

## 789 Contents

790	<b>A Additional Preliminary and Reviews</b>	<b>21</b>
791	A.1 Stochastic Optimal Control (SOC) . . . . .	21
792	A.2 Schrödinger Bridge (SB) . . . . .	22
793	A.3 Additional Related Works . . . . .	23
794	<b>B Proofs</b>	<b>24</b>
795	B.1 Preliminary and Additional Theoretical Results . . . . .	24
796	B.2 Proof of Theorem 3.1 . . . . .	25
797	B.3 Proof of Theorem 4.1 . . . . .	25
798	B.4 Proof of Theorem 4.2 . . . . .	26
799	<b>C Practical Implementation of ASBS</b>	<b>27</b>
800	<b>D Experiment Details</b>	<b>30</b>
801	D.1 Synthetic Energy Functions . . . . .	30
802	D.2 Alanine dipeptide . . . . .	32
803	D.3 Amortized conformer generation . . . . .	32
804	D.4 Additional Experiments and Discussions . . . . .	34

## 805 A Additional Preliminary and Reviews

### 806 A.1 Stochastic Optimal Control (SOC)

807 In this subsection, we expand Section 2 with details. Recall the SOC problem in (4):

$$\min_u \mathbb{E}_{X \sim p^u} \left[ \int \frac{1}{2} \|u_t(X_t)\|^2 dt + g(X_1) \right] \quad (20a)$$

$$\text{s.t. } dX_t = [f_t(X_t) + \sigma_t u_t(X_t)] dt + \sigma_t dW_t, \quad X_0 \sim \mu. \quad (20b)$$

808 Similar to (8), the optimal control to (20) can be characterized through an optimality equation:

$$u_t^*(x) = -\sigma_t \nabla V_t(x), \quad \text{where} \quad V_t(x) = -\log \int p_{1|t}^{\text{base}}(y|x) e^{-V_1(y)} dy, \quad V_1(x) = g(x) \quad (21)$$

809 is the value function known to satisfy the Hamilton–Jacobi–Bellman (HJB) equation (Bellman,  
810 1954). This value function  $V_t(x)$  characterizes not only the optimal control  $u^*$  but also the optimal  
811 distribution  $p^*$ :

$$p^*(X) = p^{\text{base}}(X) e^{-V_1(X_1) + V_0(X_0)}, \quad (22)$$

812 where we shorthand  $X \equiv X_{[0,1]}$  and denote  $p^{\text{base}}$  the base distribution induced by (20b) with  $u := 0$ ,  
813 *i.e.*, the uncontrolled distribution. Note that (5) can be recovered by marginalizing over  $t \in (0, 1)$ .

814 **Adjoint Matching (AM)** Scalable computational methods for solving (20) have been challenging,  
815 as naively back-propagating through (20) induces prohibitively high computational cost. Instead,  
816 Adjoint Matching (Domingo-Enrich et al., 2025) employs a matching-based objective, named Adjoint  
817 Matching (AM):

$$u^* = \arg \min_u \mathbb{E}_{X \sim p^u} [\|u_t(X_t) + \sigma_t a_t\|^2], \quad \bar{u} = \text{stopgrad}(u), \quad (23a)$$

$$\text{where} \quad -da_t = a_t \cdot \nabla f_t(X_t) dt, \quad a_1 = \nabla g(X_1) \quad (23b)$$

is the backward dynamics of the (lean) adjoint state  $a_t \equiv a(t; X_{[t,1]})$ . It has been proven that the unique critical point of (23) is the optimal control  $u^*$ , implying a new characteristics of the optimal control  $u^*$  using the adjoint state:

$$u_t^*(x) = -\sigma_t \mathbb{E}_{p^*}[a_t | X_t = x]. \quad (24)$$

**Adjoint Sampling (AS)** Recently, Havens et al. (2025) introduced an adaptation of AM tailored to sampling Boltzmann distribution  $\nu(x) \propto e^{-E(x)}$  by considering

$$f_t := 0, \quad \mu(x) := \delta_0(x), \quad g(x) := \log \frac{p_1^{\text{base}}(x)}{\nu(x)}. \quad (25)$$

That is, AS considers the following SOC problem with a degenerate base drift, a Dirac delta prior, and a specific instantiation of the terminal cost  $g(x) := \log \frac{p_1^{\text{base}}(x)}{\nu(x)}$ :

$$\min_u \mathbb{E}_{X \sim p^u} \left[ \int_0^1 \frac{1}{2} \|u_t(X_t)\|^2 dt + \log \frac{p_1^{\text{base}}(X_1)}{\nu(X_1)} \right] \quad \text{s.t. } dX_t = \sigma_t u_t(X_t) dt + \sigma_t dW_t, \quad X_0 = 0. \quad (26)$$

Notably, this SOC problem (26) admits a simplified adjoint state  $a_t$  and a degenerate initial value function  $V_0(x)$ :

$$a_t \stackrel{(23b)}{=} \nabla g(X_1) \stackrel{(25)}{=} \nabla \log p_1^{\text{base}}(X_1) + \nabla E(X_1) \quad \forall t \in [0, 1] \quad (27)$$

$$V_0(x) \stackrel{(25)}{=} -\log \int p_1^{\text{base}}(y) \frac{\nu(y)}{p_1^{\text{base}}(y)} dy = -\log 1 = 0, \quad (28)$$

which further implies that the optimal distribution  $p^*$  is a reciprocal process (Léonard et al., 2014):

$$p^*(X) \stackrel{(28)}{=} p^{\text{base}}(X) e^{-V_1(X_1)} \stackrel{(25)}{=} p^{\text{base}}(X) \frac{\nu(X_1)}{p_1^{\text{base}}(X_1)} = p^{\text{base}}(X|X_1) p^*(X_1). \quad (29)$$

Combining the adjoint characteristics of the optimal control (24) with the simplified adjoint state  $a_t$  (27) and optimal distribution  $p^*$  (29) motivates the following *Reciprocal Adjoint Matching (RAM)* objective used in AS, where the unique critical point remains to be the optimal control  $u^*$  in (21).

$$u^* = \arg \min_u \mathbb{E}_{p_{t|1}^{\text{base}} p_1^{\bar{u}}} [\|u_t(X_t) + \sigma_t (\nabla E + \nabla \log p_1^{\text{base}})(X_1)\|^2], \quad \bar{u} = \text{stopgrad}(u). \quad (30)$$

**Remark** The reciprocal representation of the optimal-controlled distribution  $p^*$  in (29) extends to general SOC problems (20) with non-trivial base drifts and source distributions. Specifically, any optimal-controlled distribution that solves (20) can be factorized by

$$p^*(X) = p^{\text{base}}(X|X_0, X_1) p^*(X_0, X_1). \quad (31)$$

We leave a formal statement in Theorem B.3 and Corollary B.4.

## A.2 Schrödinger Bridge (SB)

In this subsection, we provide additional clarification on SB and specifically the derivation of (13). Recall the optimality equations of SB in (8):

$$u_t^*(x) = \sigma_t \nabla \log \varphi_t(x), \quad \text{where} \quad \begin{cases} \varphi_t(x) = \int p_{1|t}^{\text{base}}(y|x) \varphi_1(y) dy, & \varphi_0(x) \hat{\varphi}_0(x) = \mu(x) \\ \hat{\varphi}_t(x) = \int p_{t|0}^{\text{base}}(x|y) \hat{\varphi}_0(y) dy, & \varphi_1(x) \hat{\varphi}_1(x) = \nu(x) \end{cases} \quad (32a) \quad (32b)$$

Just like how the value function of an SOC problem fully characterizes the optimal control and its corresponding optimal distribution, so does the SB potential  $\varphi_t(x)$ :

$$p^*(X) = p^{\text{base}}(X) \frac{\varphi_1(X_1)}{\varphi_0(X_0)} = p^{\text{base}}(X|X_0) \varphi_1(X_1) \hat{\varphi}_0(X_0), \quad (33)$$

where the last equality is due to  $p^{\text{base}}(X) = p^{\text{base}}(X|X_0)\mu(X_0)$  and then invoking (32a). Note that (33) recovers (10) by marginalizing over  $t \in (0, 1)$ . Due to the construction of  $\varphi_t(x)$  and  $\hat{\varphi}_t(x)$  in (32), the marginal optimal distribution admits a strikingly simple factorization:

$$\begin{aligned} p_t^*(x) &= \int p^{\text{base}}(X, X_t = x|X_0) \varphi_1(X_1) \hat{\varphi}_0(X_0) dX \\ &= \int \int p^{\text{base}}(X_1|X_t = x) p^{\text{base}}(X_t = x|X_0) \varphi_1(X_1) \hat{\varphi}_0(X_0) dX_0 dX_1 \\ &= \left( \int p^{\text{base}}(X_t = x|X_0) \hat{\varphi}_0(X_0) dX_0 \right) \left( \int p^{\text{base}}(X_1|X_t = x) \varphi_1(X_1) dX_1 \right) \\ &= \hat{\varphi}_t(x) \varphi_t(x), \end{aligned} \quad (34)$$

or, more generally,

$$p_{s,t}^*(y, x) = p_{t|s}^{\text{base}}(x|y) \hat{\varphi}_s(y) \varphi_t(x), \quad s \leq t. \quad (35)$$

**Derivation of (13)** We now provide a simpler derivation of (13) compared to its original derivation based on path measure theory (Shi et al., 2023):

$$\begin{aligned} \nabla \log \hat{\varphi}_t(x) &\stackrel{(32b)}{=} \frac{1}{\hat{\varphi}_t(x)} \nabla_x \int p_{t|0}^{\text{base}}(x|y) \hat{\varphi}_0(y) dy \\ &= \frac{1}{\hat{\varphi}_t(x)} \int \nabla_x \log p_{t|0}^{\text{base}}(x|y) p_{t|0}^{\text{base}}(x|y) \hat{\varphi}_0(y) dy \\ &= \int \nabla_x \log p_{t|0}^{\text{base}}(x|y) p_{0|t}^*(y|x) dy, \end{aligned} \quad (36)$$

where the last equality follows by

$$p_{0|t}^*(y|x) \stackrel{(34)}{=} \frac{p_{0,t}^*(y, x)}{\hat{\varphi}_t(x) \varphi_t(x)} \stackrel{(35)}{=} \frac{p_{t|0}^{\text{base}}(x|y) \hat{\varphi}_0(y) \varphi_t(x)}{\hat{\varphi}_t(x) \varphi_t(x)} = \frac{p_{t|0}^{\text{base}}(x|y) \hat{\varphi}_0(y)}{\hat{\varphi}_t(x)}.$$

Equation (36) implies a matching-based variational formulation of  $\nabla \log \hat{\varphi}_t(\cdot)$ —also known as the *bridge matching* objective in data-driven SB (Shi et al., 2023; Liu et al., 2023).

$$\nabla \log \hat{\varphi}_t = \arg \min_h \mathbb{E}_{p_{0,t}^*} [\|h_t(X_t) - \nabla_{x_t} \log p^{\text{base}}(X_t|X_0)\|^2]. \quad (37)$$

Equation (37) recovers (13) at  $t = 1$ .

### A.3 Additional Related Works

In this subsection, we provide additional review on existing learning-based methods for sampling Boltzmann distributions.

**Learning-augmented MCMC** This class of methods can be thought of as extension of classical sampling methods—such as MCMC (Metropolis et al., 1953; HASTINGS, 1970), Sequential Monte Carlo (SMC; Del Moral et al., 2006) and Annealed Importance Sampling (AIS; Neal, 2001)—where traditional proposal distributions are replaced with modern machine learning models. For instance, Arbel et al. (2021) and Gabri   et al. (2022) use normalizing flows (Chen et al., 2018) as learned proposal distributions, whereas Matthews et al. (2022) employ stochastic normalizing flow (Wu et al., 2020). More recently, Chen et al. (2025) have explored the use of diffusion models (Song et al., 2021; Ho et al., 2020). However, training these models typically requires computing importance weights, which necessitates a large number of energy evaluations.

**MCMC-augmented Diffusion Samplers** Alternatively, methods of this class adopt modern generative models to sampling Boltzmann distributions and incorporate MCMC techniques to mitigate the lack of explicit target samples. For example, Phillips et al. (2024), (De Bortoli et al., 2024) and (Akhound-Sadegh et al., 2024) employ score matching objective from score-based diffusion models (Song et al., 2021; Ho et al., 2020). In contrast, Albergo and Vanden-Eijnden (2024) base their method on action matching objectives (Neklyudov et al., 2023). However, estimating target samples requires computing importance weights, which makes these methods computationally expensive in terms of energy function evaluations.

## 870 B Proofs

### 871 B.1 Preliminary and Additional Theoretical Results

872 **Lemma B.1** (Itô lemma (Itô, 1951)). *Let  $X_t$  be the solution to the Itô SDE:*

$$dX_t = f_t(X_t)dt + \sigma_t dW_t.$$

873 *Then, the stochastic process  $v_t(X_t)$ , where  $v \in C^{1,2}([0, 1], \mathbb{R}^d)$ , is also an Itô process:*

$$dv_t(X_t) = \left[ \partial_t v_t(X_t) + \nabla v_t(X_t) \cdot f + \frac{1}{2} \sigma_t^2 \Delta v_t(X_t) \right] dt + \sigma_t \nabla v_t(X_t) \cdot dW_t. \quad (38)$$

874 **Lemma B.2** (Laplacian trick). *For any twice-differentiable function  $\pi$  such that  $\pi(x) \neq 0$ , it holds*  
875 *that*

$$\frac{1}{\pi(x)} \Delta \pi(x) = \|\nabla \log \pi(x)\|^2 + \Delta \log \pi(x) \quad (39)$$

*Proof.*

$$\begin{aligned} \Delta \pi(x) &= \nabla \cdot \nabla \pi(x) \\ &= \nabla \cdot (\pi(x) \nabla \log \pi(x)) \\ &= \nabla \pi(x) \cdot \nabla \log \pi(x) + \pi(x) \Delta \log \pi(x) \\ &= \pi(x) (\|\nabla \log \pi(x)\|^2 + \Delta \log \pi(x)) \end{aligned}$$

876

□

877 **Theorem B.3** (SB characteristics of SOC). *The optimal distribution  $p^*$  of the SOC problem in (20)*  
878 *is also the solution to the following SB problem:*

$$\arg \min_p \{D_{\text{KL}}(p \| p^{\text{base}}) : p_0 = \mu, \quad p_1 = p_1^*\}. \quad (40)$$

879 *Proof.* We aim to show that there exist a transform such that the SOC's optimality equation (21) can  
880 be reinterpreted as the ones for SB (32). To this end, consider

$$\varphi_t(x) := e^{-V_t(x)}, \quad \hat{\varphi}_t(x) := e^{V_t(x)} p_t^*(x). \quad (41)$$

881 One can verify that the value function  $V_t(x)$  defined in (21) can be rewritten as

$$\varphi_t(x) = \int p_{1|t}^{\text{base}}(y|x) \varphi_1(y) dy.$$

882 On the other hand, we can expand  $\hat{\varphi}_t(x)$  by

$$\begin{aligned} \hat{\varphi}_t(x) &= e^{V_t(x)} \int p^*(X|X_t = x) dX \\ &= e^{V_t(x)} \int p^{\text{base}}(X_1|X_t = x) p^{\text{base}}(X_t = x, X_0) e^{-V_1(X_1) + V_0(X_0)} dX_1 dX_0 && \text{by (22)} \\ &= e^{V_t(x)} \int p^{\text{base}}(X_t = x, X_0) e^{-V_t(x) + V_0(X_0)} dX_0 && \text{by (21)} \\ &= \int p^{\text{base}}(X_t = x|X_0) \mu(X_0) e^{V_0(X_0)} dX_0 \\ &= \int p_{t|0}^{\text{base}}(x|y) \hat{\varphi}_0(y) dy. && \text{by (41)} \end{aligned}$$

883 Combined, the optimality equation (21) for the SOC problem can be rewritten equivalently as

$$u_t^*(x) = \sigma_t \nabla \log \varphi_t(x), \quad \text{where} \quad \begin{cases} \varphi_t(x) = \int p_{1|t}^{\text{base}}(y|x) \varphi_1(y) dy, & \varphi_0(x) \hat{\varphi}_0(x) = \mu(x), \\ \hat{\varphi}_t(x) = \int p_{t|0}^{\text{base}}(x|y) \hat{\varphi}_0(y) dy, & \varphi_1(x) \hat{\varphi}_1(x) = p_1^*(x). \end{cases}$$

884 We conclude that  $p^*$  indeed solves (40). □

885 **Corollary B.4** (Reciprocal process of the SOC problem). *The optimal distribution  $p^*$  of the SOC*  
886 *problem in (20) is a reciprocal process, i.e.,*

$$p^*(X) = p^{\text{base}}(X|X_0, X_1) p^*(X_0, X_1). \quad (43)$$

## 887 B.2 Proof of Theorem 3.1

888 Comparing (8a) to (21), we can reinterpret  $\varphi_t(x)$  as an value function  $V_t(x)$  by reinterpreting

$$V_t(x) := -\log \varphi_t(x), \quad g(x) := -\log \varphi_1(x) \stackrel{(8b)}{=} \log \frac{\hat{\varphi}_1(x)}{\nu(x)}.$$

889 That is, the kinetic-optimal drift of SB solves an SOC problem (4) with a terminal cost  $g(x) := \frac{\hat{\varphi}_1(x)}{\nu(x)}$ .  
890  $\square$

## 891 B.3 Proof of Theorem 4.1

892 For notational simplicity, we will denote  $q \equiv q^{\bar{h}^{(k-1)}}$  throughout the proof. We first rewrite the  
893 backward SDE (17) in the forward direction (Nelson, 2020):

$$dX_t = [f_t - \sigma_t^2 \nabla \log \phi_t + \sigma_t^2 \nabla \log q_t] dt + \sigma_t dW_t, \quad X_1 \sim \nu,$$

894 where we rewrite  $\phi_t(x)$  w.r.t. the forward time coordiante:

$$\phi_t(x) = \int p_{t|0}^{\text{base}}(x|y) \phi_0(y) dy, \quad \phi_1(x) = \bar{h}^{(k-1)}(x). \quad (44)$$

895 Note that (44) admits an equivalent PDE form by invoking Feynman-Kac formula (Le Gall, 2016):

$$\partial_t \phi_t(x) = -\nabla \cdot (f_t \phi_t) + \frac{\sigma_t^2}{2} \Delta \phi_t(x), \quad \phi_1(x) = \bar{h}^{(k-1)}(x). \quad (45)$$

896 On the other hand, the dynamics of  $\partial_t q$  follows the Fokker Plank equation (Øksendal, 2003):

$$\begin{aligned} \partial_t q_t &= -\nabla \cdot ((f_t - \sigma_t^2 \nabla \log \phi_t + \sigma_t^2 \nabla \log q_t) q_t) + \frac{1}{2} \sigma_t^2 \Delta q_t \\ &= \nabla \cdot ((\sigma_t^2 \nabla \log \phi_t - f_t) q_t) - \frac{1}{2} \sigma_t^2 \Delta q_t, \end{aligned}$$

897 and straightforward calculation yields

$$\partial_t \log q_t = \sigma_t^2 \Delta \log \phi_t - \nabla \cdot f_t + (\sigma_t^2 \nabla \log \phi_t - f_t) \cdot \nabla \log q_t - \frac{1}{2} \sigma_t^2 \|\nabla \log q_t\|^2 - \frac{1}{2} \sigma_t^2 \Delta \log q_t, \quad (46)$$

898 where we apply the Laplacian trick (39) to  $\frac{1}{q} \Delta q = \|\nabla \log q_t\|^2 + \Delta \log q_t$ .

899 Now, recall that  $p$  is the path distribution induced by the following SDE:

$$dX_t = [f_t(X_t) + \sigma_t u_t(X_t)] dt + \sigma_t dW_t, \quad X_0 \sim \mu. \quad (47)$$

900 Invoke Ito Lemma (38) to  $\log q_t(X_t)$ , where  $X_t$  follows (47):

$$\begin{aligned} d \log q_t &= \left[ \partial_t \log q_t + \nabla \log q_t \cdot (f_t + \sigma_t u_t) + \frac{1}{2} \sigma_t^2 \Delta \log q_t \right] dt + \sigma_t \nabla \log q_t \cdot dW_t \\ &\stackrel{(46)}{=} \left[ \sigma_t^2 \Delta \log \phi_t - \nabla \cdot f_t + \sigma_t^2 \nabla \log \phi_t \cdot \nabla \log q_t - \frac{1}{2} \sigma_t^2 \|\nabla \log q_t\|^2 + \nabla \log q_t \cdot (\sigma_t u_t) \right] dt \\ &\quad + \sigma_t \nabla \log q_t \cdot dW_t \end{aligned} \quad (48)$$

901 Likewise, invoke Ito Lemma (38) to  $\log \phi_t(X_t)$ , where  $X_t$  follows (47):

$$\begin{aligned} d \log \phi_t &= \left[ \partial_t \log \phi_t + \nabla \log \phi_t \cdot (f_t + \sigma_t u_t) + \frac{1}{2} \sigma_t^2 \Delta \log \phi_t \right] dt + \sigma_t \nabla \log \phi_t \cdot dW_t \\ &\stackrel{(45)}{=} \left[ -\nabla \cdot f_t + \frac{\sigma_t^2}{2} \frac{\Delta \phi_t}{\phi_t} + \nabla \log \phi_t \cdot (\sigma_t u_t) + \frac{1}{2} \sigma_t^2 \Delta \log \phi_t \right] dt + \sigma_t \nabla \log \phi_t \cdot dW_t \\ &\stackrel{(39)}{=} \left[ -\nabla \cdot f_t + \frac{\sigma_t^2}{2} (\|\nabla \log \phi_t\|^2 + \Delta \log \phi_t) + \nabla \log \phi_t \cdot (\sigma_t u_t) + \frac{1}{2} \sigma_t^2 \Delta \log \phi_t \right] dt + \sigma_t \nabla \log \phi_t \cdot dW_t \\ &= \left[ -\nabla \cdot f_t + \frac{\sigma_t^2}{2} \|\nabla \log \phi_t\|^2 + \nabla \log \phi_t \cdot (\sigma_t u_t) + \sigma_t^2 \Delta \log \phi_t \right] dt + \sigma_t \nabla \log \phi_t \cdot dW_t \end{aligned} \quad (49)$$

902 Subtracting (49) from (48) leads to

$$\mathrm{d} \log \phi_t - \mathrm{d} \log q_t = \left[ \frac{1}{2} \|u_t + \sigma_t \nabla \log \phi_t - \sigma_t \nabla \log q_t\|^2 - \frac{1}{2} \|u_t\|^2 \right] \mathrm{d}t + \sigma_t \nabla \log \frac{\phi_t}{q_t} \cdot \mathrm{d}W_t. \quad (50)$$

903 Finally, we are ready to compute the variational objective in (16):

$$\begin{aligned} D_{\mathrm{KL}}(p||q^{\bar{h}^{(k-1)}}) &= \mathbb{E}_{X \sim p^u} \left[ \int_0^1 \frac{1}{2} \|u_t(X_t) + \sigma_t \nabla \log \phi_t(X_t) - \sigma_t \nabla \log q_t(X_t)\|^2 \mathrm{d}t \right] \\ &\stackrel{(50)}{=} \mathbb{E}_{X \sim p^u} \left[ \int_0^1 \left( \frac{1}{2} \|u_t(X_t)\|^2 + \mathrm{d} \log \phi_t(X_t) - \mathrm{d} \log q_t(X_t) \right) \mathrm{d}t \right] \\ &= \mathbb{E}_{X \sim p^u} \left[ \int_0^1 \frac{1}{2} \|u_t(X_t)\|^2 \mathrm{d}t + \log \frac{\phi_1(X_1)}{q_1(X_1)} - \log \frac{\phi_0(X_0)}{q_0(X_0)} \right] \end{aligned} \quad (51)$$

$$\propto \mathbb{E}_{X \sim p^u} \left[ \int_0^1 \frac{1}{2} \|u_t(X_t)\|^2 \mathrm{d}t + \log \frac{\bar{h}^{(k-1)}(X_1)}{\nu(X_1)} \right]. \quad (52)$$

904 That is, we have shown that the variational objective  $D_{\mathrm{KL}}(p||q^{\bar{h}^{(k-1)}})$  is equivalent (up to an additive  
905 constant) to an SOC problem (52). Applying Reciprocal Adjoint Matching (Havens et al., 2025) with  
906 the reciprocal process from Corollary B.4 conclude that  $D_{\mathrm{KL}}(p||q^{\bar{h}^{(k-1)}})$  is minimized by  $p^{u^{(k)}}$ .  $\square$

#### 907 B.4 Proof of Theorem 4.2

908 For notational simplicity, we will denote  $p^{(k)} \equiv p^{u^{(k)}}$  throughout the proof. Let  $q$  be the path  
909 distribution induced by a backward SDE, propagating along the time coordinate  $s := 1 - t$ :

$$\mathrm{d}Y_s = [-f_s(Y_s) + \sigma_s v_s(Y_s)] \mathrm{d}s + \sigma_s \mathrm{d}W_s, \quad Y_0 \sim \nu.$$

910 Next, rewrite the forward SDE  $p^{(k)}$  in the backward direction:

$$\mathrm{d}Y_s = \left[ -f_s - \sigma_s u_s^{(k)} + \sigma_s^2 \nabla \log p_s^{(k)} \right] \mathrm{d}s + \sigma_s \mathrm{d}W_s, \quad Y_0 \sim p_t^{(k)}|_{t=1}.$$

911 By Theorem B.3, we know that  $p^{(k)}$  is the SB solution, thereby satisfying

$$u_t^{(k)}(x) = \sigma_t \nabla \log \varphi_t(x), \text{ where } \begin{cases} \varphi_t(x) = \int p_{1|t}^{\text{base}}(y|x) \varphi_1(y) \mathrm{d}y, & \varphi_0(x) \hat{\varphi}_0(x) = \mu(x) \\ \hat{\varphi}_t(x) = \int p_{t|0}^{\text{base}}(x|y) \hat{\varphi}_0(y) \mathrm{d}y, & \varphi_1(x) \hat{\varphi}_1(x) = p_1^{(k)}(x) \end{cases} \quad (53a)$$

912 Since we are working with the backward time coordinate  $s$ , it is convenience to define  $\phi_s := \hat{\varphi}_{1-t}$   
913 and rewrite (53b) by

$$\phi_s(y) = \int p_{1-s|0}^{\text{base}}(y|z) \phi_1(z) \mathrm{d}z, \quad \phi_0(y) = \frac{p_1^{(k)}(y)}{\varphi_1(y)}. \quad (54)$$

914 Now, expanding the variational objective with Girsanov Theorem yields (Särkkä and Solin, 2019)

$$D_{\mathrm{KL}}(p^{(k)}||q) = \mathbb{E}_{Y \sim p^{(k)}} \left[ \int_0^1 \frac{1}{2} \left\| -\sigma_s \nabla \log \varphi_s(Y_s) + \sigma_s \nabla \log p_s^{(k)}(Y_s) - v_s(Y_s) \right\|^2 \mathrm{d}s \right], \quad (55)$$

915 which is minimized point-wise at

$$v_s^*(y) = \sigma_s \nabla \log \frac{p_s^{(k)}(y)}{\varphi_s(y)} \stackrel{(34)}{=} \sigma_s \nabla \log \hat{\varphi}_s(y).$$

916 In other words, the backward SDE that minimizes (55) must obey

$$\mathrm{d}Y_s = [-f_s(Y_s) + \sigma_s^2 \nabla \log \phi_s(Y_s)] \mathrm{d}s + \sigma_s \mathrm{d}W_s, \quad Y_0 \sim \nu,$$

---

**Algorithm 2** Adjoint Schrödinger Bridge Sampler (ASBS)

---

**Require:** Sample-able source  $X_0 \sim \mu$ , differentiable energy  $E(x)$ , parametrized drift  $u_\theta(t, x)$  and corrector  $h_\phi(x)$ , replay buffers  $\mathcal{B}_{\text{adj}}$  and  $\mathcal{B}_{\text{crt}}$ , number of stages  $K$ , numbers of AM and CM epochs  $M_{\text{adj}}$  and  $M_{\text{crt}}$ , number of resamples  $N$ , number of gradient steps  $L$ , time scaling  $\lambda_t$ , maximum energy gradient norm  $\alpha_{\text{max}}$ .

```

1: Initialize  $h_\phi^{(0)} := 0$  ▷ IPF initialization
2: for stage  $k$  in  $1, 2, \dots, K$  do
3:   for epoch in  $1, 2, \dots, M_{\text{adj}}$  do ▷ adjoint matching
4:     Sample from model  $\{(X_0^{(i)}, X_1^{(i)})\}_{i=1}^N \sim p^{\bar{u}^{(k)}}$ , where  $\bar{u}^{(k)} = \text{stopgrad}(u_\theta^{(k)})$ 
5:     Compute adjoint target  $a_t^{(i)} := \text{stopgrad}(\text{clip}(\nabla E(X_1^{(i)}), \alpha_{\text{max}}) + h_\phi^{(k)}(X_1^{(i)}))$ 
6:     Update replay buffer  $\mathcal{B}_{\text{adj}} \leftarrow \mathcal{B}_{\text{adj}} \cup \{(X_0^{(i)}, X_1^{(i)}, a_t^{(i)})\}_{i=1}^N$ 
7:     Take  $L$  gradient steps  $\nabla_\theta \mathcal{L}_{\text{AM}}$  w.r.t. the AM objective:
           
$$\mathcal{L}_{\text{AM}}(\theta) := \mathbb{E}_{t \sim \mathcal{U}[0,1], (X_0, X_1, a_t) \sim \mathcal{B}_{\text{adj}}, X_t \sim p^{\text{base}}(\cdot | X_0, X_1)} \left[ \lambda_t \|u_\theta^{(k)}(t, X_t) + \sigma_t a_t\|^2 \right]$$

8:   end for
9:   for epoch in  $1, 2, \dots, M_{\text{crt}}$  do ▷ corrector matching
10:    Sample from model  $\{(X_0^{(i)}, X_1^{(i)})\}_{i=1}^N \sim p^{\bar{u}^{(k)}}$ , where  $\bar{u}^{(k)} = \text{stopgrad}(u_\theta^{(k)})$ 
11:    Update replay buffer  $\mathcal{B}_{\text{crt}} \leftarrow \mathcal{B}_{\text{crt}} \cup \{(X_0^{(i)}, X_1^{(i)})\}_{i=1}^N$ 
12:    Take  $L$  gradient steps  $\nabla_\phi \mathcal{L}_{\text{CM}}$  w.r.t. the CM objective:
           
$$\mathcal{L}_{\text{CM}}(\phi) := \mathbb{E}_{(X_0, X_1) \sim \mathcal{B}_{\text{crt}}} \left[ \|h_\phi^{(k)}(X_1) - \nabla_{x_1} \log p^{\text{base}}(X_1 | X_0)\|^2 \right]$$

13:   end for
14: end for
15: return Kinetic-optimal drift  $u^* \approx u_\theta(t, x)$ 

```

---

917 with  $\phi_s$  defined in (54). That is, we have concluded so far that

$$q^{p_1^{(k)}/\varphi_1} = \arg \min_q \{D_{\text{KL}}(p^{(k)} \| q) : q_1 = \nu\}. \quad (56)$$

918 Hence, it remains to be shown that the minimizer  $\bar{h}_1^{(k)}$  of the CM objective at stage  $k$  equals  $\frac{p_1^{(k)}}{\varphi_1}$ .

919 This is indeed the case since  $p^{(k)}$  is the SB solution:

$$\nabla \log \bar{h}^{(k)} \stackrel{(15)}{:=} \arg \min_h \mathbb{E}_{p_{0,1}^{(k)}} [\|h(X_1) - \nabla_{x_1} \log p^{\text{base}}(X_1 | X_0)\|^2] \stackrel{(37)}{=} \nabla \log \hat{\varphi}_1 \stackrel{(34)}{=} \nabla \log \frac{p_1^{(k)}}{\varphi_1}.$$

920 □

## 921 C Practical Implementation of ASBS

922 Algorithm 2 summarizes the practical implementation of ASBS, where we expand the adjoint and  
 923 corrector matching steps (*i.e.*, lines 3 and 4 in Algorithm 1) to full details. Table 5 provides the  
 924 hyper-parameters for each task. We break down each component as follows:

925 **Harmonic prior**  $\mu_{\text{harmonic}}$  Recall the harmonic prior in (19):

$$\mu_{\text{harmonic}}(x) \propto \exp(-\frac{\alpha}{2} \sum_{i,j} \|x_i - x_j\|^2). \quad (57)$$

926 In practice, we set  $\alpha = 1$  and implement (57) as an anisotropic Gaussian. For instance, for a 2-particle  
 927 system in 3D, *i.e.*,  $x = [x_1; x_2] \in \mathbb{R}^6$ , we can rewrite (57) as a quadratic potential,

$$\exp(-\frac{1}{2}\|x_1 - x_2\|^2) = \exp(x^\top R x), \quad \text{where } R = \begin{bmatrix} 1 & 0 & 0 & -\frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 & 0 & -\frac{1}{2} & 0 \\ 0 & 0 & 1 & 0 & 0 & -\frac{1}{2} \\ -\frac{1}{2} & 0 & 0 & 1 & 0 & 0 \\ 0 & -\frac{1}{2} & 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{2} & 0 & 0 & 1 \end{bmatrix}, \quad (58)$$

928 and then sample  $x$  from the Gaussian  $\mathcal{N}(x; 0, (R + \epsilon I)^{-1})$ , where we set  $\epsilon = 10^{-4}$ .

929 **Noise schedule  $\sigma_t$**  We consider two types of noise schedule.

- 930 • The *geometric noise schedule* (Song et al., 2021; Karras et al., 2022) monotonically decays from  
 931  $t = 0$  to 1 according to some prescribed  $\beta_{\min}$  and  $\beta_{\max}$ :

$$\sigma_t^{\text{geometric}} := \beta_{\min} \left( \frac{\beta_{\max}}{\beta_{\min}} \right)^{1-t} \sqrt{2 \log \frac{\beta_{\max}}{\beta_{\min}}}. \quad (59)$$

932 It is convenience to further define

$$\kappa_{t|s} := \int_s^t \sigma_\tau^2 d\tau \stackrel{\text{geometric}}{=} \beta_{\max}^2 \cdot \left( \left( \frac{\beta_{\min}}{\beta_{\max}} \right)^{2s} - \left( \frac{\beta_{\min}}{\beta_{\max}} \right)^{2t} \right), \quad \bar{\beta}^2 := \beta_{\max}^2 - \beta_{\min}^2, \quad \gamma_t := \frac{\kappa_{t|0}}{\bar{\beta}^2}. \quad (60)$$

933 With them, the conditional base distribution when  $f := 0$  can be represented compactly by

$$p^{\text{base}}(X_t|X_0) = \mathcal{N}(X_t; X_0, \kappa_{t|0}I) \quad (61a)$$

$$p^{\text{base}}(X_t|X_0, X_1) = \mathcal{N}(X_t; (1 - \gamma_t)X_0 + \gamma_t X_1, \bar{\beta}^2 \gamma_t (1 - \gamma_t)I) \quad (61b)$$

- 934 • The *constant noise schedule* simply sets

$$\sigma_t^{\text{constant}} := \sigma. \quad (62)$$

935 When  $f := 0$ , the base SDE is effectively a standard Brownian motion whose conditional  
 936 distributions obey

$$p^{\text{base}}(X_t|X_0) = \mathcal{N}(X_t; X_0, \sigma^2 t I) \quad (63a)$$

$$p^{\text{base}}(X_t|X_0, X_1) = \mathcal{N}(X_t; (1 - t)X_0 + tX_1, \sigma^2 t(1 - t)I) \quad (63b)$$

937 **Replay buffers  $\mathcal{B}_{\text{adj}}$  and  $\mathcal{B}_{\text{crt}}$**  Similar to many previous diffusion samplers (Havens et al., 2025;  
 938 Akhound-Sadegh et al., 2024; Chen et al., 2025), we employ replay buffers  $\mathcal{B}$  in computation of both  
 939 adjoint (14) and corrector (15) matching objectives. Specifically, we rebase the expectation over  
 940 model samples  $p^{u^{(k)}}$  onto a replay buffer  $\mathcal{B}$ , which stores the most latest  $|\mathcal{B}|$  samples. We update the  
 941 buffer with  $N$  new samples every  $L$  gradient steps. Note that the use of replay buffers effectively  
 942 render ASBS a hybrid method between on-policy and off-policy.

943 **Parametrization of  $u_\theta$  and  $h_\phi$**  For each energy function, we parametrize the drift  $u_\theta(t, x)$  and the  
 944 corrector  $h_\phi(x)$  with two neural networks,  $v_\theta(t, x)$  and  $v_\phi(t, x)$ , of the same architecture.

945 Specifically, we parametrize the drift as  $u_\theta(t, x) := \sigma_t v_\theta(t, x)$ , which effectively eliminates the noise  
 946 schedule “ $\sigma_t$ ” in matching target (see (14)), making it time-invariant for each sampled trajectory.  
 947 The only exception is the conformer generation task, where we keep the original parametrization  
 948  $u_\theta(t, x) := v_\theta(t, x)$ , which empirically yields better results. On the other hand, since  $h_\phi(x)$  is  
 949 independent of time, we simply set a fixed time input  $t = 1$ , *i.e.*,  $h_\phi(x) := v_\phi(1, x)$ .

950 The specific parametrization  $v(t, x)$  employed for each task are detailed below.

- 951 • **MW-5:** We consider  $v(t, x)$  a standard fully-connected network with 4 layers with 64 hidden  
 952 features of the following form:

$$\text{output} = \text{layer\_n} \circ \dots \circ \text{layer\_1} \circ (\text{x\_embed}(x) + \text{t\_embed}(t))$$



- *DW-4, LJ-13, LJ-55*: We consider  $v(t, x)$  a Equivariant Graph Neural Network (EGNN; [Satorras et al., 2021](#)) with 5 layers and 128 hidden features. The architecture of EGNN is aligned with prior methods ([Akhound-Sadegh et al., 2024](#); [Havens et al., 2025](#)).
- *Alanine dipeptide*: We use the same architecture as in MW-5, except with 8 layers with 256 hidden features.
- *Conformer generation*: We consider  $v(t, x)$  a similar EGNN used in Adjoint Sampling ([Havens et al., 2025](#)), except with 20 layers. Ablation study on the same EGNN architecture can be found in Appendix D.4.

**Clipping**  $\alpha_{\max}$  We clip the energy gradient to prevent its maximum norm from exceeding  $\alpha_{\max}$ .

**Time scaling**  $\lambda_t$  Following standard practices for AM objective, we employ a time scaling  $\lambda_t$  to improve numerical stability. Note that this does not affect the minimizer of the AM objective. We set  $\lambda_t := \frac{1}{\sigma_t^2}$  for all tasks.

**Translation invariance** For DW-4, LJ-13, LJ-55, and conformer generation tasks, we follow prior methods ([Akhound-Sadegh et al., 2024](#); [Havens et al., 2025](#)) by restricting the state space to a zero center-of-mass (ZCOM) subspace and thereby enforcing translation invariance.

For a  $n$ -particle  $k$ -dimensional system, *i.e.*,  $x = [x_1; \dots; x_n]$  where  $x_i \in \mathbb{R}^k$ , the ZCOM subspace is defined as  $\mathcal{X}^{\text{ZCOM}} = \{x \in \mathbb{R}^{nk} : \sum_{i=1}^n x_i = 0\}$ . Practically, this is achieved by projecting the initial sample  $X_0 \sim \mu$ , the SDE’s noise  $dW_t$ , and the energy gradient  $\nabla E(\cdot)$  onto  $\mathcal{X}^{\text{ZCOM}}$ . Note that the output of EGNN is by construction ZCOM.

Formally, the adaption is equivalent to augmenting the SDE with a projection matrix  $A \in \mathbb{R}^{nk \times nk}$ :

$$dX_t = \sigma_t A u_t(X_t) dt + \sigma_t A dW_t, \quad X_0 = A Y_0, \quad Y_0 \sim \mu, \quad A = \left( I_n - \frac{1}{n} \mathbf{1}_n \mathbf{1}_n^\top \right) \otimes I_k, \quad (64)$$

where  $\otimes$  is the Kronecker product,  $I_n \in \mathbb{R}^{n \times n}$  is an identity matrix, and  $\mathbf{1}_n \in \mathbb{R}^n$  is a vector of ones.

**Initialization and alternate procedure** As ASBS is an instantiation of the IPF algorithm (see Theorem 3.2), it must adhere to the IPF initialization protocol to ensure theoretical convergence to the global solution. Specifically, the IPF initialization can be implemented in two ways

- Initialize with  $h_\phi^{(0)} := 0$  and run AM, CM, ... until convergence. We adopt this setup for all tasks.
- Initialize with  $u_\theta^{(0)} := 0$  and run CM, AM, ... until convergence. Since  $p^{u^{(0)}} = p^{\text{base}}$  in this setup, the optimal corrector at the first CM stage is known analytically:

$$\begin{aligned} h^{(1)}(x) &\stackrel{(15)}{=} \int p_{0|1}^{\text{base}}(y|x) \nabla_x \log p_{1|0}^{\text{base}}(x|y) dy \\ &= \int \frac{p_{0|1}^{\text{base}}(y|x)}{p_{1|0}^{\text{base}}(x|y)} \nabla_x p_{1|0}^{\text{base}}(x|y) dy \\ &= \frac{1}{p_1^{\text{base}}(x)} \nabla_x \int p_0^{\text{base}}(y) p_{1|0}^{\text{base}}(x|y) dy \\ &= \nabla \log p_1^{\text{base}}(x) \end{aligned} \quad (65)$$

In practice, we find that the two setups yield similar performance.

**RDKit pretraining** This is a warm-start that pretrains the drift  $u_\theta$  using RDKit samples. The procedure is inspired by the fact that ([Shi et al., 2023](#); [Liu et al., 2023](#)):

$$\begin{aligned} u_t^* &= \sigma_t \nabla \log \varphi_t \\ &= \arg \min_{u_t} \mathbb{E}_{p_{t,1}^*} [\|u_t(X_t) - \sigma_t \nabla_{x_t} \log p^{\text{base}}(X_1|X_t)\|^2] \\ &= \arg \min_{u_t} \mathbb{E}_{(X_0, X_1) \sim p_{0,1}^*, X_t \sim p^{\text{base}}(\cdot|X_0, X_1)} [\|u_t(X_t) - \sigma_t \nabla_{x_t} \log p^{\text{base}}(X_1|X_t)\|^2]. \end{aligned} \quad (66)$$

Table 5: Hyperparameters of ASBS for the each task.

	Synthetic energy functions				Alanine dipeptide	Conformer generation
	MW-5	DW-4	LJ-13	LJ-55		
$\mu$	$\mathcal{N}(0, 1)$	$\mu_{\text{harmonic}}$	in (19) with $\alpha=1$		$\mathcal{N}(0, 0.25)$	$\mu_{\text{harmonic}}$
$\beta_{\min}$	—	0.001	0.001	0.001	0.001	0.001
$\beta_{\max}$	—	2	1	1	0.5	1
$\sigma$	0.2	—	—	—	—	—
$K$	5	20	15	15	15	3
$M_{\text{adj}}$	100	200	300	300	4000	2500
$M_{\text{crt}}$	20	20	200	20	2000	2000
$N$	1000	1000	1000	1000	1000	128
$L$	200	100	100	100	100	100
$ \mathcal{B} $	$10^4$	$10^4$	$10^4$	$10^4$	$10^4$	1000
$\alpha_{\max}$	—	100	100	100	100	150
$\lambda_t$	$\frac{1}{\sigma_t^2}$	$\frac{1}{\sigma_t^2}$	$\frac{1}{\sigma_t^2}$	$\frac{1}{\sigma_t^2}$	$\frac{1}{\sigma_t^2}$	$\frac{1}{\sigma_t^2}$

where the last equality is due to

$$\begin{aligned}
p_{0,t,1}^*(x, y, z) &\stackrel{(33)}{=} p_{t,1|0}^{\text{base}}(y, z|x) \hat{\varphi}_0(x) \varphi_1(z) \\
&= p_{t|0,1}^{\text{base}}(y|x, z) p_{1|t}^{\text{base}}(z|y) \hat{\varphi}_0(x) \varphi_1(z) && \text{by Markov property} \\
&\stackrel{(35)}{=} p_{t|0,1}^{\text{base}}(y|x, z) p_{0,1}^*(x, z). && (67)
\end{aligned}$$

Equation (66) can be understood as an analogy of (37) for another SB potential  $\varphi_t$ . In practice, given RDKit samples  $X_1 \sim q^{\text{RDKit}}$ , we implement the pretraining for ASBS by minimizing w.r.t. the following objective:

$$\begin{aligned}
\mathcal{L}_{\text{pretrain}}(\theta) &= \mathbb{E}_{t \sim \mathcal{U}[0,1], X_0 \sim \mu, X_1 \sim q^{\text{RDKit}}, X_t \sim p^{\text{base}}(\cdot|X_0, X_1)} \left[ \tilde{\lambda}_t \|u_t(X_t) - \sigma_t \nabla_{x_t} \log p^{\text{base}}(X_1|X_t)\|^2 \right] \\
&\stackrel{(61a)}{=} \mathbb{E}_{t \sim \mathcal{U}[0,1], X_0 \sim \mu, X_1 \sim q^{\text{RDKit}}, X_t \sim p^{\text{base}}(\cdot|X_0, X_1)} \left[ \tilde{\lambda}_t \|u_t(X_t) - \frac{\sigma_t}{\kappa_{1|t}} (X_1 - X_t)\|^2 \right], && (68)
\end{aligned}$$

where  $\kappa_{1|t}$  is defined in (60) for the geometric noise schedule. We set the time scaling  $\tilde{\lambda}_t := \sqrt{\frac{\sigma_t}{\kappa_{1|t}}}$ . Note that, unlike AS, the minimizer of (68) does not equal  $u^*$ , since  $(X_0, X_1) \sim \mu \otimes q^{\text{RDKit}} \neq p_{0,1}^*$  are sampled independently.

## D Experiment Details

### D.1 Synthetic Energy Functions

#### D.1.1 Energy functions

In this section, we provide the exact setup for our synthetic energy experiments in Table 2. We consider four synthetic energy functions that have been widely used in recent literature to benchmark sampling and generative algorithms: MW-5, DW-4, LJ-13, and LJ-55.

**MW-5** The MW-5 (Many-Well in 5D) energy is a 5-particle 1D system adopted from Chen et al. (2025), where  $x = [x_1; \dots; x_5] \in \mathbb{R}^5$  with  $x_i \in \mathbb{R}$ . The energy function is defined as follows:

$$E(x) = \sum_{i=1}^5 (x_i^2 - \delta)^2 \quad (69)$$

where we set  $\delta = 4$ . This creates distinct modes centered at combinations of  $\pm\sqrt{\delta}$  in each of the  $d$  dimensions.

**DW-4** The DW-4 (Double-Well for 4 particles in 2D) energy is a physically motivated pairwise potential originally proposed in Köhler et al. (2020) and subsequently used in Akhound-Sadeh et al. (2024); Havens et al. (2025). It defines a system of four particles, each living in  $\mathbb{R}^2$ , leading to an 8D state vector  $x = [x_1; x_2; x_3; x_4] \in \mathbb{R}^8$  with  $x_i \in \mathbb{R}^2$ . The energy function reads

$$E(x) = \exp \left[ \frac{1}{2\tau} \sum_{i < j} (a(d_{ij} - d_0) + b(d_{ij} - d_0)^2 + c(d_{ij} - d_0)^4) \right], \quad (70)$$

where  $d_{ij} = \|x_i - x_j\|_2$  is the Euclidean distance between particles  $i$  and  $j$ . We follow the standard configuration with  $a = 0$ ,  $b = -4$ ,  $c = 0.9$ ,  $d_0 = 1$ , and temperature  $\tau = 1$ .

**LJ-13 and LJ-55** The Lennard-Jones (LJ) potentials are classical intermolecular potentials commonly used in physics to model atomic interactions. These are defined for a system of  $n$  particles in 3D space, with  $x = [x_1; \dots; x_n] \in \mathbb{R}^{3n}$  and  $x_i \in \mathbb{R}^3$ . The index following "LJ-" indicates the number of particles (e.g., 13 or 55). The unnormalized energy function takes the form:

$$E(x) = \frac{\epsilon}{2\tau} \sum_{i < j} \left[ \left( \frac{r_m}{d_{ij}} \right)^6 - \left( \frac{r_m}{d_{ij}} \right)^{12} \right] + \frac{c}{2} \sum_i \|x_i - C(x)\|^2, \quad (71)$$

where  $d_{ij} = \|x_i - x_j\|_2$  is the pairwise distance and  $C(x)$  denotes the center of mass of the particles. We use the parameter values  $r_m = 1$ ,  $\epsilon = 1$ ,  $c = 0.5$ , and  $\tau = 1$ , following prior work. The LJ-13 and LJ-55 systems correspond to 39D and 165D, respectively.

## D.1.2 Baselines

Here, we outline the procedure used to obtain the values reported in Table 2 for the baseline methods.

For PIS (Zhang and Chen, 2022), DDS (Vargas et al., 2023), and LV-PIS (Richter and Berner, 2024), iDEM (Akhound-Sadeh et al., 2024), and AS (Havens et al., 2025), we reuse the values reported in AS (Havens et al., 2025, Table 1) for DW-4, LJ-13, and LJ-55 energy functions. As for MW-5, which is not included in AS, we run iDEM using their official implementation and the rest of baseline methods using our own implementation in PyTorch (Paszke et al., 2019). We were unable to obtain reportable results for LV-PIS and iDEM on this energy function.

For PDDS (Phillips et al., 2024) and SCLD (Chen et al., 2025), we run their official implementations in JAX (Bradbury et al., 2018) using the default hyperparameter settings specified for the Log-Gaussian Cox Process experiment in their respective papers. To enhance stability and convergence on synthetic energy functions, we tune the gradient clipping parameters. For PDDS, we apply clipping to the gradient of the energy function. For SCLD, we clip both the energy gradient and the Langevin norm. In both cases, the clipping magnitude is selected from the set  $\{1, 10, 100, 1000\}$  based on the best validation performance. Training is performed for 100,000 iterations across all runs. For SCLD, we do not use subtrajectory splitting.

## D.1.3 Evaluation Metrics

In this subsection, we outline the evaluation criteria used to quantitatively assess the quality of samples generated from synthetic energy functions. We employ three primary metrics: Sinkhorn distance, geometric  $\mathcal{W}_2$ , and energy  $\mathcal{W}_2$ , each designed to capture different aspects of distributional similarity between generated and ground truth samples.

**Sinkhorn distance** To evaluate the similarity between the empirical distributions of generated and reference samples, we compute the Sinkhorn distance using the entropy-regularized optimal transport formulation (Peyré and Cuturi, 2019), following the implementation of Blessing et al. (2024) and Chen et al. (2025). The Sinkhorn regularization coefficient is set to  $10^{-3}$  throughout. We use 2,000 samples from both the generated and ground truth distributions to compute the metric.

**Geometric  $\mathcal{W}_2$**  For DW and LJ tasks, the potential energy functions—and consequently, the sample distributions—exhibit invariance to both particle permutations and rigid transformations such as rotations and reflections. To appropriately account for these symmetries, we employ the geometric

1042  $\mathcal{W}_2$  distance as defined by [Akhound-Sadeh et al. \(2024\)](#) and [Havens et al. \(2025\)](#). Formally, the  
1043 2-Wasserstein distance is computed as:

$$\mathcal{W}_2^2(\hat{\nu}, \nu) = \inf_{\pi \in \Pi(\hat{\nu}, \nu)} \int D(x, y)^2 \pi(x, y) dx dy, \quad (72)$$

1044 where  $\Pi(\hat{\nu}, \nu)$  denotes the set of joint couplings with prescribed marginals  $\hat{\nu}$  (generated) and  $\nu$   
1045 (ground truth), and  $D(x, y)$  is a symmetry-aware distance between samples defined as:

$$D(x, y) = \min_{R \in O(s), P \in S(n)} \|x - (R \otimes P)y\|_2. \quad (73)$$

1046 Here,  $O(s)$  denotes the group of orthogonal transformations in  $s$  spatial dimensions (rotations and  
1047 reflections), and  $S(n)$  represents the symmetric group over  $n$  particles. As exact minimization over  
1048 these symmetry groups is computationally infeasible, we adopt the approximation scheme of [Köhler  
et al. \(2020\)](#). We use 2000 samples from each generated and ground truth distribution to compute the  
1049 metric.  
1050

1051 **Energy  $\mathcal{W}_2$**  To evaluate fidelity with respect to the target energy landscape, we also compute  
1052 the 2-Wasserstein distance between the energy values of generated samples and those of ground  
1053 truth samples. For each target distribution, we generate 2,000 samples from both the model and the  
1054 reference, and compare their respective energy histograms. This scalar-based Wasserstein metric  
1055 serves as a proxy for how well the generative model captures the energy histogram of the target  
1056 distribution.

## 1057 D.2 Alanine dipeptide

1058 **Benchmark description** We adopt the experiment setup primarily from [\(Midgley et al., 2023\)](#).  
1059 Given a configuration of alanine dipeptide, which consists of 22 particles in 3D, *i.e.*,  $x =$   
1060  $[x_1; \dots; x_{22}] \in \mathbb{R}^{66}$  where  $x_i \in \mathbb{R}^3$ , we apply the same coordinate transform  $\mathcal{T}$  proposed by  
1061 [Midgley et al. \(2023\)](#). This coordinate transform maps the Cartesian coordinates to internal coor-  
1062 dinates,  $\mathcal{T}(x) =: z \in \mathbb{R}^{60}$ , which include bond lengths, bond angles, and dihedral angles ([Stimper  
et al., 2022](#)). This process effectively removes six degrees of freedom—three for translation and three  
1063 for rotation—thereby enforcing structural invariance. Note that the internal coordinate transformation  
1064 is bijective. Hence, we can compute the energy via  
1065

$$E(x) = E(\mathcal{T}^{-1}(z)) \quad (74)$$

1066 **Evaluation and baselines** For each sample  $x = \mathcal{T}^{-1}(z) \in \mathbb{R}^{66}$ , we extract five torsion angles,  
1067 including the backbone angles  $\phi, \psi$  and methyl rotation angles  $\gamma_1, \gamma_2, \gamma_3$ . We report two divergence  
1068 metrics with respect to the ground-truth distribution, which contains  $10^7$  samples simulated by  
1069 Molecular Dynamics. We implement the baseline methods, including PIS ([Zhang and Chen, 2022](#)),  
1070 DDS ([Vargas et al., 2023](#)), AS ([Havens et al., 2025](#)), using PyTorch ([Paszke et al., 2019](#)).

1071 For the KL divergences, we adopt setup from [\(Wu et al., 2020\)](#) and compute the divergence of the  
1072 ground-truth marginal to model marginal for each torsion angle:

$$D_{\text{KL}}(p^*(\cdot) || p^{u_\theta}(\cdot)) \approx \sum P^*(\cdot) \log \frac{P^*(\cdot) + \epsilon}{P^{u_\theta}(\cdot) + \epsilon}, \quad \epsilon = 10^{-5}, \quad (75)$$

1073 where  $P^*$  and  $P^{u_\theta}$  are histograms of  $10^7$  samples, discretized between  $[-\pi, \pi]$  with 200 intervals.

1074 For the Wasserstein-2 distance, we use the Geometric  $\mathcal{W}_2$  in (72), where each sample is now in 2D,  
1075  $x = [\phi, \psi] \in \mathbb{R}^2$ . Due to the high computational cost, we compute the value using a subset of  $10^4$   
1076 samples from the test set ground-truth samples, which is fixed for all methods.

1077 Finally, both Ramachandran plots in Figure 5 are generated using  $10^7$  samples.

## 1078 D.3 Amortized conformer generation

1079 In this subsection, we provide some context for the experimental results found in Table 4 regarding  
1080 the generation of conformers.

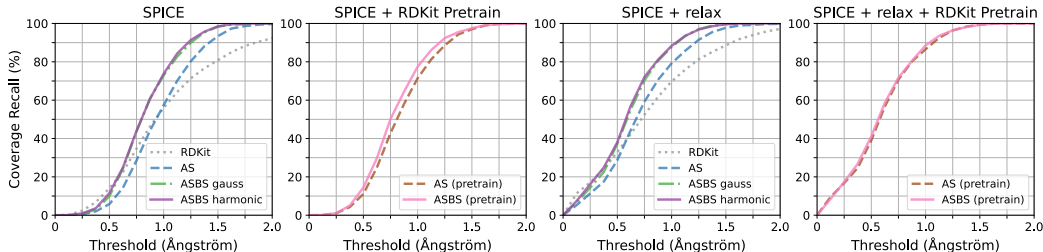


Figure 6: Recall coverage curve on amortized conformer generation on the SPICE test set.

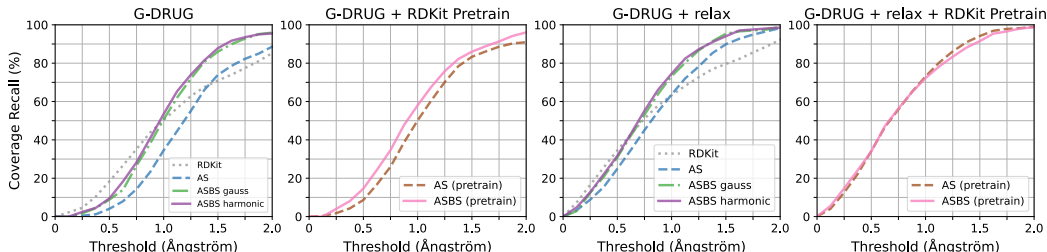


Figure 7: Recall coverage curve on amortized conformer generation on the GEOM-DRUG test set.

**Benchmark description** Conformers are atomic representations of molecules in cartesian space with their constituent atoms arranged into local minima on the potential energy surface. Molecules are defined to be a graph of atoms (nodes) connected by bonds (edges); conformers are geometric realizations of that molecule. Torsion angles, or rotatable bonds, are particularly important degrees of freedom for defining conformations since bond lengths and bond angles are typically much more stable due to a high sensitivity to perturbations. It is common to consider bond lengths and bond angles fixed, while the torsional degrees of freedom define the conformer.

The task in this benchmark is to take a representation of the molecular graph, usually a SMILES string (Weininger, 1988), and comprehensively sample the conformational configuration space. In flexible molecules, there can be a large number of conformers with many separated modes in a  $3n - 6$  dimensional space. (Where  $n$  represents the number of atoms and 6 comes from the irrelevance of rotation and translation of the conformer.) We quantify the notion of comprehensively sampling the space by comparing generated structures to a set of conformers sampled using expensive, standard search techniques (Pracht et al., 2024) that were further relaxed using extremely precise density function theory-based, quantum chemistry methods (Neese, 2012; Levine et al., 2025). A detailed description of this benchmark can be found in its source (Havens et al., 2025, Appendix F).

**Evaluation and baselines** The method of comparison between proposed structure and reference conformer is to use RDKit’s (Landrum, 2006) implementation of *Root Mean Squared Displacement* (RMSD), a measure of distance between atomic structures that is invariant to translation and rotation. We set a threshold RMSD for two structures to match and computed the Recall Coverage and Recall Average Minimum RMSD (AMR). The experiment was performed with both generated structures and with generated structures after a so-called relaxation, i.e. geometry optimization of energy, using eSEN (Fu et al., 2025). The equations for computing these metrics are:

$$\text{COV-R}(\delta) := \frac{1}{L} |\{l \in \{1, \dots, L\} : \exists k \in \{1, \dots, K\}, \text{RMSD}(C_k, C_l^*) < \delta\}| \quad (76)$$

$$\text{AMR-R} := \frac{1}{L} \sum_{l \in \{1, \dots, L\}} \min_{k \in \{1, \dots, K\}} \text{RMSD}(C_k, C_l^*) \quad (77)$$

where  $\delta = 0.75 \text{ \AA}$  is the coverage threshold,  $L = \max(L', 128)$ , where  $L'$  is the number of reference conformers,  $K = 2L$ , and let  $\{C_l^*\}_{l \in [1, L]}$  and  $\{C_k\}_{k \in [1, K]}$  be the sets of ground truth and generated conformers respectively. We capped the reference conformers per molecule at 512 in COV-R.

The values for the baselines are adopted from AS (Havens et al., 2025).

Table 6: Ablation study on amortized conformer generation using the same EGNN architecture as in AS (Havens et al., 2025). We report the recall at the thresholds **0.75Å** and **1.25Å**, where the latter was reported in AS.

Method	without relaxation				with relaxation			
	SPICE		GEOM-DRUGS		SPICE		GEOM-DRUGS	
	Coverage $\uparrow$	AMR $\downarrow$	Coverage $\uparrow$	AMR $\downarrow$	Coverage $\uparrow$	AMR $\downarrow$	Coverage $\uparrow$	AMR $\downarrow$
Threshold 0.75Å	AS (Havens et al., 2025)	28.56 $\pm$ 32.84	0.96 $\pm$ 0.28	14.32 $\pm$ 25.20	1.27 $\pm$ 0.52	59.21 $\pm$ 33.80	0.71 $\pm$ 0.29	45.07 $\pm$ 34.94
	ASBS w/ Gaussian prior (Ours)	38.98 $\pm$ 35.26	0.88 $\pm$ 0.25	21.48 $\pm$ 29.73	1.08 $\pm$ 0.37	65.86 $\pm$ 31.63	0.64 $\pm$ 0.25	48.38 $\pm$ 34.66
	ASBS w/ Harmonic prior (Ours)	40.83 $\pm$ 36.24	0.86 $\pm$ 0.24	24.51 $\pm$ 31.93	1.05 $\pm$ 0.42	68.84 $\pm$ 31.56	0.61 $\pm$ 0.24	50.83 $\pm$ 33.58
	AS +RDKit pretrain (Havens et al., 2025)	42.79 $\pm$ 35.77	0.85 $\pm$ 0.25	25.98 $\pm$ 31.89	1.05 $\pm$ 0.49	70.75 $\pm$ 30.75	0.60 $\pm$ 0.25	55.90 $\pm$ 33.95
	ASBS +RDKit pretrain (Ours)	44.80 $\pm$ 37.11	0.83 $\pm$ 0.24	28.30 $\pm$ 30.98	0.98 $\pm$ 0.34	70.10 $\pm$ 30.19	0.60 $\pm$ 0.24	53.17 $\pm$ 32.67
Threshold 1.25Å	AS (Havens et al., 2025)	80.28 $\pm$ 27.68	0.96 $\pm$ 0.28	55.05 $\pm$ 37.93	1.27 $\pm$ 0.52	91.49 $\pm$ 18.27	0.71 $\pm$ 0.29	78.26 $\pm$ 29.74
	ASBS w/ Gaussian prior (Ours)	86.91 $\pm$ 22.01	0.88 $\pm$ 0.25	69.86 $\pm$ 33.02	1.08 $\pm$ 0.37	95.06 $\pm$ 10.79	0.64 $\pm$ 0.25	81.09 $\pm$ 29.08
	ASBS w/ Harmonic prior (Ours)	89.53 $\pm$ 19.46	0.86 $\pm$ 0.24	73.86 $\pm$ 32.71	1.05 $\pm$ 0.42	97.02 $\pm$ 7.20	0.61 $\pm$ 0.24	83.55 $\pm$ 25.94
	AS +RDKit pretrain (Havens et al., 2025)	88.58 $\pm$ 19.96	0.85 $\pm$ 0.25	70.16 $\pm$ 35.39	1.05 $\pm$ 0.49	96.36 $\pm$ 8.92	0.60 $\pm$ 0.25	86.14 $\pm$ 23.73
	ASBS +RDKit pretrain (Ours)	90.69 $\pm$ 17.91	0.83 $\pm$ 0.24	77.34 $\pm$ 30.82	0.98 $\pm$ 0.34	97.23 $\pm$ 6.75	0.60 $\pm$ 0.24	85.64 $\pm$ 25.16

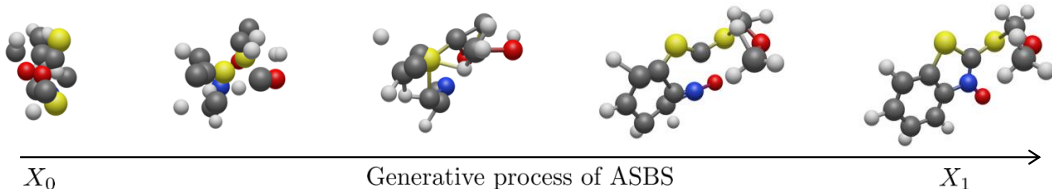


Figure 8: Generative process of ASBS with harmonic prior  $X_0 \sim \mu_{\text{harmonic}}$ , conditioned on a unseen molecular topology  $g \in \mathcal{G}_{\text{test}}$  from the SPICE test set, COCSc1sc2ccccc2[n+]1[0-].

#### D.4 Additional Experiments and Discussions

**Full recall curves** Figures 6 and 7 reports the full recall curves that complement Table 4. In practice, we observe that ASBS achieves slightly better results with a harmonic prior compared to a Gaussian prior, with both significantly outperforming AS (Havens et al., 2025). The superior performance of ASBS over AS is consistent across all setups, except for GEOM-DRUG with relaxation and RDKit pretraining (last plot in Figure 7), where ASBS falls slightly behind AS at higher thresholds.

**Generation process of ASBS** Figure 8 visualizes the generative process of ASBS with harmonic prior  $X_0 \sim \mu_{\text{harmonic}}$ , conditioned on a unseen molecular topology  $g \in \mathcal{G}_{\text{test}}$  from the SPICE test set, COCSc1sc2ccccc2[n+]1[0-].

**Ablation study between AS and ASBS using the same EGNN** For the amortized conformer generation task in Table 4, we use an EGNN architecture with 20 layers, whereas AS employs the same architecture with 12 layers. In Table 6, we report the results of ASBS using the same 12-layer EGNN as AS. Notably, our ASBS consistently outperforms AS—by 5–18%—on both test sets without relaxation and pretraining. With pretraining, ASBS continues to outperform AS when relaxation is not applied, and achieves comparable results (within 2.8%) when relaxation is used.

**Discussion on important weights** Finally, we discuss the potential integration of ASBS with importance weights, emphasizing that our theoretical and algorithmic frameworks do not preclude the use of importance weights to further enhance performance or robustness.

Formally, the importance weights over model path  $X \sim p^u$  admit the following representation:

$$w(X) := \frac{dp^*(X)}{dp^u(X)} = \exp \left( \int_0^1 -\frac{1}{2} \|u_t(X_t)\|^2 dt - \int_0^1 u_t(X_t) \cdot dW_t - \log \frac{\hat{\varphi}_1(X_1)}{\nu(X_1)} + \log \frac{\hat{\varphi}_0(X_0)}{\mu(X_0)} \right), \quad (78)$$

which can be obtained from (51) by setting  $\bar{h} := \hat{\varphi}_1$  so that  $q^{\bar{h}} = p^*$  is the optimal distribution of SB.

1128 Note that when the source distribution degenerates to the Dirac delta  $\mu(X_0) = \delta_0(X_0)$ , the last term  
 1129  $\log \frac{\hat{\varphi}_0(X_0)}{\mu(X_0)}$  becomes a constant and—as discussed in Section 3.2— $\hat{\varphi}_1 = p_1^{\text{base}}$ , thereby recovering the  
 1130 weights used in prior SOC-based methods (Zhang and Chen, 2022; Havens et al., 2025).

1131 Equation (78) is also a more concise representation than the one derived in (Richter and Berner, 2024),  
 1132 by recognizing the following relation through the application of Ito Lemma (38) to  $\log \hat{\varphi}_t(X_t)$ :

$$\frac{\log \hat{\varphi}_1(X_1)}{\log \hat{\varphi}_0(X_0)} = \int_0^1 \left[ \frac{1}{2} \|v_t(X_t)\|^2 + (u_t \cdot v_t)(X_t) + \nabla \cdot (\sigma_t v_t(X_t) - f_t(X_t)) \right] dt + \int_0^1 v_t(X_t) \cdot dW_t, \quad (79)$$

1133 where we shorthand  $v_t(x) := \sigma_t \nabla \log \hat{\varphi}_t(x)$ .

1134 Estimating the weight in (78) requires knowing the ratios  $\frac{\hat{\varphi}_1(x)}{\nu(x)}$  and  $\frac{\hat{\varphi}_0(x)}{\mu(x)}$ , which are not immediately  
 1135 available with the current parametrization,  $u_\theta(t, x) \approx \sigma_t \nabla \log \varphi_t(x)$  and  $h_\phi(x) \approx \nabla \log \hat{\varphi}_1(x)$ . One  
 1136 accommodation is to reparametrize the functions with potential network  $v(t, x) : [0, 1] \times \mathcal{X} \rightarrow \mathbb{R}$ ,

$$u_\theta(t, x) := \sigma_t \nabla v_\theta(t, x), \quad h_\phi(x) := \nabla v_\phi(1, x) \quad (80)$$

1137 and then regress their gradients onto the adjoint and corrector targets. With that, the logarithmic  
 1138 ratios can be easily estimated:

$$\log \frac{\hat{\varphi}_1(x)}{\nu(x)} = v_\phi(1, x) + E(x), \quad \log \frac{\hat{\varphi}_0(x)}{\mu(x)} \stackrel{(34)}{=} -\log \varphi_0(x) = v_\theta(0, x). \quad (81)$$

1139 A more detailed investigation of this importance sampling scheme is left for future work.