MINIMAL-ACTION DISCRETE SCHRÖDINGER BRIDGE MATCHING FOR PEPTIDE SEQUENCE DESIGN

Shrey Goel[1], **Pranam Chatterjee**[2,3,†]

[1]Department of Computer Science, Duke University
[2]Department of Computer and Information Science, University of Pennsylvania
[3]Department of Bioengineering, University of Pennsylvania

[†]Corresponding author: pranam@seas.upenn.edu

ABSTRACT

Generative modeling of peptide sequences requires navigating a discrete and highly constrained space in which many intermediate states are chemically implausible or unstable. Existing discrete diffusion methods rely on reversing fixed corruption processes or following prescribed probability paths, which requires numerous sampling steps and forces generation through low-likelihood regions. We introduce **M**inimal-**a**ction **d**iscrete **S**chrödinger **B**ridge **M**atching (**MadSBM**), a rate-based generative framework for peptide design that formulates generation as a controlled continuous-time Markov process on the amino-acid tokens. To yield probability trajectories that remain near high-likelihood sequence spaces throughout generation, MadSBM 1) defines generation relative to a biologically informed reference process and 2) learns a control field that biases transition rates to produce transport paths from a masked prior to the data distribution. We further apply Sampling-Importance-Resampling with MadSBM to expand the design space of therapeutic peptide sequences; to our knowledge, this is the first application of discrete classifier guidance-like methods to a Schrödinger bridge-based generative framework. We release our code here.

1 INTRODUCTION

Generative modeling has become a central tool for peptide and protein design, enabling data-driven discovery of binders, modulators, and degraders across diverse biological targets (Chen et al., 2025b;a; Goel et al., 2025; Hong et al., 2025; Pacesa et al., 2025; Tang et al., 2025b; Vincoff et al., 2025b;a; Zhang et al., 2025a). Recent autoregressive, diffusion, and flow-based models generate sequences by reversing fixed corruption processes or interpolating in probability space (Chen et al., 2025c; Tang et al., 2025a;c; Zhang et al., 2025b; Stark et al., 2024). While effective, these approaches impose prescribed probability paths that can force generation through low-likelihood or unstable intermediates (Ho et al., 2020; Sahoo et al., 2024; Lipman et al., 2022; Domingo i Enrich et al., 2024), which is problematic for biological sequences that are highly sensitive to local perturbations.

An alternative is treating generation as a transport problem (Chen et al., 2021). Optimal transport and Schrödinger bridge formulations define stochastic paths that connect prior to data distributions while minimizing an action functional (Schrödinger, 1932; Léonard, 2013; De Bortoli et al., 2021). In continuous settings this yields smoother and more stable generative trajectories, but classical Schrödinger bridge solvers rely on forward and backward iterations that are difficult to scale to discrete sequence spaces with large vocabularies and edit-based dynamics (Shi et al., 2023; Vargas et al., 2021; Genevay et al., 2018). Although recent work has extended Schrödinger bridge methods to discrete domains, including iterative forward-backward projections for categorical variables and diffusion-style CTMC bridges, these approaches are validated on VQ image tokens or graph data that do not directly align with long, language-like sequence spaces (Kim et al., 2024; Ksenofontov & Korotin, 2025). Recent simulation-free transport methods demonstrate that optimal transport paths can be approximated without fully solving the forward and backward bridge (Tong et al., 2023), suggesting that a discrete variant may be feasible for biological sequences.

In this work, we introduce **Minimal-Action Discrete Schrödinger Bridge Matching (MadSBM)**, a discrete generative framework that models peptide sequence evolution as a controlled continuous-time Markov chain on the amino-acid edit graph. MadSBM tilts a biologically informed reference process through learnable control fields so that generation follows low-action transport paths that remain near high-probability regions of sequence space, enabling efficient sampling without relying on fixed interpolation schemes as in diffusion or flow models.

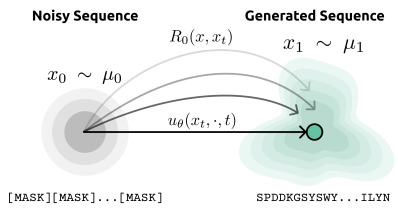

Figure 1: **MadSBM diagram**. We leverage a principled reference process R_0 so MadSBM requires only a time-conditioned control field u_θ to steer samples toward high-likelihood regions of the sequence space.

We introduce a **simulation-free learning rule** for Schrödinger bridge control that reduces training to a simple cross-entropy objective, avoiding explicit forward–backward bridge solvers while remaining consistent with minimal-action transport. Finally, we show MadSBM improves sample efficiency and stability over discrete diffusion baselines and supports objective-guided peptide design under low sampling budgets.

2 Related Works

A growing line of bridge-based methods have interpreted steering generative models as sampling from an exponentially tilted path measure of the form $p_{\text{target}} \propto p_{\text{base}} \exp(r(x))$, where p_{base} is a pre-trained generative process and $r(x)$ is a reward or energy function (Domingo-Enrich et al., 2024; Potaptchik et al., 2025). In this view, improved generative quality is achieved by modifying the transition dynamics of an *existing* diffusion model to favor high-reward trajectories, and recent work has extend this idea to discrete domains by tilting a pre-trained discrete diffusion models (Tang et al., 2025d). Critically, these approaches assume access to a learned base path measure that already transports noise to data and apply control as a post hoc modification of the learned process. In contrast, MadSBM addresses *de novo* generation in discrete sequence spaces, where no pre-trained path measure is available. We show that a carefully designed reference stochastic process leads to a simple training objective for learning a *control field* that tilts this process into a Schrödinger bridge between the noise and data distribution.

3 Preliminaries

3.1 Discrete Sequences and Markov Dynamics

Let $x = (x_1, \ldots, x_L) \in \{0,1\}^{L \times \mathcal{V}}$ denote a discrete sequence of length L, where each token is represented as a one-hot vector over a vocabulary $\mathcal{V}$. The time evolution of a sequence is modeled as a Continuous-Time Markov Chain (CTMC) with generator R, where off-diagonal entries $R(x, x')$ define instantaneous transition rates. We use this to introduce the reference and controlled processes that shape the generator-induced dynamics toward target distributions.

3.2 Reference and Controlled Processes

Reference Dynamics Schrödinger bridge models begin by specifying a simple reference path measure $\mathbb{P}_0$ (Schrödinger, 1932; Léonard, 2013). In discrete sequence spaces, a natural baseline is an uninformed CTMC generator R_0 corresponding to a uniform random walk over sequence tokens.

$$R_0(x, x') = \begin{cases} \frac{1}{L|\mathcal{V}|} & x' \neq x, \\ -\sum_{y \neq x} R_0(x, y) & x' = x. \end{cases} \tag{1}$$

The process evolves tokens independently and quickly drifts into low-probability and biologically implausible regions of sequence space. Although not suitable for direct sequence generation, this process provides a canonical reference relative to which a *controlled process* can be defined.

Exponential Tilting We formulate generation as steering the reference process toward the data distribution along an optimal transport path, *i.e.* matching endpoint marginals while minimizing deviation from $\mathbb{P}_0$. To this end, we parameterize a controlled generator R_u via an exponential tilt of the reference rates $R_u(x, x') = R_0(x, x') \exp\big(u(x, x')\big)$. Here, the control field $u : \mathcal{V}^L \times \mathcal{V}^L \to \mathbb{R}$ acts as a learnable log-potential on transitions, which are combined with the reference generator to

define time-dependent transition rates. This construction ensures that the controlled path measure $\mathbb{P}_u$ remains absolutely continuous with respect to $\mathbb{P}_0$, as required for the Schrödinger bridge formulation, while allowing the model to encode context-dependent biases from the reference over sequence edits.

3.3 The Schrödinger Bridge Objective

Given endpoint distributions μ_0 and μ_1 over discrete sequences, the Schrödinger bridge problem seeks a controlled stochastic process whose path measure minimizes relative entropy to a reference process $\mathbb{P}_0$ (Schrödinger, 1932):

$$\mathbb{P}_{u^\star} = \arg\min_{\mathbb{P}_u \in \mathcal{P}} \left\{ \mathrm{KL}(\mathbb{P}_u \,\|\, \mathbb{P}_0) \mid (\mathbb{P}_u)_0 = \mu_0,\ (\mathbb{P}_u)_T = \mu_1 \right\}. \tag{2}$$

The solution $\mathbb{P}_{u^\star}$ defines a stochastic interpolation between μ_0 and μ_1. Classical approaches compute this solution via iterative forward–backward projections of the Markov semigroup (Vargas et al., 2021; Genevay et al., 2018), which are intractable in high-dimensional discrete sequence spaces (Sokolov & Korotin, 2025).

4 Methods

We now formalize Minimal-Action Discrete Schrödinger Bridge Matching as a generative transport problem on discrete biological sequences. We consider a reference continuous-time Markov process on sequence space and a controlled process that transports a simple prior distribution to the data distribution along low-action paths. Our objective is to learn a parametric control field u_θ whose induced path measure follows the minimal-action Schrödinger bridge connecting the prior and the data.

4.1 Path Measures and Relative Entropy

We posit that learning the path measure $\mathbb{P}_u$ induced by the controlled generator R_u is directly possible through a Schrödinger bridge objective in Eq. equation 2. To explicitly solve this minimization problem, we must quantify the KL divergence between the controlled path measure $\mathbb{P}_u$ and the reference $\mathbb{P}_0$ induced by the controlled and uninformed generators R_u and R_0, respectively. Critically, we derive a tractable form of the path-space KL divergence:

Theorem 4.1. *(Path-space KL decomposition for CTMCs). Let $\mathbb{P}_u$ and $\mathbb{P}_0$ be path measures induced by time-inhomogeneous CTMCs. The relative entropy is the time-integral over instantaneous intensity differences.*

$$\mathrm{KL}(\mathbb{P}_u \,\|\, \mathbb{P}_0) = \mathbb{E}_{\mathbb{P}_u}\left[\int_0^T \sum_{x' \neq X_t} \Big(R_u(X_t, x') \log \frac{R_u(X_t, x')}{R_0(X_t, x')} - R_u(X_t, x') + R_0(X_t, x') \Big)\, dt \right]. \tag{3}$$

The proof is given in Appendix A.1.1.

4.2 The Action Functional

In statistcal physics, the *action* quantifies the accumulated cost of forcing a system away from its equilibrium dynamics. Naturally, we seek the minimal action required to steer the reference process toward the data distribution using the controlled path measure $\mathbb{P}_u$. The cost of this intervention is quantified by the relative entropy between the controlled and reference path measures, which simplifies to the action functional.

Corollary 4.2 (The Action Functional)**.** *By substituting the exponential tilt parameterization, $R_u(x, x') = R_0(x, x') \exp(u(x, x'))$, into the general relative entropy form (Theorem 4.1), we simplify the path-space relative entropy into the **Action Functional** $\mathcal{A}(u)$:*

$$\mathcal{A}(u) = \mathbb{E}_{\mathbb{P}_u}\left[\int_0^T \sum_{x' \neq X_t} R_0(X_t, x')\, \Psi\big(u(X_t, x')\big)\, dt \right], \tag{4}$$

where $\Psi(z) = ze^z - e^z - 1$ is the strictly convex cost function associated with the local change of measure.

The proof is given in Appendix A.2. Consequently, minimal-action transport seeks a control $u^\star$ that minimizes $\mathcal{A}(u)$ subject to prescribed endpoint constraints. The resulting process represents the *path of least resistance* that transports the prior to the data distribution with the minimum amount of information injected into the reference dynamics to achieve the desired behavior.

4.3 Learning Minimal-Action Control Fields

To minimize the action functional $A(u)$ without solving the full Schrödinger bridge, we seek a time-dependent control field $u_\theta(x, x', t) : \mathcal{V}^L \times \mathcal{V}^L \times [0, 1] \to \mathbb{R}^{L\times\mathcal{V}}$ that approximates the optimal control $u^\star$. In MadSBM, this control field can be learned with a neural network with parameters θ, which in turn defines a learned, time-dependent generator for the controlled dynamics $R_{\theta,t}(x, x') = R_0(x, x') \exp\big(u_\theta(x, x', t)\big)$. with diagonal entries $R_{\theta,t}(x, x) = -\sum_{x'\neq x} R_{\theta,t}(x, x')$. A similar modification of time-dependency can be made to the cost function $\Psi(u(X_t, x', t))$ used to define the relative entropy between the path measures $\mathbb{P}_u$ and $\mathbb{P}_0$ in Eq. equation 3 and the action of control $\mathcal{A}(u) = \mathbb{E}_{\mathbb{P}_u}\left[\int_0^T \sum_{x'\neq X_t} R_0(X_t, x')\, \Psi\big(u(X_t, x', t)\big)\, dt\right]$ in Eq. equation 4.

Since our rate matrix is a tilted uniform generator, the time-evolution of this marginal density satisfies the discrete Kolmogorov Forward Equation (KFE). If we let ρ_t^θ denote the marginal distribution of X_t under $R_{\theta,t}$ the discrete KFE can be written as $\frac{d}{dt}\rho_t^\theta(x) = \sum_y \rho_t^\theta(y) R_{\theta,t}(y, x) - \rho_t^\theta(x) \sum_y R_{\theta,t}(x, y)$.. Integrating this controlled process backward from $t = 1 \to 0$ yields a generative procedure analogous to the backward integration of continuous flows. Our formulation overall defines the process $(R_{\theta,t}, \mu_0)$ as the MadSBM generative model.

4.4 Training MadSBM

Optimal Couplings and Interpolation To learn a time-dependent control field, MadSBM requires intermediate sequence states x_t interpolating between a noise distribution μ_0 and the data distribution μ_1. We construct such states by pairing fully masked sequences with clean targets and applying a simple masking-based interpolation.

Specifically, we draw a timestep $k \sim \mathrm{Uniform}\{1, \ldots, T\}$ and set $t = k/T$. At time $t \in [0, 1]$, each token $x_t^{(i)}$ independently reveals the target token $x_1^{(i)}$ with probability t and remains masked with probability $1 - t$, defining the forward perturbation kernel

$$p_t\big(x^{(i)} \mid x_1^{(i)}\big) = (1 - t)\, \delta_{\mathcal{M}}(x^{(i)}) + t\, \delta_{x_1^{(i)}}(x^{(i)}), \tag{5}$$

where $\delta_{\mathcal{M}}$ is the Kronecker delta on the mask token $\mathcal{M}$. Uniform timestep sampling follows standard practice in discrete diffusion and flow-based models (Wang et al., 2024) and provides a tractable approximation to intermediate Schrödinger bridge marginals.

Learning the Control Field Given corrupted sequences from Eq. equation 5, we learn a control field that tilts a reference process toward the data distribution along minimal-action paths. Under the exponential tilt parameterization, transition rates decompose as $\log R_u(x, x') = \log R_0(x, x') + u_\theta(x, x')$. While a uniform random-walk reference R_0 is analytically convenient, it fails to capture peptide biophysics and causes trajectories to drift rapidly into low-probability regions.

We therefore define R_0 using a biologically informed prior derived from a pre-trained encoder-only protein language model. Specifically, we use logits $f_\phi(x_t) \in \mathbb{R}^{L\times\mathcal{V}}$ from the frozen ESM-2-650M masked language model (Lin et al., 2023) as reference transition scores. Although ESM-2 is not generative (Appendix A.3.2), its logits provide a local measure of token plausibility suitable for defining reference dynamics. The resulting learned transition distribution decomposes to

$$\log p_\theta(\cdot \mid x_t, t) \propto \underbrace{u_\theta(x_t, \cdot, t)}_{\text{Learned tilt}} + \underbrace{(1 - t)\, f_\phi(x_t)}_{\text{Reference } R_0}, \tag{6}$$

where the factor $(1 - t)$ downweights the reference prior under heavy masking ($t \to 1$), where the language model $f_\phi(\cdot)$ is outside its training distribution.

With Eq. equation 6, training reduces to maximizing the transition intensity toward the target sequence, equivalent to minimizing the cross-entropy loss on corrupted tokens,

$$\mathcal{L}(\theta) = -\sum_{i=1}^{L} \mathbf{1}_{\{x_t^{(i)} \neq x_1^{(i)}\}} \log p_\theta(x_1^{(i)} \mid x_t, t). \tag{7}$$

Analysis of the Training Objective Although simple, this objective yields a consistent approximation to minimal-action control, formalized in the following proposition.

Proposition 4.3 (Consistency with Minimal-Action Control)**.** *Minimizing the cross-entropy loss in Eq. equation 7 is equivalent to minimizing the KL divergence between the model transition kernel and the optimal forward transitions of the Schrödinger bridge. Consequently, the learned control field u_θ converges to the unique minimal-action velocity field transporting μ_0 to μ_1 under reference dynamics R_0.*

The proof and additional discussion are provided in Appendix A.6.

Implementation Details We parameterize the control field u_θ using a Diffusion Transformer (DiT) architecture (Peebles & Xie, 2023), enabling time conditioning over ESM-2 latent embeddings. We use a 50M-parameter DiT with a frozen 650M-parameter ESM-2 model to define reference rates, yielding a total inference footprint of approximately 700M parameters while backpropagating gradients only through the 50M parameter control field. Full architectural details, datasets, and training algorithms are provided in Appendices A.3.3, A.3.1, and Algorithm 1.

4.5 Generative Sampling

After learning the control field u_θ, sampling begins from the fully masked prior $\mu_0 := \{[\texttt{MASK}]\}_{i=1}^{L}$. We evolve the sequence backward in time from $t = 1 \to 0$ by simulating a CTMC with the learned generator R_θ. In practice, we discretize time into N steps of size $\Delta t = 1/N$. At each discrete time $t_k = k/N$, we compute the total exit rate from the current sequence x_t via an exponential tilt of the control field, $R_{\text{tot}}(x_t) = \sum_{v \in \mathcal{V}} \exp\left(\lambda \cdot u_\theta(x_t, v, t_k)\right)$, where $\lambda = \|u_\theta\|_2 \cdot \gamma \cdot L$ and $\gamma = 0.005$ as a static rate scale. Next, transitions between CTMC states are governed by a Poisson jump process, a standard result in CTMC theory. The probability that a token updates within the interval Δt is thus $p_{\text{jump}} = 1 - \exp\left(-R_{\text{tot}}(x_t) \cdot \beta \cdot \Delta t\right)$ with $\beta = 0.05$ as a jump scale. Conditioned on a jump occurring, new tokens are sampled from a multinomial distribution ($n = 1$) induced by the control field. Specifically, we define the token distribution produced by the model as $p_\theta(v \mid x_t, t_k) = \text{Softmax}\left(\text{Top-}p\left(\frac{u_\theta(x_t,\cdot,t_k)+(1-t)f_\phi(x_t)}{\tau}\right)\right)$ where $\tau = 0.5$ and nucleus sampling uses $p = 0.9$. Then for each position, we sample a Bernoulli mask $\mathbf{z} \sim \text{Bern}(p_{\text{jump}})$ and update the sequence with new tokens x_{new} via $x_{t+\Delta t} = \mathbf{z} \odot x_{\text{new}} + (1 - \mathbf{z}) \odot x_t$, where $x_{\text{new}} \sim \textsc{Cat}(p_\theta(v \mid x_t, t_k))$.

Objective-Guided Sampling While unconditional peptide generation enables broad exploration of sequence space, such samples are unlikely to exhibit the binding properties required for therapeutic relevance. Prior work on guiding generative modeling has focused on continuous domains (Dhariwal & Nichol, 2021), with more recent extensions to discrete sequence spaces for biological sequence design (Nisonoff et al., 2024; Gruver et al., 2023; Vincoff et al., 2025a), considering multiple competing objectives (Chen et al., 2025c; Tang et al., 2025b), and optimization of specific sequence tokens (Goel et al., 2025).

As a case-study, we improve the binding affinity of MadSBM-generated peptides by leveraging Sampling-Importance-Resampling (SIR) within our rate-based Schrödinger bridge framework. Specifically, we guide the underlying Poisson process using a surrogate binding affinity classifier that quantifies the affinity of a peptide to the target protein. At each timestep, instead of sampling a single x_{new}, we draw $M = 16$ candidate sequences $\{x_{\text{new}}^{(m)}\}_{m=1}^{M}$ from the categorical $p_\theta(\cdot \mid x_t, t_k)$. Each candidate is then scored using the external classifier model. The resulting affinity scores $\{a^{(m)}\}_{m=1}^{M}$ are converted into selection weights via $w^{(m)} = \text{Softmax}\left(\frac{a^{(m)}}{\tau}\right)$, where $\tau = 0.5$. A single candidate is then sampled according to $w^{(m)}$ and used as x_{new} in the sequence update. Our use of the term "guidance" follows prior discrete generative work, where non-gradient, inference-time steering through MCTS (Tang et al., 2025b) or ranking-based filtering (Chen et al., 2025c) is commonly categorized as guidance. The full sampling procedure is shown in Algorithm 2 and we prove its convergence in Appendix A.1.8.

5 Results

5.1 Unconditional Sequence Generation Quality

Setup Unconditional peptide sequence generation broadens the space of potential therapeutic designs. To this end, we use MadSBM to sample 20 sequences for each sequence length

$L \in \{5, \ldots, 50\}$. For each generated sequence, we evaluate biological plausibility using the ProtBERT pseudo-perplexity (Elnaggar et al., 2021) (PPL; see Appendix A.3.2) and assess structural stability using the predicted local distance difference test (pLDDT) score (Jumper et al., 2021) of the folded structure produced by AlphaFold2 (AF2) (Jumper et al., 2021). We compare MadSBM against EvoFlow (https://huggingface.co/fredzzp/EvoFlow-150M) and the Diffusion Protein Language Model (DPLM) (Wang et al., 2024), two state-of-the-art discrete diffusion protein language models both built on the Reparameterized Diffusion Model framework (Appendix A.3.2) (Zheng et al., 2023). We elect to compare MadSBM against discrete diffusion models as they have recently become the *de facto* paradigm for biological sequence modeling (Wang et al., 2024), outperforming traditional autoregressive baselines (Nijkamp et al., 2023; Ferruz et al., 2022). Since both models were trained on a subset of UniProt sequences and peptides are biologically considered to be short proteins, we generate peptide samples by simply restricting sequence lengths to the range of 5–50 residues. Both EvoFlow and DPLM sample new peptide sequences using the Path-Planning scheme ("self-planner" variant) introduced by Peng et al. To provide both a stricter and fair comparison, we use the 150M-parameter EvoFlow model and 650M-parameter DPLM model when benchmarking against our 50M-parameter DiT-based MadSBM. Each model is evaluated at various sampling step budgets, $N \in \{32, 64, 128, 512\}$.

Results MadSBM generates peptide sequences with lower and competitive PPL and competitive structural confidence scores relative to both DD baselines across all sampling budgets, indicating higher biological alignment to the peptide sequence space (Table 1). All models outperform the held-out test set, which is a proxy for the natural peptide sequence distribution. Across the sampling budgets, MadSBM produces sequences with lower or competitive PPLs than both DD baselines.

Table 1: **Unconditional sequence generation quality across varying sampling step budgets (N).** The mean across all sequence lengths and standard deviation are reported.

Steps	Model	PPL (↓)	pLDDT (↑)	% Invalid (↓)
$N = 32$	ESM	4.189 ± 6.531	—	89.130
	EvoFlow	8.192 ± 8.861	83.597 ± 10.092	0.000
	DPLM	7.887 ± 8.336	77.497 ± 9.651	6.957
	MadSBM	$\mathbf{6.415 \pm 7.837}$	$\mathbf{83.007 \pm 9.424}$	**0.000**
$N = 64$	ESM	3.974 ± 5.871	—	88.261
	EvoFlow	7.676 ± 9.221	$\mathbf{85.856 \pm 8.823}$	0.000
	DPLM	7.171 ± 8.491	79.453 ± 9.891	9.130
	MadSBM	$\mathbf{6.387 \pm 7.403}$	82.307 ± 9.603	**0.000**
$N = 128$	ESM	4.085 ± 6.185	—	88.261
	EvoFlow	7.304 ± 9.270	$\mathbf{86.742 \pm 8.687}$	0.000
	DPLM	6.864 ± 9.850	82.120 ± 11.266	6.957
	MadSBM	$\mathbf{6.445 \pm 7.898}$	83.921 ± 9.434	**0.000**
$N = 512$	ESM	5.486 ± 8.112	—	84.783
	EvoFlow	6.484 ± 9.246	$\mathbf{87.450 \pm 8.022}$	0.000
	DPLM	$\mathbf{5.884 \pm 9.619}$	84.456 ± 12.036	16.522
	MadSBM	6.447 ± 8.375	81.564 ± 9.755	**0.000**
Test Set		8.244 ± 6.090	76.617 ± 10.493	—

Notably, the DD baselines produce sequences that fold into higher-confidence structures, as reflected by their higher average pLDDT scores. However, it is important to note that folding models used to compute metrics such as pLDDT rely on evolutionary information for predictions, which can obscure these models' ability to directly assess sequence–structure compatibility (Korbeld et al., 2025). Additionally, prior work has shown that diffusion models that reverse a predefined corruption process require sufficiently large sampling steps to maintain sample quality, as coarse discretizations that do not form a smooth transition from noise to data distributions result in large jumps in the token space (Xue et al., 2024; Shih et al., 2023). Notably, MadSBM outperforms the DD baselines even under low sampling budgets of $N = 32, 64, 128$, achieving lower PPL while maintaining competitive pLDDT scores. Further, DPLM is trained with ∼500 discretization steps; when evaluated under comparable budgets ($N = 512$), MadSBM remains competitive while maintaining strong performance in the low-step regime. Overall, these results indicate that MadSBM improves on current state-of-the-art DD models in unconditionally generating biologically relevant peptide sequences in a sample-efficient manner.

5.2 Navigating Probability Paths in the Discrete Sequence Space

Setup Discretized sampling processes in continuous and discrete generative models often force the evolving sequence through low-probability or biologically invalid regions of the sample space, potentially degrading final sample quality (Sahoo et al., 2024; Domingo-Enrich et al., 2024; Xue et al., 2024; Shih et al., 2023). Accordingly, we analyze the likelihood evolution of sequences generated by MadSBM and baseline models to assess adherence to the protein manifold during generation.

During the unconditional sampling procedure described in Section 4.5, we compute the negative log-likelihood (NLL) of the intermediate states x_t using the ProtBERT model at each timestep. This metric serves as a proxy for biological plausibility, as $e^{-\text{NLL}} \propto$ PPL. The NLL for all sequence lengths at the given sampling step were averaged.

Results We compare the NLL of different models across sampling iterations, using this metric as a proxy of the overall sequence likelihood across the generative trajectory. Figure A1 shows that the DD baseline follows a constrained and low-variance trajectory, while MadSBM exhibits greater path diversity and variance, reflecting its ability to explore a broader set of stochastic transport paths. Specifically, the DD baseline is confined to a narrow likelihood window throughout sampling, whereas MadSBM exhibits more variance of likelihoods, enabling it to reach higher-likelihood regions of the sequence manifold earlier in the generation process. Additionally, while MadSBM exhibits a lower likelihood floor during sampling compared to DD, which could result in exploration of a wider and potentially lower-quality region of the sequence space, this does not compromise generative quality as the model ultimately converges to superior or competitive sequence likelihoods (Table 1).

5.3 Ablation on Reference Dynamics

Setup Recall that MadSBM constructs its generative distribution by applying a learned tilt to the biologically-inspired reference process, which is defined through the logits produced by the ESM-2 model. To assess the benefit of using this biologically-relevant reference, we ablate the use of ESM-2 during training by 1) removing the time-gating mechanism and 2) replacing the ESM-derived reference with a random walk over sequence tokens modeled by a uniform generator (Eq. equation 1) that does not incorporate any biophysical peptide representations.

Results Table 2 shows that ablating the gating mechanism and ESM-2 itself leads to an increase in perplexity when evaluating the model on the held-out test set, indicating a degradation in generative quality when the biophysical reference dynamics is removed. Interestingly, fully ablating ESM-2 and the time-gating achieves a lower PPL, outperforming the variant that solely removes time-gating. While initially counterintuitive, this behavior is explained by the training distribution of ESM-2: the model is trained under a masked language modeling objective in which only 15% of tokens are replaced by the `[MASK]` token (Appendix A.3.2). As a result, ESM-2 representations become increasingly unreliable at higher masking rates, where the model operates out of distribution.

Table 2: **Ablation of reference process components in MadSBM**. MadSBM models were trained without time-gating and without ESM dependency and assess resulting perplexity on test set.

	$\log R_0$	Test PPL (↓)
MadSBM	$(1-t)f_\phi(x_t)$	**4.503**
w/o gating	$f_\phi(x_t)$	4.987
& w/o ESM-2	0	4.750

Removing the time-gating mechanism forces the model to rely on ESM logits even in these high-masking regimes, leading to noisier reference dynamics and poorer performance. In contrast, fully ablating ESM-2 eliminates this mismatch entirely, requiring the DiT backbone to jointly encode the peptide biophysical properties while learning the optimal tilt. These results overall validate our choice of using a principled, time-controlled reference process that appropriately modulates the influence of the biological prior across the corruption trajectory.

5.4 Minimal-Action Sampling Trajectories

Setup A central idea of MadSBM is that minimizing a simple cross-entropy objective corresponds to learning low-action transport paths between a simple reference process and the data distribution. To empirically evaluate this claim, we measure $\mathcal{A}_L(u^w)$, the *instantaneous actional* (the actional at each sampling step) as a proxy for the control cost incurred by our learned tilt. While sampling the same sequences used in the unconditional sequence generation and probability path evaluations, we computed the worst-case instantaneous actionals for various sampling budgets using the maximum logit value M. This worst-case scenario serves as the upper bound on the actionals that MadSBM should produce. We encourage the reader to review the derivation of the instantaneous actional in Appendix A.3.4.

Results Table 3 reports the maximum observed logits M and the corresponding worst-case actions $\mathcal{A}_L(u^w)$, and Figure A2 records the instantaneous actionals produced by MadSBM and its ablated counterpart models. Figure A2 shows that the instantaneous actionals incurred by MadSBM remain comfortably below the corresponding worst-case actionals in Table 3 throughout the sampling

trajectory. This behavior is consistent with our theoretical framing of MadSBM learning minimal-action transport paths: although biologically meaningful, the reference process can be noisy, requiring greater action from the learned control field to tilt toward the data distribution at each timestep.

In contrast, the ESM-ablated model exhibits near-constant actionals across sampling steps, which is also expected given the control field must not learn to adjust distorted peptide sequence representations from the reference process. Interestingly, we observe that MadSBM actionals increase as the sampling process converges. This reflects the fact that later generative steps must commit high-confidence tokens to the sequence, which further aligns with the low NLL variance at later sampling steps in Figure. A1.

Table 3: **Instantaneous worst-case actionals for different sampling budget discretizations** (Δt). Maximum observed logit M and corresponding instantaneous actional evaluated for each model.

Δt	Model	M	$\mathcal{A}_L(u^w)$
$1/32$	MadSBM	15.308787	1.991×10^6
	w/o ESM-2	14.275414	6.571×10^5
$1/64$	MadSBM	15.308787	9.953×10^5
	w/o ESM-2	14.275414	3.286×10^5
$1/128$	MadSBM	15.308787	4.976×10^5
	w/o ESM-2	14.275414	1.643×10^5
$1/512$	MadSBM	15.308787	1.244×10^5
	w/o ESM-2	14.275414	4.107×10^4

5.5 Binding Affinity Optimization

Setup We evaluate MadSBM's ability to design valid peptides by generating candidate binders for four disease targets with known binders and two targets with no known binders. For each target, we generate 60 peptide sequences at various sequence lengths by concatenating an L-length sequence of `[MASK]` tokens to the target amino acid sequence and running the MadSBM sampling procedure. To enhance binding affinity, we subsequently resample 60 peptides per target using objective-guided sampling (Section 4.5), where the guidance signal is provided by a pre-trained binding affinity predictor. Specifically, we use the unpooled wild-type to wild-type binding affinity model from PeptiVerse (Zhang et al., 2026). In the unconditional and guidance cases, we use $N = 32$ sampling steps to highlight MadSBM's sample step efficiency and sequence lengths $L \in \{10, 15, 20\}$ to align with the length distribution of experimentally characterized peptide binders.

Results Table 4 shows the binding affinities, ipTM scores, and AutoDock VINA scores for four targets with existing binderse and for a target without an existing binder.

Table 4: **Binding affinity scores for designed & existing binders.** Existing binder affinities and sequences taken from (Tang et al., 2025c). ipTM values sourced from AlphaFold2 and docking scores sourced from AutoDock VINA. Values reported as average across 10, 15, and 20-length peptides. Targets 5E1C, 4EZN, 1AYC, and 5KRI have existing binders. 3HVE and 6A9P have no existing binders.

	Binding Affinity (↑)			**Best ipTM** (↑)			**Docking Score** (kcal/mol) (↓)		
Target	Existing	Unconditional	Guided	Existing	Unconditional	Guided	Existing	Unconditional	Guided
5E1C	4.932	5.416	6.109	0.83	0.04	0.75	-4.3	-6.0	-6.3
4EZN	6.176	5.468	6.072	0.53	0.28	0.68	-4.1	-6.0	-6.8
1AYC	6.576	7.272	7.982	0.58	0.34	0.57	-5.3	-6.8	-7.5
5KRI	4.932	5.416	6.109	0.83	0.05	0.76	-3.5	-6.5	-6.7
3HVE	**	5.646	6.334	**	0.410	0.810	–	-6.3	-6.8
6A9P	**	5.288	5.917	**	0.100	0.190	–	-4.2	-4.3

Notably, for three of the four targets with binders, MadSBM-generated peptides *without guidance* have greater binding affinities and docking scores than existing peptides, indicating that MadSBM learned a therapeutically potent peptide distribution. Applying guidance further improves upon binding affinities and docking scores, demonstrating, to our knowledge, the first-ever application of discrete classifier guidance (Gruver et al., 2023; Nisonoff et al., 2024; Goel et al., 2025; Tang et al., 2025b; Vincoff et al., 2025a) to a Schrödinger bridge matching-based generative model. We further highlight that MadSBM was constrained to a low sampling budget relative to what's required by state-of-the-art discrete diffusion models, and that while existing binders are often manually curated through slow experimental screening, MadSBM rapidly produces high-quality candidates.

DISCUSSION

We introduce **MadSBM**, which formulates peptide sequence generation as a rate-based stochastic transport problem and shows that low-action Schrödinger bridge dynamics can be learned directly in discrete sequence spaces using an informative biological reference process. By defining generation relative to pre-trained protein language model logits and learning a time-dependent control field, MadSBM maintains probability trajectories near high-likelihood peptide regions throughout sampling, achieving improved perplexity and competitive structural confidence at substantially lower sampling budgets than diffusion- and flow-based baselines. The method inherits limitations from its reference dynamics, including dependence on pre-trained language models that may be poorly calibrated under extreme masking and sensitivity to time-gating and rate scaling choices. Future work will focus on experimental validation, developing more chemistry- and structure-aware reference processes, scaling to larger protein domains, incorporating higher-fidelity physical or experimental feedback, and extending the framework to other sequence domains such as nucleic acids and natural language. More broadly, MadSBM highlights the potential of discrete generative models that explicitly model and control the full probability trajectory rather than only the final distribution.

REFERENCES

Osama Abdin, Satra Nim, Han Wen, and Philip M Kim. Pepnn: a deep attention model for the identification of peptide binding sites. *Communications biology*, 5(1):503, 2022.

Anton Bushuiev, Roman Bushuiev, Petr Kouba, Anatolii Filkin, Marketa Gabrielova, Michal Gabriel, Jiri Sedlar, Tomas Pluskal, Jiri Damborsky, Stanislav Mazurenko, et al. Learning to design protein-protein interactions with enhanced generalization. *arXiv preprint arXiv:2310.18515*, 2023.

Leo Tianlai Chen, Zachary Quinn, Madeleine Dumas, Christina Peng, Lauren Hong, Moises Lopez-Gonzalez, Alexander Mestre, Rio Watson, Sophia Vincoff, Lin Zhao, et al. Target sequence-conditioned design of peptide binders using masked language modeling. *Nature Biotechnology*, pp. 1–9, 2025a.

Tong Chen, Zachary Quinn, Yinuo Zhang, and Pranam Chatterjee. moPPIt-v3: Motif-specific peptides generated via multi-objective-guided discrete flow matching. In *2nd edition of Frontiers in Probabilistic Inference: Learning meets Sampling*, 2025b. URL https://openreview.net/forum?id=8wr2Krx1Fm.

Tong Chen, Yinuo Zhang, Sophia Tang, and Pranam Chatterjee. Multi-objective-guided discrete flow matching for controllable biological sequence design. In *ICML 2025 Generative AI and Biology (GenBio) Workshop*, 2025c. URL https://openreview.net/forum?id=8YIMLoHP9J.

Yongxin Chen, Tryphon T Georgiou, and Michele Pavon. Optimal transport in systems and control. *Annual Review of Control, Robotics, and Autonomous Systems*, 4(1):89–113, 2021.

Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. Diffusion schrödinger bridge with applications to score-based generative modeling. *Advances in neural information processing systems*, 34:17695–17709, 2021.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Carles Domingo-Enrich, Michal Drozdzal, Brian Karrer, and Ricky TQ Chen. Adjoint matching: Fine-tuning flow and diffusion generative models with memoryless stochastic optimal control. *arXiv preprint arXiv:2409.08861*, 2024.

Carles Domingo i Enrich, Jiequn Han, Brandon Amos, Joan Bruna, and Ricky TQ Chen. Stochastic optimal control matching. *Advances in Neural Information Processing Systems*, 37:112459–112504, 2024.

Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, et al. Prottrans: toward understanding the language of life through self-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):7112–7127, 2021.

Noelia Ferruz, Steffen Schmidt, and Birte Höcker. Protgpt2 is a deep unsupervised language model for protein design. *Nature communications*, 13(1):4348, 2022.

Aude Genevay, Gabriel Peyré, and Marco Cuturi. Learning generative models with sinkhorn divergences. In *International Conference on Artificial Intelligence and Statistics*, pp. 1608–1617. PMLR, 2018.

Shrey Goel, Peregrine Michael Schray, Yinuo Zhang, Sophia Vincoff, Huong T. Kratochvil, and Pranam Chatterjee. Token-level guided discrete diffusion for membrane protein design. In *NeurIPS AI4Science Workshop*, 2025. URL https://openreview.net/forum?id=I0hzddNny7.

Nate Gruver, Samuel Stanton, Nathan Frey, Tim GJ Rudner, Isidro Hotzel, Julien Lafrance-Vanasse, Arvind Rajpal, Kyunghyun Cho, and Andrew G Wilson. Protein design with guided discrete diffusion. *Advances in neural information processing systems*, 36:12489–12517, 2023.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Lauren Hong, Tianzheng Ye, Tian Z Wang, Divya Srijay, Howard Liu, Lin Zhao, Rio Watson, Sophia Vincoff, Tianlai Chen, Kseniia Kholina, et al. Programmable protein stabilization with language model-derived peptide guides. *Nature Communications*, 16(1):3555, 2025.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

Jun Hyeong Kim, Seonghwan Kim, Seokhyun Moon, Hyeongwoo Kim, Jeheon Woo, and Woo Youn Kim. Discrete diffusion schr\" odinger bridge matching for graph transformation. *arXiv preprint arXiv:2410.01500*, 2024.

Kerlen T Korbeld, Vsevolod Viliuga, and Maximilian Fürst. Limitations of the refolding pipeline for de novo protein design. *bioRxiv*, pp. 2025–12, 2025.

Grigoriy Ksenofontov and Alexander Korotin. Categorical schr\" odinger bridge matching. *arXiv preprint arXiv:2502.01416*, 2025.

Christian Léonard. A survey of the schr\" odinger problem and some of its connections with optimal transport. *arXiv preprint arXiv:1308.0215*, 2013.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Erik Nijkamp, Jeffrey A Ruffolo, Eli N Weinstein, Nikhil Naik, and Ali Madani. Progen2: exploring the boundaries of protein language models. *Cell systems*, 14(11):968–978, 2023.

Hunter Nisonoff, Junhao Xiong, Stephan Allenspach, and Jennifer Listgarten. Unlocking guidance for discrete state-space diffusion and flow models. *arXiv preprint arXiv:2406.01572*, 2024.

Martin Pacesa, Lennart Nickel, Christian Schellhaas, Joseph Schmidt, Ekaterina Pyatova, Lucas Kissling, Patrick Barendse, Jagrity Choudhury, Srajan Kapoor, Ana Alcaraz-Serna, et al. One-shot design of functional protein binders with bindcraft. *Nature*, pp. 1–10, 2025.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

Fred Zhangzhi Peng, Zachary Bezemek, Sawan Patel, Jarrid Rector-Brooks, Sherwood Yao, Avishek Joey Bose, Alexander Tong, and Pranam Chatterjee. Path planning for masked diffusion model sampling. *arXiv preprint arXiv:2502.03540*, 2025.

Peter Potaptchik, Cheuk-Kit Lee, and Michael S Albergo. Tilt matching for scalable sampling and fine-tuning. *arXiv preprint arXiv:2512.21829*, 2025.

Subham Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin Chiu, Alexander Rush, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems*, 37:130136–130184, 2024.

Erwin Schrödinger. Sur la théorie relativiste de l'électron et l'interprétation de la mécanique quantique. In *Annales de l'institut Henri Poincaré*, volume 2, pp. 269–310, 1932.

Yuyang Shi, Valentin De Bortoli, Andrew Campbell, and Arnaud Doucet. Diffusion schrödinger bridge matching. *Advances in Neural Information Processing Systems*, 36:62183–62223, 2023.

Andy Shih, Suneel Belkhale, Stefano Ermon, Dorsa Sadigh, and Nima Anari. Parallel sampling of diffusion models. *Advances in Neural Information Processing Systems*, 36:4263–4276, 2023.

Kirill Sokolov and Alexander Korotin. Exponential convergence rate for iterative markovian fitting. *arXiv preprint arXiv:2508.02770*, 2025.

Hannes Stark, Bowen Jing, Chenyu Wang, Gabriele Corso, Bonnie Berger, Regina Barzilay, and Tommi Jaakkola. Dirichlet flow matching with applications to dna sequence design. In *Forty-first International Conference on Machine Learning*, 2024.

Martin Steinegger and Johannes Söding. Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature biotechnology*, 35(11):1026–1028, 2017.

Sophia Tang, Yinuo Zhang, and Pranam Chatterjee. Entangled schrödinger bridge matching. *ArXiv*, pp. arXiv–2511, 2025a.

Sophia Tang, Yinuo Zhang, and Pranam Chatterjee. Peptune: De novo generation of therapeutic peptides with multi-objective-guided discrete diffusion. In *Forty-second International Conference on Machine Learning*, 2025b. URL https://openreview.net/forum?id=FQoy1Y1Hd8.

Sophia Tang, Yinuo Zhang, Alexander Tong, and Pranam Chatterjee. Gumbel-softmax score and flow matching for discrete biological sequence generation. In *ICLR 2025 Workshop on AI for Nucleic Acids*, 2025c. URL https://openreview.net/forum?id=ITpCmDhSfu.

Sophia Tang, Yuchen Zhu, Molei Tao, and Pranam Chatterjee. Tr2-d2: Tree search guided trajectory-aware fine-tuning for discrete diffusion. *arXiv preprint arXiv:2509.25171*, 2025d.

Alexander Tong, Nikolay Malkin, Kilian Fatras, Lazar Atanackovic, Yanlei Zhang, Guillaume Huguet, Guy Wolf, and Yoshua Bengio. Simulation-free schr\" odinger bridges via score and flow matching. *arXiv preprint arXiv:2307.03672*, 2023.

Francisco Vargas, Pierre Thodoroff, Austen Lamacraft, and Neil Lawrence. Solving schrödinger bridges via maximum likelihood. *Entropy*, 23(9):1134, 2021.

Sophia Vincoff, Oscar Davis, Ismail Ilkan Ceylan, Alexander Tong, Joey Bose, and Pranam Chatterjee. SOAPIA: Siamese-guided generation of off target-avoiding protein interactions with high target affinity. In *ICML 2025 Workshop on Scaling Up Intervention Models*, 2025a. URL https://openreview.net/forum?id=j0OpIG7leX.

Sophia Vincoff, Shrey Goel, Kseniia Kholina, Rishab Pulugurta, Pranay Vure, and Pranam Chatterjee. Fuson-plm: a fusion oncoprotein-specific language model via adjusted rate masking. *Nature Communications*, 16(1):1436, 2025b.

Xinyou Wang, Zaixiang Zheng, Fei Ye, Dongyu Xue, Shujian Huang, and Quanquan Gu. Diffusion language models are versatile protein learners. *arXiv preprint arXiv:2402.18567*, 2024.

Shuchen Xue, Zhaoqiang Liu, Fei Chen, Shifeng Zhang, Tianyang Hu, Enze Xie, and Zhenguo Li. Accelerating diffusion sampling with optimized time steps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8292–8301, 2024.

Chengxin Zhang, Xi Zhang, Lydia Freddolino, and Yang Zhang. Biolip2: an updated structure database for biologically relevant ligand–protein interactions. *Nucleic Acids Research*, 52(D1): D404–D412, 2024.

Yinuo Zhang, Divya Srijay, and Pranam Chatterjee. Metalorian: De novo generation of heavy metal-binding peptides with classifier-guided diffusion sampling. In *ICLR 2025 Workshop on Generative and Experimental Perspectives for Biomolecular Design*, 2025a. URL https://openreview.net/forum?id=CVokYznJ1i.

Yinuo Zhang, Sophia Tang, and Pranam Chatterjee. ScooBDoob: Schrödinger bridge with doob's \textit{h}-transform for molecular dynamics. In *EurIPS 2025 Workshop on SIMBIOCHEM*, 2025b. URL https://openreview.net/forum?id=0BVnuuyKW8.

Yinuo Zhang, Sophia Tang, Tong Chen, Elizabeth Mahood, Sophia Vincoff, and Pranam Chatterjee. Peptiverse: A unified platform for therapeutic peptide property prediction. *bioRxiv*, pp. 2025–12, 2026.

Lin Zheng, Jianbo Yuan, Lei Yu, and Lingpeng Kong. A reparameterized discrete diffusion model for text generation. *arXiv preprint arXiv:2302.05737*, 2023.

A APPENDIX

You may include other additional sections here.

A.1 THEORETICAL PROOFS

In this section we provide theoretical guarantees for Minimal-Action Schrödinger Bridges (MadSBM). We first derive a path-space relative entropy identity for controlled continuous-time Markov chains (CTMCs) on discrete sequence spaces. We then characterize the discrete Schrödinger bridge as an endpoint-tilted reference path measure and show that its optimal generator is a Doob h-transform of the reference generator. This yields a discrete Hamilton–Jacobi–Bellman (HJB) equation for the optimal log-potential and an explicit expression for the optimal edge control. Finally, we analyze the cross-entropy objective, provide stability bounds that relate control error to path-space divergence, establish representational completeness of the exponential-tilt parameterization, and prove convergence of the time-discretized sampler.

Notation We adopt the notation from the main text. Let $\mathcal{V}$ denote the finite alphabet and $\mathcal{X} := \mathcal{V}^L$ as the length-L sequence space. We set μ_0 as the simple prior on $\mathcal{X}$ and μ_1 as the data distribution on $\mathcal{X}$. For our dynamics, let R_0 be the reference generator on $\mathcal{X}$ with $R_0(x, x') \geq 0$ for $x' \neq x$ and $R_0(x, x) = -\sum_{y \neq x} R_0(x, y)$, let R_t be the time-dependent controlled generator with induced path law on $[0, T]$ denoted by $\mathbb{P}$, and let the reference path law induced by (μ_0, R_0) be denoted by $\mathbb{P}_0$. We define the multiplicative control parameterization as $R_t(x, x') = R_0(x, x')\, a_t(x, x')$ with $a_t(x, x') > 0$ for $x' \neq x$ and sometimes use $a_t(x, x') = \exp(u_t(x, x'))$ with log-control $u_t(x, x') \in \mathbb{R}$. The Schrödinger bridge path law between (μ_0, μ_1) relative to $\mathbb{P}_0$ is denoted as $\mathbb{P}^\star$, with generator $R_t^\star$.

A.1.1 PATH-SPACE RELATIVE ENTROPY FOR CONTROLLED CTMCS

We derive the standard relative-entropy identity for CTMC path measures. This result is the discrete analogue of Girsanov-type identities for diffusion processes.

Theorem A.1 (Path-space KL decomposition for CTMCs)**.** *Let $\mathbb{P}$ and $\mathbb{P}_0$ be path measures on $D([0, T]; \mathcal{X})$ induced by time-inhomogeneous CTMCs with generators R_t and R_0 and initial distributions ν and ν_0, respectively. Assume absolute continuity in the sense that for all $t \in [0, T]$ and $x \neq x'$,*

$$R_t(x, x') > 0 \;\Rightarrow\; R_0(x, x') > 0. \tag{8}$$

Then $\mathbb{P} \ll \mathbb{P}_0$ and the path-space relative entropy satisfies

$$\mathrm{KL}(\mathbb{P}_u \,\|\, \mathbb{P}_0) = \mathbb{E}_{\mathbb{P}_u}\left[\int_0^T \sum_{x' \neq X_t} \Big(R_u(X_t, x') \log \tfrac{R_u(X_t, x')}{R_0(X_t, x')} - R_u(X_t, x') + R_0(X_t, x') \Big)\, dt\right]. \tag{9}$$

Proof. A sample path of a CTMC on a finite state space can be described by an initial state $X_0 = x_0$ having N jumps with jump times $0 < \tau_1 < \cdots < \tau_N \leq T$, and post-jump states $x_1, \ldots, x_N$, where $x_i \neq x_{i-1}$ for each i and $X_t = x_i$ for $t \in [\tau_i, \tau_{i+1})$ with $\tau_0 = 0$ and $\tau_{N+1} = T$. Let the exit rates be defined as

$$\lambda_t(x) := \sum_{y \neq x} R_t(x, y), \qquad \lambda_0(x) := \sum_{y \neq x} R_0(x, y). \tag{10}$$

We now compute the likelihood of a realized path under the MadSBM model specified by (ν, R_t). The probability of starting in state x_0 is given by

$$\mathbb{P}(X_0 = x_0) = \nu(x_0). \tag{11}$$

While the process remains in state x_i, no jump occurs on the interval $[\tau_i, \tau_{i+1})$. For a time-inhomogeneous CTMC, the corresponding survival probability is

$$\exp\left(-\int_{\tau_i}^{\tau_{i+1}} \lambda_t(x_i)\, dt\right). \tag{12}$$

Multiplying over all inter-jump intervals yields the total survival contribution

$$\prod_{i=0}^{N} \exp\left(-\int_{\tau_i}^{\tau_{i+1}} \lambda_t(x_i)\, dt\right) = \exp\left(-\sum_{i=0}^{N} \int_{\tau_i}^{\tau_{i+1}} \lambda_t(x_i)\, dt\right). \tag{13}$$

At each jump time τ_i, the instantaneous probability density of transitioning from x_{i-1} to x_i is given by the transition rate $R_{\tau_i}(x_{i-1}, x_i)$. The contribution from all jumps is therefore given by the product

$$\prod_{i=1}^{N} R_{\tau_i}(x_{i-1}, x_i). \tag{14}$$

Combining the initial distribution, survival probabilities, and jump intensities, the likelihood of the path under $\mathbb{P}$ is

$$p_{\mathbb{P}} = \nu(x_0) \left(\prod_{i=1}^{N} R_{\tau_i}(x_{i-1}, x_i) \right) \exp\left(-\sum_{i=0}^{N} \int_{\tau_i}^{\tau_{i+1}} \lambda_t(x_i)\, dt \right). \tag{15}$$

A completely analogous expression holds for $p_{\mathbb{P}_0}$ with (ν_0, R_0).

Since we aim to find the KL divergence, we will take the ratio and the logarithm. Using the support assumption yields

$$\log \frac{p_{\mathbb{P}}}{p_{\mathbb{P}_0}} = \log \frac{\nu(X_0)}{\nu_0(X_0)} + \sum_{i=1}^{N} \log \frac{R_{\tau_i}(X_{\tau_i^-}, X_{\tau_i})}{R_0(X_{\tau_i^-}, X_{\tau_i})} - \int_0^T (\lambda_t(X_t) - \lambda_0(X_t))\, dt. \tag{16}$$

Taking expectation with respect to $\mathbb{P}$ yields

$$\begin{aligned} \mathrm{KL}(\mathbb{P} \,\|\, \mathbb{P}_0) &= \mathbb{E}_{\mathbb{P}} \left[\log \frac{p_{\mathbb{P}}}{p_{\mathbb{P}_0}} \right] \\ &= \mathbb{E}_{\mathbb{P}} \left[\log\left(\frac{v(X_0)}{v_0(X_0}\right) \right] + \mathbb{E}_{\mathbb{P}} \left[\sum_{x' \neq X_t} \log \frac{R_t(X_t, x')}{R_0(X_t, x')}\, dt \right] - \mathbb{E}_{\mathbb{P}} \left[\int_0^T (\lambda_t(X_t) - \lambda_0(X_t))\, dt \right]. \end{aligned} \tag{17}$$

Now we simplify. The first term yields $\mathrm{KL}(\nu \| \nu_0)$ by definition and vanishes to zero. For the second term (sum over jumps), we use the Martingale compensator identity for CTMCs, which states for any bounded measurable function $f(t, x, x')$:

$$\mathbb{E}_{\mathbb{P}} \left[\sum_{i=1}^{N} f(\tau_i, X_{\tau_i^-}, X_{\tau_i}) \right] = \mathbb{E}_{\mathbb{P}} \left[\int_0^T \sum_{x' \neq X_t} R_t(X_t, x') f(t, X_t, x')\, dt \right]. \tag{18}$$

Applying this with $f(t, x, x') := \log \frac{R_t(x,x')}{R_0(x,x')}$ transforms the sum over jumps into an integral over time, giving:

$$\mathbb{E}_{\mathbb{P}} \left[\int_0^T \sum_{x' \neq x X_t} R_t(X_t, x') \log \frac{R_t(X_t, x')}{R_0(X_t, x')} dt \right] \tag{19}$$

For the third and final term, note that

$$\lambda_t(X_t) - \lambda_0(X_t) = \sum_{x' \neq X_t} (R_t(X_t, x') - R_0(X_t, x')) \tag{20}$$

Substituting these identities into equation 16 yields equation 9. □

Corollary A.2 (Action form under exponential tilting)**.** *Assume $\nu = \nu_0 = \mu_0$ and $R_t(x, x') = R_0(x, x') \exp(u_t(x, x'))$ for $x' \neq x$. Then*

$$\mathrm{KL}(\mathbb{P} \,\|\, \mathbb{P}_0) = \mathbb{E}_{\mathbb{P}} \left[\int_0^T \sum_{x' \neq X_t} R_0(X_t, x') \left(e^{u_t(X_t, x')} u_t(X_t, x') - e^{u_t(X_t, x')} + 1 \right) dt \right]. \tag{21}$$

Proof. Apply Theorem A.1 with $R_t = R_0 e^{u_t}$ and cost function $\Psi(z) = e^z - z - 1$. For each edge,

$$\begin{aligned} R_t \log \frac{R_t}{R_0} - R_t + R_0 &= R_t \left(\log \frac{R_t}{R_0} \right) - R_t + R_0 \\ &= R_t \cdot u - R_t + R_0 && (\text{since } \log \frac{R_t}{R_0} = u) \\ &= (R_0 e^u)\, u - R_0 e^u + R_0 && (\text{since } R_t = R_0 e^u) \\ &= R_0 \big(e^u u - e^u + 1 \big). \end{aligned} \tag{22}$$

and summing over $x' \neq X_t$ yields Eq. equation 21. □

A.1.2 Discrete Schrödinger Bridge Structure

We now characterize the Schrödinger bridge as the KL projection of the reference path law onto the set of path laws with prescribed endpoint marginals.

Theorem A.3 (Endpoint-tilted form and uniqueness)**.** *Let $\mathcal{C}$ denote the set of path measures on $D([0,T];\mathcal{X})$ whose endpoint marginals satisfy $X_0 \sim \mu_0$ and $X_T \sim \mu_1$. Assume $\mathcal{C}$ is nonempty and that there exists at least one $\mathbb{Q} \in \mathcal{C}$ with $\mathrm{KL}(\mathbb{Q}\|\mathbb{P}_0) < \infty$. Then the optimization problem*

$$\mathbb{P}^\star \in \arg\min_{\mathbb{Q}\in\mathcal{C}} \mathrm{KL}(\mathbb{Q} \,\|\, \mathbb{P}_0) \tag{23}$$

admits a unique minimizer $\mathbb{P}^\star$. Moreover, there exist functions $f, g : \mathcal{X} \to (0,\infty)$ such that

$$\frac{d\mathbb{P}^\star}{d\mathbb{P}_0}(X) = \frac{f(X_0)\, g(X_T)}{\mathbb{E}_{\mathbb{P}_0}[f(X_0) g(X_T)]}. \tag{24}$$

Proof. **Uniqueness.** The feasible set $\mathcal{C}$ is convex because endpoint marginal constraints are linear in $\mathbb{Q}$. The functional $\mathbb{Q} \mapsto \mathrm{KL}(\mathbb{Q}\|\mathbb{P}_0)$ is strictly convex on $\{\mathbb{Q} : \mathbb{Q} \ll \mathbb{P}_0\}$, hence the minimizer is unique.

Factor form. Consider the constrained minimization of $\mathrm{KL}(\mathbb{Q}\|\mathbb{P}_0)$ over $\mathbb{Q} \ll \mathbb{P}_0$ with endpoint constraints. Introduce Lagrange multipliers $\alpha, \beta : \mathcal{X} \to \mathbb{R}$ for the constraints $\mathbb{Q}(X_0 = x) = \mu_0(x)$ and $\mathbb{Q}(X_T = x) = \mu_1(x)$. The Lagrangian is

$$\mathcal{L}(\mathbb{Q}, \alpha, \beta) = \mathrm{KL}(\mathbb{Q}\|\mathbb{P}_0) + \sum_{x\in\mathcal{X}} \alpha(x) \left(\mu_0(x) - \mathbb{Q}(X_0 = x) \right) + \sum_{x\in\mathcal{X}} \beta(x) \left(\mu_1(x) - \mathbb{Q}(X_T = x) \right). \tag{25}$$

A standard variational argument on densities $d\mathbb{Q}/d\mathbb{P}_0$ yields the stationarity condition

$$\log \frac{d\mathbb{Q}}{d\mathbb{P}_0}(X) = \alpha(X_0) + \beta(X_T) - c \tag{26}$$

for some constant c. Exponentiating gives

$$\frac{d\mathbb{Q}}{d\mathbb{P}_0}(X) \propto e^{\alpha(X_0)} e^{\beta(X_T)}. \tag{27}$$

Setting $f(x) := e^{\alpha(x)}$ and $g(x) := e^{\beta(x)}$ yields equation 24 after normalization. The functions f, g are then selected to satisfy the endpoint constraints, which is possible by assumption that $\mathcal{C}$ is nonempty and a finite-KL feasible point exists. □

A.1.3 Doob h-Transform and Optimal Generator

The endpoint tilt implies that $\mathbb{P}^\star$ is Markov and admits an explicit generator as a Doob h-transform of R_0.

Theorem A.4 (Doob transform form of the Schrödinger bridge)**.** *Let $\mathbb{P}^\star$ be as in Theorem A.3. Define the backward potential*

$$h_t(x) := \mathbb{E}_{\mathbb{P}_0}[g(X_T) \,|\, X_t = x], \qquad t \in [0,T], \tag{28}$$

where g is the endpoint factor from equation 24. Then $h_t(x) > 0$ and the optimal bridge $\mathbb{P}^\star$ is a time-inhomogeneous Markov process with generator

$$R_t^\star(x, x') = R_0(x, x') \frac{h_t(x')}{h_t(x)}, \qquad x' \neq x, \tag{29}$$

and diagonal entries $R_t^\star(x,x) = -\sum_{y\neq x} R_t^\star(x,y)$. *Equivalently, in log-control form,*

$$u_t^\star(x,x') = \log h_t(x') - \log h_t(x). \tag{30}$$

Proof. Fix $t \in [0,T)$ and states $x \neq x'$. Under $\mathbb{P}^\star$, the conditional law of the future given the present is obtained by tilting the corresponding conditional law under $\mathbb{P}_0$ by $g(X_T)$. More precisely, for any event A measurable with respect to the future σ-algebra generated by $(X_s)_{s\in[t,T]}$,

$$\mathbb{P}^\star(A \mid X_t = x) = \frac{\mathbb{E}_{\mathbb{P}_0}[\mathbf{1}_A g(X_T) \mid X_t = x]}{\mathbb{E}_{\mathbb{P}_0}[g(X_T) \mid X_t = x]} = \frac{\mathbb{E}_{\mathbb{P}_0}[\mathbf{1}_A g(X_T) \mid X_t = x]}{h_t(x)}. \tag{31}$$

Take A to be the event that a jump from x to x' occurs in $[t, t+\Delta t]$ and no other jump occurs in that interval. Under $\mathbb{P}_0$, this event has probability

$$\mathbb{P}_0(X_{t+\Delta t} = x' \mid X_t = x) = R_0(x,x')\Delta t + o(\Delta t). \tag{32}$$

Moreover, conditioning on $X_{t+\Delta t} = x'$ and using the Markov property of $\mathbb{P}_0$ gives

$$\mathbb{E}_{\mathbb{P}_0}[g(X_T) \mid X_{t+\Delta t} = x'] = h_{t+\Delta t}(x') = h_t(x') + o(1) \tag{33}$$

as $\Delta t \to 0$, by right-continuity of $t \mapsto h_t(x')$ on finite state spaces. Substituting into equation 31 yields

$$\begin{aligned}\mathbb{P}^\star(X_{t+\Delta t} = x' \mid X_t = x) &= \frac{\mathbb{P}_0(X_{t+\Delta t} = x' \mid X_t = x)\, h_{t+\Delta t}(x')}{h_t(x)} + o(\Delta t) \\ &= \frac{R_0(x,x')h_t(x')}{h_t(x)}\Delta t + o(\Delta t).\end{aligned} \tag{34}$$

Therefore the jump intensity from x to x' under $\mathbb{P}^\star$ is $R_0(x,x')h_t(x')/h_t(x)$, which is equation 29. Taking logarithms gives equation 30. □

A.1.4 Discrete HJB Equation for the Optimal Log-Potential

The Doob potential h_t satisfies a linear backward equation under the reference dynamics. Its logarithm satisfies a nonlinear discrete HJB equation whose edge increments recover the optimal control.

Theorem A.5 (Discrete HJB for Schrödinger potentials)**.** *Let h_t be defined by equation 28 and assume $t \mapsto h_t(x)$ is differentiable for each $x \in \mathcal{X}$. Then h_t solves the backward Kolmogorov equation*

$$\partial_t h_t(x) + (R_0 h_t)(x) = 0, \qquad h_T(x) = g(x), \tag{35}$$

where $(R_0 h)(x) := \sum_{x'\neq x} R_0(x,x')(h(x') - h(x))$. Define the log-potential $V_t(x) := \log h_t(x)$. Then V_t satisfies the nonlinear equation

$$\partial_t V_t(x) + \sum_{x'\neq x} R_0(x,x')\left(\exp\big(V_t(x') - V_t(x)\big) - 1\right) = 0, \qquad V_T(x) = \log g(x). \tag{36}$$

Moreover, the optimal edge control satisfies

$$u_t^\star(x,x') = V_t(x') - V_t(x). \tag{37}$$

Proof. The backward equation equation 35 follows from the definition $h_t(x) = \mathbb{E}_{\mathbb{P}_0}[g(X_T) \mid X_t = x]$ and standard Markov semigroup arguments on finite state spaces.

For the nonlinear equation, differentiate $V_t(x) = \log h_t(x)$:

$$\partial_t V_t(x) = \frac{\partial_t h_t(x)}{h_t(x)} = -\frac{(R_0 h_t)(x)}{h_t(x)}. \tag{38}$$

Expand $(R_0 h_t)(x)$:

$$\frac{(R_0 h_t)(x)}{h_t(x)} = \sum_{x'\neq x} R_0(x,x')\left(\frac{h_t(x')}{h_t(x)} - 1\right) = \sum_{x'\neq x} R_0(x,x')\left(\exp\big(V_t(x') - V_t(x)\big) - 1\right), \tag{39}$$

which yields equation 36. The edge control identity follows from Theorem A.4 with $V_t = \log h_t$. □

A.1.5 CONSISTENCY OF TRAINING OBJECTIVE

Theorem A.6 (Cross Entropy Loss Consistency)**.** *Let $u_t^\star(x, x')$ denote the optimal log-control of the Schrödinger bridge, as characterized in Eq. equation 30. Consider the population training objective induced by the cross-entropy loss,*

$$\mathcal{L}(\theta) = \mathbb{E}_{t\sim\pi}\mathbb{E}_{X_t\sim\rho_t^\star}\left[\mathrm{KL}(p_t^\star(\cdot \mid X_t) \,\|\, p_\theta(\cdot \mid X_t, t))\right], \tag{40}$$

where π is any distribution on $[0, T]$ with full support, $\rho_t^\star$ is the marginal of the optimal bridge $\mathbb{P}^\star$ at time t, and

$$p_\theta(x' \mid x, t) \;\propto\; R_0(x, x')\,\exp\big(u_\theta(x, x', t)\big), \qquad p_t^\star(x' \mid x) \;\propto\; R_0(x, x')\,\exp\big(u_t^\star(x, x')\big).$$

If a minimizer θ^* exists and the model class $\{u_\theta\}$ is sufficiently expressive to represent $u^\star$, then

$$\mathcal{L}(\theta^*) = 0 \quad\Longrightarrow\quad u_{\theta^*}(x, x', t) = u_t^\star(x, x') + c(x, t), \tag{41}$$

for $\pi(dt)\rho_t^\star(dx)$-almost every (t, x) and for all neighbors x' with $R_0(x, x') > 0$, where $c(x, t)$ is an additive normalization constant independent of x'.

Proof. Let θ^* be a minimizer of the population training objective and assume the model class is rich enough so that there exists θ with $p_\theta = p^\star$ almost everywhere. By non-negativity of the KL divergence and the definition of the training objective,

$$\mathcal{L}(\theta^*) = 0 \quad\Longrightarrow\quad \mathrm{KL}\big(p_t^\star(\cdot \mid x) \,\|\, p_{\theta^*}(\cdot \mid x, t)\big) = 0$$

This admits the reverse transition kernel as:

$$p_{\theta^*}(x' \mid x, t) = p_t^\star(x' \mid x)$$

for all $R_0(x, x') > 0$. By definition, this transition probability has a tilt with a corresponding normalization function

$$p_\theta(x' \mid x, t) = \frac{R_0(x, x')\,\exp\big(u_\theta(x, x', t)\big)}{Z_\theta(x, t)}, \qquad Z_\theta(x, t) := \sum_y R_0(x, y)\,\exp\big(u_\theta(x, y, t)\big).$$

A similar result is derived for $p_t^\star(\cdot \mid x)$ with $u_t^\star$ and $Z^\star(x)$. Taking the logarithm yields, for every neighbor x' with $R_0(x, x') > 0$,

$$\log R_0(x, x') + u_{\theta^*}(x, x', t) - \log Z_{\theta^*}(x, t) = \log R_0(x, x') + u_t^\star(x, x') - \log Z^\star(x).$$

Canceling the common $\log R_0(x, x')$ term and rearranging gives

$$u_{\theta^*}(x, x', t) - u_t^\star(x, x') = \log Z_{\theta^*}(x, t) - \log Z^\star(x).$$

The right-hand side depends only on (x, t) (through the normalization constants) and is independent of the neighbor x'. Therefore there exists a scalar function $c(x, t)$ such that

$$u_{\theta^*}(x, x', t) = u_t^\star(x, x') + c(x, t)$$

for almost every (x, t) and for every neighbor x' with $R_0(x, x') > 0$, which is exactly the claimed identifiability up to an additive constant. □

Remark. We dissect the theoretical grounding of using a cross-entropy loss to train MadSBM.

1. **Simulation-free approximation.** Standard Schrödinger bridge solvers require computationally expensive iterative fitting of forward and backward projections (e.g. IPF, IMF). In contrast, MadSBM avoids explicit computation of these potentials. By learning the control field directly from the conditional flow of data, we construct a stochastic process $(X_t)_{t\in[0,T]}$ connecting the prior $X_0 \sim \mu_0$ to the data $X_T \sim \mu_1$ that also respects the biological constraints imposed by the reference process R_0 (ESM-2).

2. **Relationship to action minimization.** The cross entropy loss acts as a regression toward the optimal transport velocity. Intuitively, by maximizing the likelihood of the target data token x_1 given a corrupted state x_t, the model learns to recover the **optimal exponential tilt** of the reference process. This ensures that the generated trajectories are *action-minimizing* relative to the reference dynamics.

3. **Trivial Training Solution.** The natural parameterization is to learn optimal rates R_t^* directly by training with MSE loss is ill-posed due to extreme sparsity. For a sequence of length L and vocabulary $\mathcal{V}$, there are $L(|\mathcal{V}-1|)$ off-diagonal transitions, but only masked positions have non-zero targets. This leads to trivial zero-rate solutions and vanishing gradients. Our exponential parameterization $R_\theta = R_0 \exp(u_\theta)$ converts rate learning into a normalized classification problem, yielding the cross-entropy objective (Eq. equation 7), which avoids this degeneracy and learns the tilt directly. Empirically, training a MadSBM model with the same DiT backbone architecture, dataset splits, and hyperparameters under the MSE objective yields a PPL of 15.109 on the test set, indicating poor generalization and confirming this failure mode.

A.1.6 Control Error Implies Path-Space Stability

We next relate the mismatch between learned and optimal controls to the divergence between the induced path measures.

Theorem A.7 (Path-space KL bound under bounded log-rate error)**.** *Let $\mathbb{P}_\theta$ denote the path law induced by the controlled generator*

$$R_{\theta,t}(x,x') = R_0(x,x')\exp\big(u_\theta(x,x',t)\big), \tag{42}$$

and let $\mathbb{P}^\star$ denote the Schrödinger bridge induced by $u^\star$. Assume both processes share the same initial distribution μ_0. Define the pointwise log-rate error

$$\varepsilon_t(x,x') := u_\theta(x,x',t) - u_t^\star(x,x'). \tag{43}$$

Assume there exists $B>0$ such that for all (t,x,x') with $R_0(x,x')>0$,

$$|\varepsilon_t(x,x')| \le B \quad \textit{and} \quad |u_t^\star(x,x')| \le B. \tag{44}$$

Then the path-space divergence satisfies

$$\mathrm{KL}(\mathbb{P}_\theta \,\|\, \mathbb{P}^\star) \le C_B\, \mathbb{E}_{\mathbb{P}_\theta}\left[\int_0^T \sum_{x'\neq X_t} R_0(X_t,x')\,\varepsilon_t(X_t,x')^2\,dt\right], \tag{45}$$

where one admissible constant is

$$C_B := \frac{1}{2}e^{2B}(2B+1). \tag{46}$$

Proof. Apply Theorem A.1 with $(\mathbb{P},R_t) = (\mathbb{P}_\theta, R_{\theta,t})$ and $(\mathbb{P}_0,R_0) = (\mathbb{P}^\star, R_t^\star)$. Since initial laws match, the initial KL term vanishes. Using

$$R_{\theta,t}(x,x') = R_t^\star(x,x')\exp(\varepsilon_t(x,x')), \qquad R_t^\star(x,x') = R_0(x,x')\exp(u_t^\star(x,x')), \tag{47}$$

the integrand of equation 3 becomes

$$\begin{aligned} &R_{\theta,t}(x,x')\log\frac{R_{\theta,t}(x,x')}{R_t^\star(x,x')} - R_{\theta,t}(x,x') + R_t^\star(x,x') \\ &= R_t^\star(x,x')\left(e^\varepsilon\varepsilon - e^\varepsilon + 1\right), \end{aligned} \tag{48}$$

where $\varepsilon = \varepsilon_t(x,x')$. Define $\phi(\varepsilon) := e^\varepsilon\varepsilon - e^\varepsilon + 1$. Note that $\phi(0)=0$, $\phi'(0)=0$, and $\phi(\varepsilon)\ge 0$ for all ε because

$$\phi(\varepsilon) = \int_0^\varepsilon te^t dt. \tag{49}$$

On the bounded interval $[-B,B]$, ϕ is twice continuously differentiable and satisfies

$$|\phi''(t)| = |e^t(t+1)| \le e^B(B+1) \quad \text{for all } t\in[-B,B]. \tag{50}$$

Since $\phi(0) = \phi'(0) = 0$, Taylor's theorem with remainder yields, for $|\varepsilon| \leq B$,

$$0 \leq \phi(\varepsilon) \leq \frac{1}{2} e^B (B+1) \varepsilon^2. \tag{51}$$

Therefore,

$$\mathrm{KL}(\mathbb{P}_\theta \| \mathbb{P}^\star) \leq \frac{1}{2} e^B (B+1)\, \mathbb{E}_{\mathbb{P}_\theta} \left[\int_0^T \sum_{x' \neq X_t} R_t^\star(X_t, x')\, \varepsilon_t(X_t, x')^2 dt \right]. \tag{52}$$

Finally, under equation 44, $R_t^\star(x, x') = R_0(x, x') e^{u_t^\star(x,x')} \leq R_0(x, x') e^B$, so

$$\sum_{x'} R_t^\star(X_t, x') \varepsilon^2 \leq e^B \sum_{x'} R_0(X_t, x') \varepsilon^2. \tag{53}$$

Combining constants yields equation 45 with $C_B = \frac{1}{2} e^{2B}(B+1)$. The stated constant equation 46 is also admissible since $2B + 1 \geq B + 1$ for $B \geq 0$. □

A.1.7 Representational Completeness of Exponential Tilting

We formalize the completeness of the exponential-tilt parameterization on a fixed reference graph.

Proposition A.8 (One-to-one parameterization on a fixed support graph)**.** *Fix a reference generator R_0 and define its support graph*

$$E := \{(x, x') \in \mathcal{X} \times \mathcal{X} : x' \neq x,\ R_0(x, x') > 0\}. \tag{54}$$

Let R_t be any time-dependent generator such that $R_t(x, x') > 0$ implies $(x, x') \in E$. Then there exists a unique log-control field $u_t(x, x')$ on E such that

$$R_t(x, x') = R_0(x, x') \exp\big(u_t(x, x')\big), \qquad (x, x') \in E. \tag{55}$$

Conversely, any measurable u_t on E defines a generator R_t on E via this formula.

Proof. If $R_t(x, x')$ is supported on E, define $u_t(x, x') := \log\left(\frac{R_t(x,x')}{R_0(x,x')}\right)$ for $(x, x') \in E$. This is well-defined because $R_0(x, x') > 0$ on E. Uniqueness follows from injectivity of the logarithm on $(0, \infty)$. The converse direction is immediate by construction. □

A.1.8 Convergence of the Time-Discretized MadSBM Sampler

We analyze convergence of the distributional dynamics under a standard Euler discretization of the Kolmogorov forward equation.

Theorem A.9 (Convergence of Euler discretization of CTMC marginals)**.** *Let ρ_t denote the marginal distribution on $\mathcal{X}$ of a time-inhomogeneous CTMC with generator R_t and initial distribution $\rho_0 = \mu_0$, so that ρ_t solves*

$$\frac{d}{dt} \rho_t = \rho_t R_t. \tag{56}$$

Assume $t \mapsto R_t$ is Lipschitz in operator norm and uniformly bounded:

$$\|R_t\| \leq M, \qquad \|R_t - R_s\| \leq L|t - s| \quad \textit{for all } s, t \in [0, T]. \tag{57}$$

Let $\widehat{\rho}_k$ be the explicit Euler approximation with step size $\Delta t = T/K$:

$$\widehat{\rho}_{k+1} = \widehat{\rho}_k \left(I + \Delta t\, R_{t_k}\right), \qquad t_k = k\Delta t, \qquad \widehat{\rho}_0 = \mu_0. \tag{58}$$

Then there exists a constant $C = C(M, L, T)$ such that

$$\|\widehat{\rho}_K - \rho_T\|_1 \leq C\, \Delta t. \tag{59}$$

Proof. Equation 56 is a linear ODE on the finite-dimensional simplex, hence admits a unique continuously differentiable solution. Standard global error bounds for explicit Euler methods on Lipschitz ODEs give $\|\widehat{\rho}_k - \rho_{t_k}\|_1 \leq C\Delta t$ uniformly in k, with C depending on the Lipschitz constants of the vector field $\rho \mapsto \rho R_t$ and the time-variation of R_t. Since $\|\rho R_t\|_1 \leq \|\rho\|_1 \|R_t\| \leq M$ and $t \mapsto R_t$ is Lipschitz, the standard argument applies directly and yields the bound at $t = T$. □

Remark. Theorem A.9 justifies distributional convergence of the time-discretized sampling procedure used in MadSBM when the discretization is interpreted as Euler integration of the forward equation for marginals. A trajectory-level convergence statement can be obtained via standard coupling results for CTMC time discretizations on finite state spaces under additional uniform rate bounds.

A.2 Extended Results

A.2.1 Probability Paths

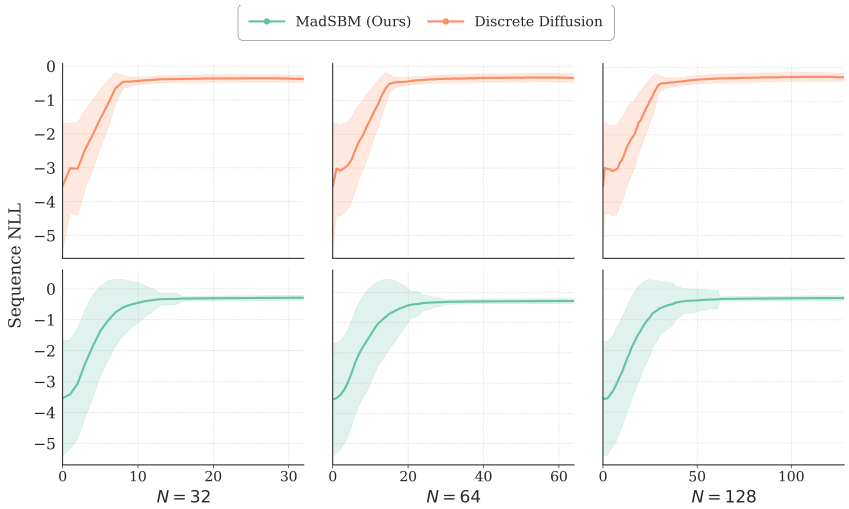

Figure A1: **Probability paths taken by models under various sampling budgets** (N). The y-axis represents the NLL of the sequence at the current iteration, assessed by the ESM-2-650M protein language model. The shaded area around the traced trajectory represents the standard deviation of the NLL at the currrent sampling iteration.

A.2.2 Low-Action Trajectories

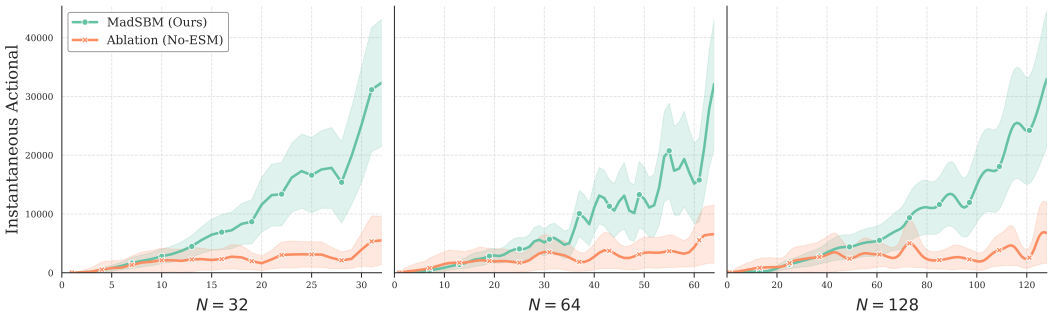

Figure A2: **Instantaneous actional values for MadSBM and ESM-ablated counterpart**. The y-axis represents the actional $\mathcal{A}_L(u)$ at the current timestep. Results are shown at various sampling budgets.

A.3 Extended Methods

A.3.1 Peptide Dataset

The dataset for MadSBM was curated from the PepNN, BioLip2, and PPIRef datasets (Abdin et al., 2022; Zhang et al., 2024; Bushuiev et al., 2023). All peptides from PepNN and BioLip2 were included, along with sequences from PPIRef ranging from 6 to 49 amino acids in length. The dataset was divided into training, validation, and test sets at an 80/10/10 ratio. To control for high sequence homology between peptides, we use MMSeqs2 to compute for each test sequence its maximum

sequence identity to any training sequence (Steinegger & Söding, 2017). We find that only 9.21% of test sequences exceed 30% identity to the training set. This indicates that a vast majority of sequences lie below standard homology thresholds, reducing the likelihood that performance is driven by memorization of highly similar training sequences.

A.3.2 Language Modeling for Biological Sequences

ESM-2 Protein Language Model Masked Language Models (MLMs) employ Transformer-based architectures to learn bi-directional sequence context, distant token relationships, and predict the identity of corrupted (masked) amino acid tokens. The model is trained under a sequence-recovery training objective, $\mathcal{L} = -\sum_{i \in \mathcal{M}} \log p_\theta(x^i | x^{\setminus \mathcal{M}})$, where $\mathcal{M}$ denotes the set of masked positions. MLMs are strong representation-learners and thus have been trained on evolutionary amino acid sequence datsets, e.g. the ESM-2 or ProtTrans family of models (Lin et al., 2023; Elnaggar et al., 2021). However, training these models to reconstruct only a minor fraction of tokens (15-40%) across a sequence makes complete de novo sequence generation difficult (Vincoff et al., 2025b), but provides a principled set of sequence representations to enable the training of generative models.

To ensure training and sampling sequences using ESM-2 logits as reference rates does not introduce evaluation bias, we use ProtBERT over ESM-2 to compute the pseudo-perplexity (PPL) metric of a sequence as a measure of biological plausibility. PPL is obtained by masking one token at a time, computing the NLL of the resulting sequence using the ProtBERT language modeling head, and averaging across all sequence positions. This procedure provides a tractable approximation to sequence likelihood for bidirectional MLMs, which do not admit a true autoregressive factorization.

Denoising Diffusion Models Diffusion models are a class of generative models defined by Markov processes (Ho et al., 2020; **?**). The *forward* diffusion steps $q(\mathbf{x}_{1:T}|\mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})$ progressively corrupt an initial data sample $\mathbf{x}_0 \sim q(\mathbf{x}_0)$ into a noisy prior $\mathbf{x}_T \sim q_{\text{noise}}$ across T timesteps. The noise distribution q_{noise} typically corresponds to a uniform categorical distribution over the vocabulary in the discrete space, Cat($|\mathcal{V}|$) (Tang et al., 2025b; Goel et al., 2025; Zhang et al., 2025a; Peng et al., 2025; Vincoff et al., 2025a), or an isotropic Gaussian, $\mathcal{N}(0, I)$, in continuous latent spaces. During inference, the learned *backward* process $p_\theta(\mathbf{x}_{0:T}) = p(\mathbf{x}_t) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$ gradually denoises the corrupted data sample to obtain samples from the true data distribution. Diffusion models are trained to maximize the evidence lower bound (ELBO): $\mathbb{E}_{q(\mathbf{x}_0)}\left[\log p_\theta(\mathbf{x}_0)\right] \geq \mathbb{E}_{q(\mathbf{x}_{0:T})}\left[\log \frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}\right]$

New data samples can be drawn by sampling from $q_{\text{noise}}(\mathbf{x}_T)$ and iteratively applying the learned denoising process $p_\theta(\mathbf{x}_{t-1}) = p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$. Various authors ((Sahoo et al., 2024), (Zheng et al., 2023)) have made simplifying assumptions about the reverse process to derive a computationally inexpensive loss function that reduces to a weighted negative log-likelihood, akin to a weighted form of the NLL over masked tokens. In particular, the state-of-the-art discrete diffusion protein language models DPLM Wang et al. (2024) and EvoFlow used to benchmark MadSBM employ the Reparameterized Diffusion Model strategy from Zheng et al..

A.3.3 Modeling MadSBM

We parameterize the control field u_θ using a ~50M parameter Diffusion Transformer (DiT) backbone operating over discrete peptide sequences. The model consists of 2 DiT layers, each with Multi-head self-attention and Adaptive LayerNormalization (AdaLN). Time-conditioning is achieved with a Gaussian Fourier projection and a learned MLP head. Dynamic batching is used during training for GPU efficiency. The DiT model was trained for 50 epochs (127k steps) on a 4xA6000 GPU system with 192 GB of shared memory. The learning rate was initialized at $1e^{-6}$ and increased with a linear schedule for 2 epochs to $1e^{-4}$, then decayed with a cosine scheduler to $1e^{-6}$. The AdamW optimzer was used with $\beta_1 = 0.9, \beta_2 = 0.999$ and weight decay of 0.01.

A.3.4 Instantaneous Actionals

The direct discretization of computing the action functional for a L-length sequence with $\mathcal{V} = 33$ vocabulary positions with an N-step sampling budget is the value of the integrand from Eq. equation 4:

$$\mathcal{A}(u) \;=\; \Delta t \cdot \frac{1}{L} \sum_{\ell=1}^{L} \sum_{v=1}^{V} R_0[\ell, v] \; \Psi\big(u[\ell, v]\big) \tag{60}$$

Table 5: Diffusion Transformer Architecture.

Layer	Input Dimension	Output Dimension
Sequence embeddings		
ESM (encoder)	vocab size	1280
ESM (LM head)	1280	vocab size
Time embedding		
Gaussian Fourier projection	1	64
Time embedding projection	64	512
DiT Blocks $\times 2$		
AdaLN	1280	1280
AdaLN time-conditioning	512	2×1280
MHSA ($h = 16$)	1280	1280
MLP (FFN) + GeLU	1280	1280
hidden dim = 5120		
Dropout + Residual	1280	1280
Final layers		
LayerNorm	1280	1280
Linear projection	1280	vocab size

where $\Psi(z) = ze^z - e^z - 1$ as before and $\Delta t := 1/N$, where N is the sampling budget. Computing this value directly is numerically unstable as both R_0 and $\Psi(u)$ are dependent on unbounded neural network logits, exacerbated through the relationship $\Psi(u) \propto e^u$. As a practical solution to obtain an interpretable upper bound, we define the reference process and control field in the *worst-case scenario*. For the reference rate, we adopt a uniform random-walk reference process at each token position, *i.e.* $R_0[\ell, \cdot] = 1/V$, so that $\sum_v R_0[\ell, v] = 1$. To simplify the control field, we consider the pathological scenario in which the model assigns large transition rates to all possible vocabulary positions for each sequence token. We approximate this large transition rate by evaluating the trained model on the held-out test set and recording the maximum observed logit value M. Recall that since $R_u(x, x') = R_0(x, x') \exp(u_\theta(x, x'))$, we derive M from the logit values produced by the DiT model parameterizing u_θ and ignore the ESM logits forming R_0. Using M, we define the constant tilt $u^w \in \mathbb{R}^{L \times V}$ with entries $u^w_{\ell,v} = M$ for all ℓ and v. Under this worst-case tilt and the uniform reference process, the total actional for an L-length sequence simplifies to

$$\begin{aligned} \mathcal{A}_L(u^w) &= \Delta t \cdot \Psi(u^w) \\ &= \Delta t \cdot L\big(Me^M - e^M - 1\big). \end{aligned} \tag{61}$$

In total, this value approximates Eq. 4, giving us a bound to asses if MadSBM's instantaneous actionals correlate to a low-cost transport plan.

A.4 ALGORITHM PSEUDOCODE

Algorithm 1 MadSBM Training

Require: Dataset $\mathcal{D}$, control field u_θ, reference prior f_ϕ (ESM-2), max time T
while not converged **do**
  Sample batch $x_1 \sim \mathcal{D}$ and prior $x_0 \sim \mu_0$ (Fully Masked)
  Sample timestep $t \sim \mathcal{U}(0, T)$
  Corrupt sequence: $x_t \leftarrow \text{Interpolate}(x_0, x_1, t)$
  Compute total logits using biological prior: $\mathbf{z}_t = u_\theta(\mathbf{x}_t, t) + f_\phi(\mathbf{x}_t)$
  Compute negative log-likelihood: $\mathcal{L}(\theta) = -\sum_{i \in \mathcal{M}} \log p_\theta(x_1^{(i)} \mid x_t, t)$
  Take gradient descent step on: $\nabla_\theta \mathcal{L}$
end while
return Trained control field u_θ

Algorithm 2 MadSBM Sampling

Require: Control field u_θ, masked prior μ_0, steps K, hyperparameters λ, β, τ, p
Initialize $x_K \sim \mu_0$ (Fully Masked) and define step size $\Delta t \leftarrow 1/K$
for $k = K \to 1$ **do**
  Set time $t \leftarrow k/K$
  Compute total rates: $\mathbf{R}_{\text{tot}} \leftarrow \sum_v \exp(\lambda \cdot u_\theta(x_k, v, t))$
  Compute jump probabilities: $\mathbf{p}_{\text{jump}} \leftarrow 1 - \exp\big(-\mathbf{R}_{\text{tot}} \cdot \beta \cdot \Delta t\big)$
  Sample active mask: $\mathbf{z} \sim \text{Bernoulli}(\mathbf{p}_{\text{jump}})$
  Filter logits: $\hat{u} \leftarrow \text{Top-}p([u_\theta(x_k, \cdot, t) + (1-t) f_\phi(xt)]/\tau)$
  Sample candidates: $x_{\text{new}} \sim \text{Multinomial}(\text{Softmax}(\hat{u}))$
  Update each state: $x_{k-1}^{(i)} \leftarrow x_{\text{new}}^{(i)}$ if $z^{(i)} = 1$ and $x_k^{(i)} = \texttt{[MASK]}$, else $x_k^{(i)}$
end for
return x_0