(i)}))\|$. Then each element of $\|\nabla\exp(-\mathcal{E}(x_{0|t}^{(i)}))\|$, i.e. $\nabla\exp(-\mathcal{E}(x_{0|t}^{(i)}))[j]$, is bounded by M. And therefore $\nabla\exp(-\mathcal{E}(x_{0|t}^{(i)}))[j]$ are sub-Gaussian.

In low-energy regions. $\exp(-\mathcal{E}(x))$ is concentrated away from 0 as $\mathcal{E}(x)$ is small. Then, there exists a constant c such that $\exp(-\mathcal{E}(x_{0|t}^{(i)})) \geq c > 0$ and thus for each element $j = 1,..,d$:

$$\begin{aligned}
\|S_K(x_t,t)[j]\| &= \left\| \frac{\frac{1}{K}\sum_{i=1}^K \nabla\exp(-\mathcal{E}(x_{0|t}^{(i)}))[j]}{\frac{1}{K}\sum_{i=1}^K \exp(-\mathcal{E}(x_{0|t}^{(i)}))} \right\| && (26)\\
&\leq \frac{\|\sum_{i=1}^K \nabla\exp(-\mathcal{E}(x_{0|t}^{(i)}))[j]\|}{Kc} \leq M/c && (27)
\end{aligned}$$

therefore, the j^{th} element of S_K, i.e.$S_K[j]$, is bounded by M/c, suggesting it is sub-Gaussian. While Inequality 13 can be expressed as

$$\sqrt{\sum_{j=1}^{d}(\mathcal{S}_K(x_t,t)[j]-S_t(x_t)[j])^2} \leq \frac{2\sqrt{2v_{0t}(x_t)\log(\frac{2}{\delta})}(1+\|\nabla\mathcal{E}_t(x_t)\|)}{m_t(x_t)\sqrt{K}} \tag{28}$$

We can roughly derive a bound elementwisely

$$|\mathcal{S}_K(x_t,t)[j]-S_t(x_t)[j]| \leq \frac{2\sqrt{2v_{0t}(x_t)\log(\frac{2}{\delta})}(1+\|\nabla\mathcal{E}_t(x_t)\|)}{m_t(x_t)\sqrt{Kd}} \tag{29}$$

which suggests that we can approximate the variance of $S_K(x_t,t)[j]$ by leveraging its sub-Gaussianess

$$\text{Var}(S_K(x_t,t)[j]) \approx \frac{4v_{0t}(x_t)(1+\|\nabla\mathcal{E}_t(x_t)\|)^2}{m_t^2(x_t)Kd} \tag{30}$$

Therefore, according to Equation 17 we can derive that

$$\begin{aligned}
\textbf{tr}(\text{Cov}[S_K(x_t,t)]) &= \sum_{j=1}^{d}\text{Var}[S_K(x_t,t)[j]] && (31)\\
&= \frac{4v_{0t}(x_t)(1+\|\nabla\mathcal{E}_t(x_t)\|)^2}{m_t^2(x_t)K} && (32)\\
&= 4(1+\|\nabla\mathcal{E}_t(x_t)\|)^2\text{Var}[E_K(x_t,t)] && (33)
\end{aligned}$$

In high-energy region. we assume that there exists a direction with a large norm pointing to low energy regions, i.e. $\exists j$ such that $\mathcal{E}(x)$ are positively related to $\nabla\mathcal{E}(x)[j]$. According to Section 9.2 in Owen (2023), the asymptotic variance of a self-normalized importance sampling estimator is given by:

$$\begin{aligned}
\mu &= \mathbb{E}_q[f(X)] && (34)\\
\tilde{\mu}_q &= \frac{\sum_{i=1}^{K}w_i f_i}{\sum_{i=1}^{K}w_i} && (35)\\
\text{Var}(\tilde{\mu}_q) &\approx \frac{1}{K}\mathbb{E}_q[w(X)]^{-2}\mathbb{E}_q[w(X)^2(f(X)-\mu)^2] && (36)
\end{aligned}$$

By substituting $\tilde{\mu}_q = S_K(x_t,t)[j]$, $f(X) = -\nabla\mathcal{E}(X)[j]$, $w(X) = \exp(-\mathcal{E}(X))$, $q = N(x;x_t,\sigma_t^2)$, $\mathbb{E}_q[w(X)] = m_t(x_t)$ and $\mathbb{E}_q[w^2(X)] = v_{0t}(x_t) + m_t^2(x_t)$, as well as using $w(X)$ and $f(X)$ are positive related, we have:

$$\begin{aligned}
\text{Var}[S_K(x_t,t)[j]] &\geq \frac{1}{K}\mathbb{E}_q[w(X)]^{-2}\mathbb{E}_q[w^2(X)]\mathbb{E}_q[(f(X)-\mu)^2] && (37)\\
&= \frac{v_{0t}(x_t)+m_t^2(x_t)}{m_t^2(x_t)K}\text{Var}_q[\nabla\mathcal{E}(x)[j]] && (38)\\
& && (39)
\end{aligned}$$

Therefore, if we further have a large variance over the system score at this region, i.e. $\text{Var}_q[\nabla\mathcal{E}(x)[j]] > 1$, then we have

$$\text{Var}[S_K(x_t,t)[j]] > \frac{v_{0t}(x_t)}{m_t^2(x_t)K} = \text{Var}[E_K(x_t,t)] \tag{40}$$

and thus $\textbf{tr}(\text{Cov}[S_K(x_t,t])) > \text{Var}[E_K(x_t,t)]$ holds. □

D Proof of Proposition 3

Proposition 3 *Given a bootstrap trajectory $\{s_i\}_{i=0}^n$ such that $\sigma_{s_i}^2 - \sigma_{s_{i-1}}^2 \leq \kappa$, where $s_0 = 0$, $s_n = 1$ and $\kappa > 0$ is a small number, suppose E_θ is incrementally optimized from 0 to 1 as follows: if*

$t \in [s_i, s_{i+1}]$, then E_θ targets an energy estimator bootstrapped from $s \in [s_{i-1}, s_i]$ using Eq. 8. Let $v_{yz}(x_z) = Var_{\mathcal{N}(x_z,(\sigma_z^2-\sigma_y^2)I)}[\exp(-\mathcal{E}(x))]$ and $m_z(x_z) = \exp(-\mathcal{E}_z(x_z))$, where $0 \le y < z \le 1$ and $\mathcal{E}_z$ is the ground truth noised energy at time z. The variance of the bootstrap energy estimator is given by

$$Var[E_K(x_t, t, s; \theta)] = \frac{v_{st}(x_t)}{v_{0t}(x_t)} Var[E_K(x_t, t)] \tag{41}$$

and the bias of $E_K(x_t, t, s; \theta)$ is given by

$$\frac{v_{0t}(x_t)}{2m_t^2(x_t)K^{n+1}} + \sum_{j=1}^{n} \frac{v_{0s_j}(x_t)}{2m_{s_j}^2(x_t)K^j}. \tag{42}$$

Proof. The variance of $E_K(x_t, t, s; \phi)$ can be simply derived by leveraging the variance of a sub-Gaussian random variable similar to Equation 17. While the entire proof for bias of $E_K(x_t, t, s; \phi)$ is organized as follows:

1. we first show the bias of Bootstrap(1) estimator, which is bootstrapped from the system energy
2. we then show the bias of Bootstrap(n) estimator, which is bootstrapped from a lower level noise convolved energy recursively, by induction.

D.1 Bootstrap(1) estimator

The Sequential estimator and Bootstrap(1) estimator are defined by:

$$E_K^{\text{Seq}}(x_t, t) := -\log \frac{1}{K} \sum_{i=1}^{K} \exp(-E_K(x_{s|t}^{(i)}, s)), \quad x_{s|t}^{(i)} \sim \mathcal{N}(x; x_t, (\sigma_t^2 - \sigma_s^2)I) \tag{43}$$

$$= -\log \frac{1}{K^2} \sum_{i=1}^{K} \sum_{j=1}^{K} \exp(-\mathcal{E}(x_{0|t}^{(ij)})), \quad x_{0|t}^{(ij)} \sim \mathcal{N}(x; x_t, \sigma_t^2 I) \tag{44}$$

$$E_K^{B(1)}(x_t, t, s; \phi) := -\log \frac{1}{K} \sum_{i=1}^{K} \exp(-E_\phi(x_{s|t}^{(i)}, s)), \quad x_{s|t}^{(i)} \sim \mathcal{N}(x; x_t, (\sigma_t^2 - \sigma_s^2)I) \tag{45}$$

The mean and variance of a Sequential estimator can be derived by considering it as the MC estimator with K^2 samples:

$$\mathbb{E}[E_K^{\text{Seq}}(x_t, t)] = \mathcal{E}_t(x_t) + \frac{v_{0t}(x_t)}{2m_{0t}^2(x_t)K^2} \quad \text{and} \quad \text{Var}(E_K^{\text{Seq}}(x_t, t)) = \frac{v_{0t}(x_t)}{m_{0t}^2(x_t)K^2} \tag{46}$$

While an optimal network obtained by targeting the original MC energy estimator 4 at s is [1] :

$$E_{\phi^*}(x_s, s) = \mathbb{E}[E_K(x_s, s)] = -\log m_s(x_s) + \frac{v_{0s}(x_s)}{2m_s^2(x_s)K} \tag{47}$$

Then the optimal Bootstrap(1) estimator can be expressed as:

$$E_K^{B(1)}(x_t, t, s; \phi^*) = -\log \frac{1}{K} \sum_{i=1}^{K} \exp\left(-\left(-\log m_s(x_{s|t}^{(i)}) + \frac{v_{0s}(x_{s|t}^{(i)})}{2m_s^2(x_{s|t}^{(i)})K}\right)\right) \tag{48}$$

Before linking the Bootstrap estimator and the Sequential one, we provide the following approximation which is useful. Let a, b two random variables and $\{a_i\}_{i=1}^K$, $\{b_i\}_{i=1}^K$ are corresponding samples. Assume that $\{b_i\}_{i=1}^K$ are close to 0 and concentrated at m_b, while $\{a_i\}_{i=1}^K$ are concentrated at m_a,

[1] We consider minimizing the L_2-norm, *i.e.* $\phi^* = \arg\min_\phi \mathbb{E}_{x_0,t}[\|E_\theta(x_t, t) - E_K(x_t, t)\|^2]$. Since the target, E_K, is noisy, the optimal outputs are given by the expectation, *i.e.* $E_\phi^* = \mathbb{E}[E_K]$.

then

$$\log \frac{1}{K}\sum_{i=1}^{K}\exp(-(a_i+b_i)) = \log \frac{1}{K}\left\{\sum_{i=1}^{K}\exp(-a_i)\left[\frac{\sum_{i=1}^{K}\exp(-(a_i+b_i))}{\sum_{i=1}^{K}\exp(-a_i)}\right]\right\} \tag{49}$$

$$= \log \frac{1}{K}\sum_{i=1}^{K}\exp(-a_i) + \log \frac{\sum_{i=1}^{K}\exp(-(a_i+b_i))}{\sum_{i=1}^{K}\exp(-a_i)} \tag{50}$$

$$\approx \log \frac{1}{K}\sum_{i=1}^{K}\exp(-a_i) + \log \frac{\sum_{i=1}^{K}\exp(-a_i)(1-b_i)}{\sum_{i=1}^{K}\exp(-a_i)} \tag{51}$$

$$= \log \frac{1}{K}\sum_{i=1}^{K}\exp(-a_i) + \log\left(1 - \frac{\sum_{i=1}^{K}\exp(-a_i)b_i}{\sum_{i=1}^{K}\exp(-a_i)}\right) \tag{52}$$

$$\approx \log \frac{1}{K}\sum_{i=1}^{K}\exp(-a_i) - \frac{\sum_{i=1}^{K}\exp(-a_i)b_i}{\sum_{i=1}^{K}\exp(-a_i)} \tag{53}$$

$$\approx \log \frac{1}{K}\sum_{i=1}^{K}\exp(-a_i) - m_b \tag{54}$$

where Approximation applies a first order Taylor expansion of $e^x \approx 1 + x$ around $x = 0$ since $\{b_i\}_{i=1}^K$ are close to 0; while Approximation uses $\log(1+x) \approx x$ under the same assumption. Notice that when K is large and $\sigma_t^2 - \sigma_s^2 \le \kappa$ is small , $\{\frac{v_{0s}(x_{s|t}^{(i)})}{2m_s^2(x_{s|t}^{(i)})K}\}_{i=1}^K$ are close to 0 and concentrated at $\frac{v_{0s}(x_t)}{2m_s^2(x_t)K}$. Therefore, by plugging them into Equation 54, Equation 48 can be approximated by

$$E_K^{B(1)}(x_t, t, s; \phi^*) \approx -\log \frac{1}{K}\sum_{i=1}^{K} m_s(x_{s|t}^{(i)}) + \frac{v_{0s}(x_t)}{2m_s^2(x_t)K} \tag{55}$$

When K is large and σ_s^2 is small, the bias and variance of $E_K(x_{s|t}^{(i)}, s)$ are small, then we have

$$-\log \frac{1}{K}\sum_{i=1}^{K} m_s(x_{s|t}^{(i)}) \approx -\log \frac{1}{K}\sum_{i=1}^{K} E_K(x_{s|t}^{(i)}, s) = E_K^{\text{Seq}}(x_t, t) \tag{56}$$

Therefore, the optimal Bootstrap estimator can be approximated as follows:

$$E_K^{B(1)}(x_t, t, s; \phi^*) \approx E_K^{\text{Seq}}(x_t, t) + \frac{v_{0s}(x_t)}{2m_s^2(x_t)K} \tag{57}$$

where its mean and variance depend on those of the Sequential estimator (46):

$$\mathbb{E}[E_K^{B(1)}(x_t, t, s; \phi^*)] = \mathcal{E}_t(x_t) + \frac{v_{0t}(x_t)}{2m_t^2(x_t)K^2} + \frac{v_{0s}(x_t)}{2m_s^2(x_t)K} \tag{58}$$

$$\text{Var}[E_K^{B(1)}(x_t, t, s; \phi^*)] = \frac{v_{0t}(x_t)}{m_t^2(x_t)K^2} \tag{59}$$

D.2 Bootstrap(n) estimator

Given a bootstrap trajectory $\{s_i\}_{i=1}^n$ where $s_0 = 0$ and $s_n = s$, and E_ϕ is well learned at $[0, s]$. Let the energy network be optimal for $u \le s_n$ by learning a sequence of Bootstrap(i) energy estimators ($i \le n$). Then the optimal value of $E_\theta(x_s, s)$ is given by $\mathbb{E}[E_K^{B(n-1)}(x_s, s)]$. We are going to show the variance of a Bootstrap(n) estimator by induction. Suppose we have:

$$E_{\phi^*}(x_s, s) = \mathcal{E}_s(x_s) + \sum_{j=1}^{n} \frac{v_{0s_j}(x_s)}{2m_{s_j}^2(x_s)K^j} \tag{60}$$

Then for any $t \in (s, 1]$, the learning target of $E_\theta(x_t, t)$ is bootstrapped from $s_n = s$,

$$E_K^{B(n)}(x_t, t) = -\log \frac{1}{K} \sum_{i=1}^{K} \exp(-E_{\phi^*}(x_{s|t}^{(i)}, s)), \quad x_{s|t}^{(i)} \sim \mathcal{N}(x; x_t, (\sigma_t^2 - \sigma_s^2)I) \tag{61}$$

$$= -\log \frac{1}{K} \sum_{i=1}^{K} \exp\left(-\mathcal{E}_s(x_{s|t}^{(i)}) - \sum_{j=1}^{n} \frac{v_{0s_j}(x_{s|t}^{(i)})}{2m_{s_j}^2(x_{s|t}^{(i)})K^j} \right) \tag{62}$$

Assume that $\sigma_t^2 - \sigma_s^2$ is small and K is large, then we can apply Approximation 54 and have

$$E_K^{B(n)}(x_t, t) \approx -\log \frac{1}{K} \sum_{i=1}^{K} \exp(-\mathcal{E}_s(x_{s|t}^{(i)})) + \sum_{j=1}^{n} \frac{v_{0s_j}(x_t)}{2m_{s_j}^2(x_t)K^j} \tag{63}$$

In Bootstrap(n) setting, σ_s^2 is not small and we can't approximate $\mathcal{E}_s(x_{s|t}^{(i)})$ simply by a MC estimator E_K. However, we can sequentially estimate such energy by bootstrapping through the trajectory $\{s_i\}_{i=1}^n$, resembling a Sequential(n) estimator which is equivalent to $E_{K^{n+1}}$,

$$E_K^{B(n)}(x_t, t) \approx E_{K^{n+1}}(x_t, t) + \sum_{j=1}^{n} \frac{v_{0s_j}(x_t)}{2m_{s_j}^2(x_t)K^j} \tag{64}$$

therefore, the optimal output of the energy network at (x_t, t) by learning this Bootstrap(n) estimator is

$$\mathbb{E}[E_K^{B(n)}(x_t, t)] \approx \mathcal{E}_t(x_t) + \frac{v_{0t}(x_t)}{2m_t^2(x_t)K^{n+1}} + \sum_{j=1}^{n} \frac{v_{0s_j}(x_t)}{2m_{s_j}^2(x_t)K^j} \tag{65}$$

which suggests that the accumulated bias of a Bootstrap(n) estimator is given by

$$\frac{v_{0t}(x_t)}{2m_t^2(x_t)K^{n+1}} + \sum_{j=1}^{n} \frac{v_{0s_j}(x_t)}{2m_{s_j}^2(x_t)K^j} \tag{66}$$

□

E Incorporating Symmetry Using NEM

We consider applying NEM and BNEM in physical systems with symmetry constraints like n-body system. We prove that our MC energy estimator E_K is G-invariant under certain conditions, given in the following Proposition.

Proposition 4 *Let G be the product group SE(3) $\times \mathbb{S}_n \hookrightarrow O(3n)$ and p_0 be a G-invariant density in $\mathbb{R}^d$. Then the Monte Carlo energy estimator of $E_K(x_t, t)$ is G-invariant if the sampling distribution $x_{0|t} \sim \mathcal{N}(x_{0|t}; x_t, \sigma_t^2)$ is G-invariant, i.e.,*

$$\mathcal{N}(x_{0|t}; g \circ x_t, \sigma_t^2) = \mathcal{N}(g^{-1}x_{0|t}; x_t, \sigma_t^2).$$

Proof. Since p_0 is G-invariant, then $\mathcal{E}$ is G-invariant as well. Let $g \in G$ acts on $x \in \mathbb{R}^d$ where $g \circ x = gx$. Since $x_{0|t}^{(i)} \sim \overline{\mathcal{N}}(x_{0|t}; x_t, \sigma_t^2)$ is equivalent to $g \circ x_{0|t}^{(i)} \sim \overline{\mathcal{N}}(x_{0|t}; g \circ x_t, \sigma_t^2)$. Then we have

$$E_K(g \circ x_t, t) = -\log \frac{1}{K} \sum_{i=1}^{K} \exp(-\mathcal{E}(g \circ x_{0|t}^{(i)})) \tag{67}$$

$$= -\log \frac{1}{K} \sum_{i=1}^{K} \exp(-\mathcal{E}(x_{0|t}^{(i)})) = E_K(x_t, t) \tag{68}$$

$$x_{(0|t)}^{(i)} \sim \overline{\mathcal{N}}(x_{0|t}; x_t, \sigma_t^2) \tag{69}$$

Therefore, E_K is invariant to $G = \text{SE}(3) \times \mathbb{S}_n$. □

Furthermore, $E_K(x_t, t, s; \phi)$ is obtained by applying a learned energy network, which is G-invariant, to the analogous process and therefore is G-invariant as well.

F NEM for General SDEs

Diffusion models can be generalized to any SDEs as $dx_t = f(x_t, t)dt + g(t)dw_t$, where $t \geq 0$ and w_t is a Brownian motion. Particularly, we consider $f(x, t) := -\alpha(t)x$, *i.e.*

$$dx_t = -\alpha(t)x_t dt + g(t)dw_t \tag{70}$$

Then the marginal of the above SDE can be analytically derived as:

$$x_t = \beta(t)x_0 + \beta(t)\sqrt{\int_0^t (g(s)\beta(s))^2 \mathrm{d}s}\epsilon, \quad \beta(t) := e^{-\int_0^t \alpha(s)\mathrm{d}s} \tag{71}$$

where $\epsilon \sim \mathcal{N}(0, I)$. For example, when $g(t) = \sqrt{\bar{\beta}(t)}$ and $\alpha(t) = \frac{1}{2}\bar{\beta}(t)$, where $\bar{\beta}(t)$ is a monotonic function (*e.g.* linear) increasing from $\bar{\beta}_{\min}$ to $\bar{\beta}_{\max}$, the above SDE resembles a Variance Preserving (VP) process (Song et al., 2020). In DMs, VP can be a favor since it constrains the magnitude of noisy data across t; while a VE process doesn't, and the magnitude of data can explode as the noise explodes. Therefore, we aim to discover whether any SDEs rather than VE can be better by generalizing NEM and DEM to general SDEs.

In this work, we provide a solution for general SDEs (70) rather than a VE SDE. For simplification, we exchangeably use $\beta(t)$ and β_t. Given a SDE as Equation 70 for any integrable functions α and g, we can first derive its marginal as Equation 71, which can be expressed as:

$$\beta_t^{-1}x_t = x_0 + \sqrt{\int_0^t (g(s)\beta(s))^2 \mathrm{d}s}\epsilon \tag{72}$$

Therefore, by defining $y_t = \beta_t^{-1}x_t$ we have $y_0 = x_0$ and therefore:

$$y_t = y_0 + \sqrt{\int_0^t (g(s)\beta(s))^2 \mathrm{d}s}\epsilon \tag{73}$$

which resembles a VE SDE with noise schedule $\tilde{\sigma}^2(t) = \int_0^t (g(s)\beta(s))^2 ds$. In fact, we can also derive this by changing variables:

$$\begin{aligned} dy_t &= (\beta^{-1}(t))' x_t dt + \beta^{-1}(t)\mathrm{d}x_t && (74)\\ &= \beta^{-1}(t)\alpha(t)x_t dt + \beta^{-1}(t)(-\alpha(t)x_t dt + g(t)\mathrm{d}w_t) && (75)\\ &= \beta^{-1}(t)g(t)\mathrm{d}w_t && (76) \end{aligned}$$

which also leads to Equation 73. Let $\tilde{p}_t$ be the marginal distribution of y_t and p_t the marginal distribution of x_t, with $y_{0|t}^{(i)} \sim \mathcal{N}(y; y_t, \tilde{\sigma}_t^2 I)$ we have

$$\tilde{p}_t(y_t) \propto \int \exp(-\mathcal{E}(y))\mathcal{N}(y_t; y, \tilde{\sigma}_t^2 I)\mathrm{d}y \tag{77}$$

$$\tilde{S}_t(y_t) = \nabla_{y_t} \log \tilde{p}_t(y_t) \approx \nabla_{y_t} \log \sum_{i=1}^K \exp(-\mathcal{E}(y_{0|t}^{(i)})) \tag{78}$$

$$\tilde{\mathcal{E}}_t(y_t) \approx -\log \frac{1}{K}\sum_{i=1}^K \exp(-\mathcal{E}(y_{0|t}^{(i)})) \tag{79}$$

Therefore, we can learn scores and energies of y_t simply by following DEM and NEM for VE SDEs. Then for sampling, we can simulate the reverse SDE of y_t and eventually, we have $x_0 = y_0$.

Instead, we can also learn energies and scores of x_t. By changing the variable, we can have

$$p_t(x_t) = \beta_t^{-1}\tilde{p}_t(\beta_t^{-1}x_t) = \beta_t^{-1}\tilde{p}_t(y_t) \tag{80}$$

$$S_t(x_t) = \beta_t^{-1}\tilde{S}_t(\beta_t^{-1}x_t) = \beta_t^{-1}\tilde{S}_t(y_t) \tag{81}$$

which provides us the energy and score estimator for x_t:

$$\mathcal{E}_t(x_t) \approx -\log \beta_t^{-1} \frac{1}{K} \sum_{i=1}^{K} \exp(-\mathcal{E}(x_{0|t}^{(i)})) \tag{82}$$

$$S_t(x_t) \approx \beta_t^{-1} \nabla_{x_t} \log \sum_{i=1}^{K} \exp(-\mathcal{E}(x_{0|t}^{(i)})) \tag{83}$$

$$x_{0|t}^{(i)} \sim \mathcal{N}(x; \beta_t^{-1} x_t, \tilde{\sigma}^2(t) I) \tag{84}$$

Typically, α is a non-negative function, resulting in $\beta(t)$ decreasing from 1 and can be close to 0 when t is large. Therefore, the above equations realize that even though both the energies and scores for a general SDE can be estimated, the estimators are not reliable at large t since β_t^{-1} can be extremely large; while the SDE of y_t (76) indicates that this equivalent VE SDE is scaled by β_t^{-1}, resulting that the variance of y_t at large t can be extremely large and requires much more MC samples for a reliable estimator. This issue can be a bottleneck of generalizing DEM, NEM, and BNEM to other SDE settings, therefore developing more reliable estimators for both scores and energies is of interest in future work.

G Linking iDEM and iEFM Through Tweedie's Formula

In this supplementary work, we propose TWEEDIE DEM (TweeDEM), by leveraging the Tweedie's formula (Efron, 2011) into DEM, *i.e.* $\nabla_x \log p_t(x) = \mathbb{E}_{p(x_0|x_t)}\left[\frac{x_0 - x_t}{\sigma_t^2}\right]$. Surprisingly, TweeDEM can be equivalent to the iEFM-VE proposed by Woo and Ahn (2024), which is a variant of iDEM corresponding to another family of generative model, flow matching.

We first derive an MC estimator denoiser, *i.e.* the expected clean data given a noised data x_t at t

$$\mathbb{E}[x_0|x_t] = \int x_0 p(x_0|x_t) dx_0 \tag{85}$$

$$= \int x_0 \frac{q_t(x_t|x_0) p_0(x_0)}{p_t(x_t)} dx_0 \tag{86}$$

$$= \int x_0 \frac{\mathcal{N}(x_t; x_0, \sigma_t^2 I) \exp(-\mathcal{E}(x_0))}{\exp(-\mathcal{E}_t(x_t))} dx_0 \tag{87}$$

where the numerator can be estimated by an MC estimator $\mathbb{E}_{\mathcal{N}(x_t, \sigma_t^2 I)}[x \exp(-\mathcal{E}(x))]$ and the denominator can be estimated by another similar MC estimator $\mathbb{E}_{\mathcal{N}(x_t, \sigma_t^2 I)}[\exp(-\mathcal{E}(x))]$, suggesting we can approximate this denoiser through self-normalized importance sampling as follows

$$D_K(x_t, t) := \sum_{i=1}^{K} \frac{\exp(-\mathcal{E}(x_{0|t}^{(i)}))}{\sum_{j=1}^{K} \exp(-\mathcal{E}(x_{0|t}^{(j)}))} x_{0|t}^{(i)} \tag{88}$$

$$= \sum_{i=1}^{K} w_i x_{0|t}^{(i)} \tag{89}$$

where $x_{0|t}^{(i)} \sim \mathcal{N}(x_t, \sigma_t^2 I)$, w_i are the importance weights and $D_K(x_t, t) \approx \mathbb{E}[x_0|x_t]$. Then a new MC score estimator can be constructed by plugging the denoiser estimator D_K into Tweedie's formula

$$\tilde{S}_K(x_t, t) := \sum_{i=1}^{K} w_i \frac{x_{0|t}^{(i)} - x_t}{\sigma_t^2} \tag{90}$$

where $\frac{x_{0|t}^{(i)} - x_t}{\sigma_t^2}$ resembles the vector fields $v_t(x_t)$ in Flow Matching. In another perspective, these vector fields can be seen as scores of Gaussian, *i.e.* $\nabla \log \mathcal{N}(x; x_t, \sigma_t^2 I)$, and therefore $\tilde{S}_K$ is an importance-weighted sum of Gaussian scores while S_K can be expressed as an importance-weighted sum of system scores $-\nabla \mathcal{E}$. In addition, Karras et al. (2022) demonstrates that in

Denoising Diffusion Models, the optimal scores are an importance-weighted sum of Gaussian scores, while these importance weights are given by the corresponding Gaussian density, *i.e.* $S_{\text{DM}}(x_t, t) = \sum_i \tilde{w}_i(x_{0|t}^{(i)} - x_t)/\sigma_t^2$ and $\tilde{w}_i \propto \mathcal{N}(x_{0|t}^{(i)}; x_t, \sigma_t^2 I)$. We summarize these three different score estimators as follows:

• DEM, sum of system scores, weighted by system energies:

$$S_K(x_t, t) = \sum_i w_i(-\nabla \mathcal{E}(x_{0|t}^{(i)}))$$

• DDM, sum of perturbation kernel scores, weighted by perturbation densities:

$$S_{\text{DM}}(x_t, t) = \sum_i \tilde{w}_i \nabla \log \mathcal{N}(x_{0|t}^{(i)}; x_t, \sigma_t^2 I)$$

• EFM-VE=DEM+Tweedie, sum of perturbation kernel scores, weighted by system energies:

$$\tilde{S}_K(x_t, t) = \sum_i w_i \nabla \log \mathcal{N}(x_{0|t}^{(i)}; x_t, \sigma_t^2 I)$$

H Additional Experimental Details

H.1 Energy functions

GMM. A Gaussian Mixture density in 2-dimensional space with 40 modes, which is proposed by Midgley et al. (2023). Each mode in this density is evenly weighted, with identical covariances,

$$\Sigma = \begin{pmatrix} 40 & 0 \\ 0 & 40 \end{pmatrix} \tag{91}$$

and the means $\{\mu_i\}_{i=1}^{40}$ are uniformly sampled from $[-40, 40]^2$, i.e.

$$p_{\text{gmm}}(x) = \frac{1}{40} \sum_{i=1}^{40} \mathcal{N}(x; \mu_i, \Sigma) \tag{92}$$

Then its energy is defined by the negative-log-likelihood, i.e.

$$\mathcal{E}^{\text{GMM}}(x) = -\log p_{gmm}(x) \tag{93}$$

For evaluation, we sample 1000 data from this GMM with $TORCH.RANDOM.SEED(0)$ following Midgley et al. (2023); Akhound-Sadegh et al. (2024) as a test set.

DW-4. First introduced by Köhler et al. (2020), the DW-4 dataset describes a system with 4 particles in 2-dimensional space, resulting in a task with dimensionality $d = 8$. The energy of the system is given by the double-well potential based on pairwise Euclidean distances of the particles,

$$\mathcal{E}^{\text{DW}}(x) = \frac{1}{2\tau} \sum_{ij} a(d_{ij} - d_0) + b(d_{ij} - d_0)^2 + c(d_{ij} - d_0)^4 \tag{94}$$

where a, b, c and d_0 are chosen design parameters of the system, τ the dimensionless temperature and $d_{ij} = \|x_i - x_j\|_2$ are Euclidean distance between two particles. Following Akhound-Sadegh et al. (2024), we set $a = 0$, $b = -4$, $c = 0.9$ $d_0 = 4$ and $\tau = 1$, and we use validation and test set from the MCMC samples in Klein et al. (2023a) as the "Ground truth" samples for evaluating.

LJ-n. This dataset describes a system consisting of n particles in 3-dimensional space, resulting in a task with dimensionality $d = 3n$. Following Akhound-Sadegh et al. (2024), the energy of the system is given by $\mathcal{E}^{\text{Tot}}(x) = \mathcal{E}^{\text{LJ}}(x) + c\mathcal{E}^{\text{osc}}(x)$ with the Lennard-Jones potential

$$\mathcal{E}^{\text{LJ}}(x) = \frac{\epsilon}{2\tau} \sum_{ij} \left(\left(\frac{r_m}{d_{ij}} \right)^6 - \left(\frac{r_m}{d_{ij}} \right)^{12} \right) \tag{95}$$

and the harmonic potential

$$\mathcal{E}^{\text{osc}}(x) = \frac{1}{2} \sum_i \|x_i - x_{COM}\|^2 \tag{96}$$

where $d_{ij} = \|x_i - x_j\|_2$ are Euclidean distance between two particles, r_m, τ and ϵ are physical constants, x_{COM} refers to the center of mass of the system and c the oscillator scale. We use $r_m = 1$, $\tau = 1$, $\epsilon = 1$ and $c = 0.5$ the same as Akhound-Sadegh et al. (2024). We test our models in LJ-13 and LJ-55, which correspond to $d = 65$ and $d = 165$ respectively. And we use the MCMC samples given by Klein et al. (2023a) as a test set.

H.2 Evaluation Metrics

2-Wasserstein distance $\mathcal{W}_2$. Given empirical samples μ from the sampler and ground truth samples ν, the 2-Wasserstein distance is defined as:

$$\mathcal{W}_2(\mu, \nu) = (\inf_\pi \int \pi(x, y) d^2(x, y) dx dy)^{\frac{1}{2}} \tag{97}$$

where π is the transport plan with marginals constrained to μ and ν respectively. Following Akhound-Sadegh et al. (2024), we use the Hungarian algorithm as implemented in the Python optimal transport package (POT) (Flamary et al., 2021) to solve this optimization for discrete samples with the Euclidean distance $d(x, y) = \|x - y\|_2$. $x - \mathcal{W}_2$ is based on the data and $\mathcal{E} - \mathcal{W}_2$ is based on the corresponding energy.

Total Variation (TV). The total variation measures the dissimilarity between two probability distributions. It quantifies the maximum difference between the probabilities assigned to the same event by two distributions, thereby providing a sense of how distinguishable the distributions are. Given two distribution P and Q, with densities p and q, over the same sample space Ω, the TV distance is defined as

$$TV(P, Q) = \frac{1}{2} \int_\Omega |p(x) - q(x)| dx \tag{98}$$

Following Akhound-Sadegh et al. (2024), for low-dimentional datasets like GMM, we use 200 bins in each dimension. For larger equivariant datasets, the total variation distance is computed over the distribution of the interatomic distances of the particles.

H.3 Experiment Settings

We set the reverse SDE integration steps for iDEM, NEM, and BNEM in our main experiment as 100. In experiments for TweeDEM, we set this integration step 1000 for TweeDEM and iDEM.

GMM-40. For the basic model f_θ, we use an MLP with sinusoidal and positional embeddings which has 3 layers of size 128 as well as positional embeddings of size 128. The replay buffer is set to a maximum length of 10000.

During training, the generated data was in the range $[-1, 1]$ so to calculate the energy it was scaled appropriately by unnormalizing by a factor of 50. All models are trained with a geometric noise schedule with $\sigma_{\min} = 1e-5$, $\sigma_{\max} = 1$, $K = 500$ samples for computing both the MC score estimator S_K and MC energy estimator E_K, $K = 400$ samples for computing the Bootstrap energy estimator E_K^B and Bootstrap score estimator $S_K^B = \nabla E_K^B$ and we clipped the norm of S_K and S_K^B to 70. All models are trained with a learning rate of $5e-4$.

DW-4. All models use an EGNN with 3 message-passing layers and a 2-hidden layer MLP of size 128. All models are trained with a geometric noise schedule with $\sigma_{\min} = 1e-5$, $\sigma_{\max} = 3$, a learning rate of $1e-3$, $K = 1000$ samples for computing S_K and E_K, $K = 400$ samples for computing S_K^B and E_K^B, and we clipped S_K and S_K^B to a max norm of 20.

LJ-13. All models use an EGNN with 5 hidden layers and hidden layer size 128. All models are trained with a geometric noise schedule with $\sigma_{\min} = 0.01$ and $\sigma_{\max} = 2$, a learning rate of $1e-3$, $K = 1000$ samples for S_K and E_K, $K = 100$ samples for E_K^B and S_K^B, and we clipped S_K and S_K^B to a max norm of 20.

For all datasets. We use clipped scores as targets for iDEM and TweeDEM training for all tasks. Meanwhile, we also clip scores during sampling in outer-loop of training, when calculating the reverse SDE integral. These settings are shown to be crucial especially when the energy landscape is non-smooth and exists extremely large energies or scores, like LJ-13 and LJ-55. In fact, targeting the clipped scores refers to learning scores of smoothed energies. While we're learning unadjusted energy

for NEM and BNEM, the training can be unstable, and therefore we blue smooth the Lennard-Jones potential through the cubic spline interpolation, according to Moore et al. (2024). Besides, we predict per-particle energies for LJ-n datasets. It shows that this setting can significantly stabilize training and boost performance.

I Supplementary Experients

I.1 Comparing the Robustness of Energy-Matching and Score-Matching

In this section, we discussed the robustness of the energy-matching model(NEM) with the score-matching model(iDEM) by analyzing the influence of the numbers of MC samples used for estimators and choice of noise schedule on the sampler's performance.

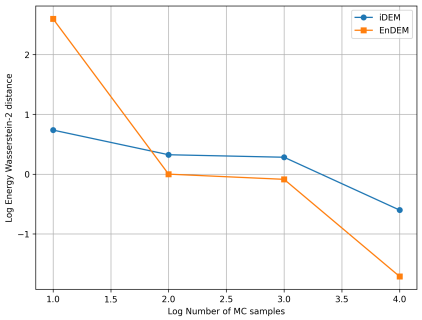
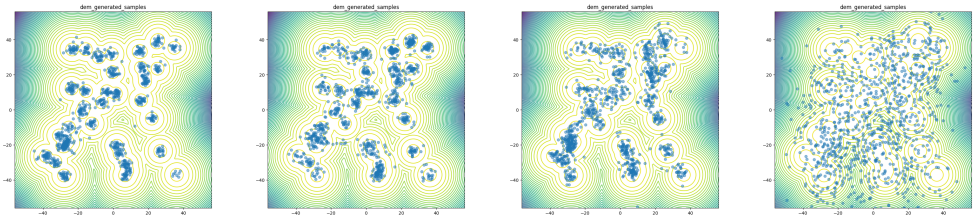


Figure 2: Comparison of the Energy Wasserstein-2 distance in DW4 benchmark between iDEM and NEM across varying numbers of MC samples.

As in Figure 2, NEM consistently outperforms iDEM when more than 100 MC samples are used for the estimator. Besides, NEM shows a faster decline when the number of MC samples increases. Therefore, we can conclude that the low variance of Energy-matching makes it more beneficial when we boost with more MC samples.

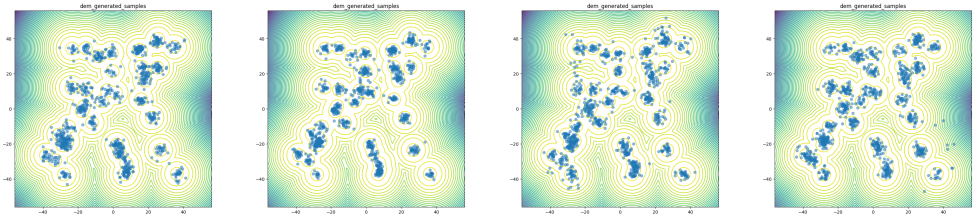


(a) Geometric: 4.16 (b) Cosine: 6.32 (c) Quadratic: 3.95 (d) Linear: 9.87

Figure 3: Comparison of iDEM sampler when employing different noise schedules. The performances of x-$\mathcal{W}_2$ are listed.

Then, we evaluate the performance differences when applying various noise schedules. The following four schedules were tested in the experiment:

- **Geometric noise schedule**: The noise level decreases geometrically in this schedule. The noise at step t is given by: $\sigma_t = \sigma_0^{1-t} \cdot \sigma_1^t$ where $\sigma_0 = 0.0001$ is the initial noise level, $\sigma_1 = 1$ is the maximum noise level, and t is the time step.
- **Cosine noise schedule**: The noise level follows a cosine function over time, represented by: $sigma_t = \sigma_1 \cdot \cos(\pi/2 \frac{1+\delta-t}{1+\delta})^2$, where $\delta = 0.008$ is a hyper-parameter that controls the decay rate.
- **Quadratic noise schedule**: The noise level follows a quadratic decay:$\sigma_t = \sigma_0 t^2$ where σ_0 is the initial noise level. This schedule applies a slow decay initially, followed by a more rapid reduction.



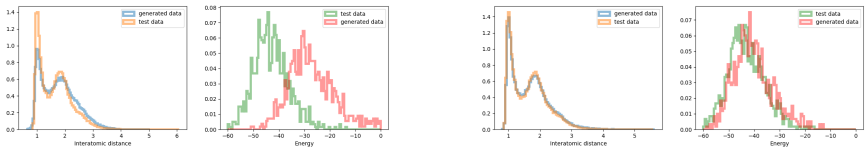
(a) Geometric: 2.64 (b) Cosine: 2.29 (c) Quadratic: 2.53 (d) Linear: 3.13

Figure 4: Comparison of NEM sampler when employing different noise schedules. The performances of x-$\mathcal{W}_2$ are listed.

- **Linear noise schedule**: In this case, the noise decreases linearly over time, represented as: $\sigma_t = \sigma_1 t$

The experimental results are depicted in Figure 3 and Figure 4. It is pretty obvious that for iDEM the performance varied for different noise schedules. iDEM favors noise schedules that decay more rapidly to 0 when t approaches 0. When applying the linear noise schedules, the samples are a lot more noisy than other schedules. This also proves our theoretical analysis that the variance would make the score network hard to train. On the contrary, all 4 schedules are able to perform well on NEM. This illustrates that the reduced variance makes NEM more robust and requires less hyperparameter tuning.

I.2 Scaling up to Lennard Jones Potential



(a) iDEM (b) NEM

Figure 5: Performance comparison of iDEM and NEM for LJ-13 potential

Besides GMM and DW-4, we would also like to show preliminary experimental results on the LJ-13 potentials. The distributions of interatomic distances and energies are shown in Figure 5.

The performance comparison between iDEM and NEM for the LJ-13 potential demonstrates the superior accuracy of NEM in capturing both the interatomic distance distribution and energy distribution. These results suggest that NEM's capabilities allow it to generalize better in complex many-body systems, like the Lennard-Jones potential, providing more reliable energy estimations and physical interpretations.

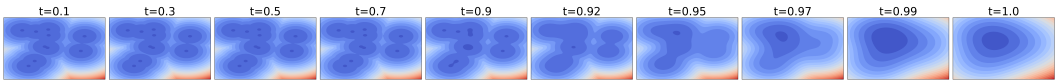
I.3 Empirical Analysis of the Variance of E_K and S_K

To justify the theoretical results for the variance of the MC energy estimator (4) and MC score estimator (2), we first empirically explore a 2D GMM. For better visualization, the GMM is set to be evenly weighted by 10 modes located in $[-1, 1]^2$ with identical variance $1/40$ for each component, resulting in the following density

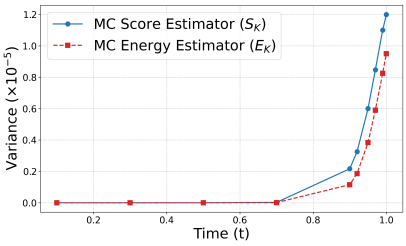
$$p'_{GMM}(x) = \frac{1}{10}\sum_{i=1}^{10}\mathcal{N}\left(x; \mu_i, \frac{1}{40}I\right) \tag{99}$$

while the marginal perturbed distribution at t can be analytically derived from Gaussian's property:

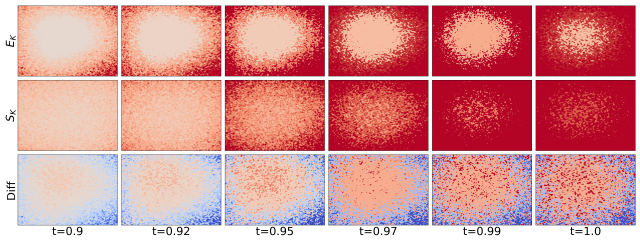
$$p_t(x) = (p'_{GMM} * \mathcal{N}(0, \sigma_t^2))(x_t) = \frac{1}{10}\sum_{i=1}^{10}\mathcal{N}\left(x; \mu_i, \left(\frac{1}{40} + \sigma_t^2\right)I\right) \tag{100}$$



(a) Ground truth energy across $t \in [0, 1]$

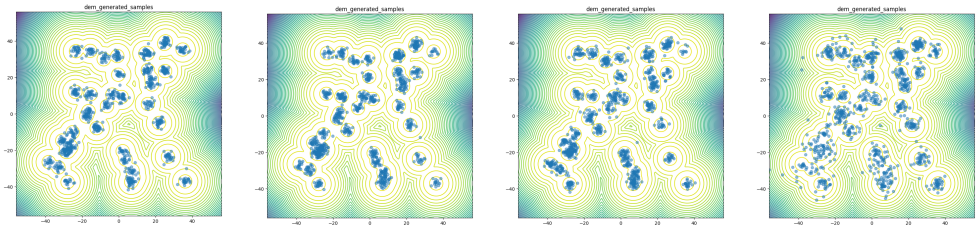


(b) Expected variance of estimators



(c) Point-wise variance for $t \in [0.9, 1]$

Figure 6: (a) the ground truth energy of the target GMM from $t = 0$ to $t = 1$; (b) the estimation of expected variance of x from $t = 0$ to $t = 1$, computed by a weighted sum over the variance of estimator at each location with weights equal to the marginal density p_t; (c) the variance of MC score estimator and MC energy estimator, and their difference (Var[score]-Var[energy]) for t from 0.9 to 1, we ignore the plots from $t \in [0, 0.9]$ since the variance of both estimators are small. The colormap ranges from blue (low) to red (high), where blues are negative and reds are positive.



(a) $\tilde{S}_K$ (ours) (b) S_K (c) TweeDEM (ours) (d) DEM

Figure 7: Sampled points from samplers applied to GMM-40 potentials, with the ground truth represented by contour lines. $\tilde{S}_K$ and S_K represent using these ground truth estimators for reverse SDE integration.

given a VE noising process.

We empirically estimate the variance for each pair of (x_t, t) by simulating 10 times the MC estimators. Besides, we estimate the expected variance over x for each time t, i.e. $\mathbb{E}_{p_t(x_t)}[\text{Var}(E_K(x_t, t))]$ and $\mathbb{E}_{p_t(x_t)}[\text{Var}(S_K(x_t, t))]$.

Figure 6a shows that, the variance of both MC energy estimator and MC score estimator increase as time increases. In contrast, the variance of E_K can be smaller than that of S_K in most areas, especially when the energies are low (see Figure 6c), aligning our Proposition 2. Figure 6b shows that in expectation over true data distribution, the variance of E_K is always smaller than that of S_K across $t \in [0, 1]$.

I.4 Experiments for TweeDEM

In Appendix G, we propose TweeDEM, a variant of DEM by leveraging Tweedie's formula (Efron, 2011), which theoretically links iDEM and iEFM-VE and suggests that we can simply replace the score estimator S_K (2) with $\tilde{S}_K$ (90) to reconstruct a iEFM-VE. We conduct experiments for this variant with the aforementioned GMM and DW-4 potential functions.

Setting. We follow the ones aforementioned, but setting the steps for reverse SDE integration 1000, the number of MC samples 500 for GMM and 1000 for DW-4. We set a quadratic noise schedule ranging from 0 to 3 for TweeDEM in DW-4.

Table 2: Sampler performance comparison for GMM-40 and DW-4 energy function. we measured the performance using data Wasserstein-2 distance (x-$\mathcal{W}_2$), Energy Wasserstein-2 distance ($\mathcal{E}$-$\mathcal{W}_2$) and Total Variation (TV). † We compare the optimal number reported by Woo and Ahn (2024) and Akhound-Sadegh et al. (2024). - indicates metric non-reported. **Bold** indicates the best values.

Energy →	**GMM-40** ($d = 2$)			**DW-4** ($d = 8$)		
Sampler ↓	**x-$\mathcal{W}_2$↓**	**$\mathcal{E}$-$\mathcal{W}_2$↓**	**TV↓**	**x-$\mathcal{W}_2$↓**	**$\mathcal{E}$-$\mathcal{W}_2$↓**	**TV↓**
S_K	2.864	**0.010**	**0.812**	1.841	**0.040**	0.092
$\tilde{S}_K$ (ours)	**2.506**	0.124	0.826	**1.835**	0.145	**0.087**
iDEM†	3.98	-	0.81	2.09	-	0.09
iDEM (rerun)	6.406	46.90	0.859	**1.862**	**0.030**	**0.093**
iEFM-VE†	4.31	-	-	2.21	-	-
iEFM-OT†	4.21	-	-	2.07	-	-
TweeDEM (ours)	**3.182**	**1.753**	**0.815**	1.910	0.217	0.120

To compare the two score estimators S_K and $\tilde{S}_K$ fundamentally, we first conduct experiments using these ground truth estimations for reverse SDE integration, *i.e.* samplers without learning. In addition, we consider using a neural network to approximate these estimators, *i.e.* iDEM and TweeDEM.

Table 2 reports x-$\mathcal{W}_2$, $\mathcal{E}-\mathcal{W}_2$, and TV for GMM and DW-4 potentials. We reports the numbers from Akhound-Sadegh et al. (2024) and Woo and Ahn (2024). We also rerun the iDEM and evaluate it for a more reliable comparisom. Table 2 shows that when using the ground truth estimators for sampling, there's no significant evidence demonstrating the privilege between S_K and $\tilde{S}_K$. However, when training a neural sampler, TweeDEM can significantly outperform iDEM (rerun), iEFM-VE, and iEFM-OT for GMM potential. While for DW4, TweeDEM outperforms iEFM-OT and iEFM-VE in terms of $x - \mathcal{W}_2$ but are not as good as our rerun iDEM.

Figure 7 visualizes the generated samples from ground truth samplers, *i.e.* S_K and $\tilde{S}_K$, and neural samplers, *i.e.* TweeDEM and iDEM. It shows that the ground truth samplers can generate well mode-concentrated samples, as well as TweeDEM, while samples generated by iDEM are not concentrated on the modes and therefore result in the high value of $\mathcal{W}_2$ based metrics. Also, this phenomenon aligns with the one reported by Woo and Ahn (2024), where the iEFM-OT and iEFM-VE can generate samples more concentrated on the modes than iDEM.

Above all, simply replacing the score estimator S_K with $\tilde{S}_K$ can improve generated data quality and outperform iEFM in GMM and DW-4 potentials. Though TweeDEM can outperform the previous state-of-the-art sampler iDEM on GMM, it is still not as capable as iDEM on DW-4. Except scaling up and conducting experiments on larger datasets like LJ-13, combing S_K and $\tilde{S}_K$ is of interest in the future, which balances the system scores and Gaussian ones and can possibly provide more useful and less noisy training signals. In addition, we are considering implementing a denoiser network for TweeDEM as our future work, which might stabilize the training process.