ACCELERATION OF DIFFUSION LANGUAGE MODEL THROUGH DISCRETE AVERAGE GENERATOR

Anonymous authors
Paper under double-blind review

ABSTRACT

Discrete diffusion models and flow matching have emerged as powerful frameworks for generative modeling over discrete state spaces, yet efficient few-step generation remains a fundamental challenge. In this work, we introduce the *Discrete Average Generator*, a principled extension of MeanFlow to Continuous-Time Markov Chains (CTMCs). Analogously to how MeanFlow defines an average velocity field over a time interval in continuous spaces, we define an average generator as the normalized increment of the transition kernel over a time interval. We show that this average generator satisfies a self-consistency identity, which provides the foundation for our training objective. We further develop training strategies that align with the standard training paradigm of diffusion language models while keeping the resulting objective tractable. When projected onto per-coordinate marginals, the self-consistency identity admits a closed-form expression, enabling efficient training and inference. In Potts model simulations, our objective reduces the total variation distance of the K-step sampler by up to 67%. On OpenWebText, our method achieves the lowest generative perplexity among the evaluated methods for 8 to 64 sampling steps while enabling a $16\times$ acceleration, and achieves comparable performance to existing methods on ImageNet.

1 INTRODUCTION

Diffusion models and flow matching have achieved remarkable success in continuous domains such as image and audio generation (Lipman et al., 2024). Extending these generative frameworks to discrete spaces, such as natural language, has attracted growing interest, with discrete flow matching and masked diffusion models emerging as promising approaches (Campbell et al., 2024; Gat et al., 2024; Shi et al., 2024; Sahoo et al., 2024). However, a fundamental challenge shared with their continuous counterparts remains: sampling requires multiple sequential network evaluations, making inference computationally expensive. The difficulty is compounded in discrete spaces, where practical samplers update all coordinates independently within a step: with few steps, each coordinate is drawn from a law computed from the instantaneous rate rather than from the finite-step kernel, and the dependence between coordinates updated together is lost.

In continuous settings, this bottleneck has been addressed by consistency models (Song et al., 2023), progressive distillation (Salimans & Ho, 2022), and more recently MeanFlow (Geng et al., 2025a), which defines an average velocity field over time intervals and enables high-quality generation in very few steps without multi-step distillation. The key insight of MeanFlow is that the average velocity satisfies a self-consistent identity involving the instantaneous velocity and its material derivative, which can be exploited as a fixed-point training objective.

In this work, we propose the *Discrete Average Generator*, a principled extension of MeanFlow to discrete state spaces governed by Continuous-Time Markov Chains (CTMCs). The central challenge is that the continuous notion of velocity and its spatial gradient has no direct analogue in discrete spaces. We resolve this by defining the CTMC average generator over an interval $[t, r]$ as the normalized increment of the transition kernel, i.e. the unique object for which a single Euler step of size $r - t$ reproduces the exact transition law. We further derive an average generator identity that forms the basis of our training objective. Inspired by the progress of velocity prediction (u-prediction / v-prediction) to data prediction (x-prediction) made in continuous MeanFlow literature described in

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

Section 2.2, we develop a corresponding progression from transition rate prediction (U-prediction / Q-prediction) to posterior prediction ($\tilde{p}$-prediction) in Section 3.2.

Although the $\tilde{p}$-prediction is compatible with existing pretrained diffusion language models, it is intractable at the sequence level because of the huge computational cost associated with the multiplication of the generator matrix with dimension $|\mathcal{S}|^{\mathcal{D}} \times |\mathcal{S}|^{\mathcal{D}}$. By projecting the identity onto per-coordinate marginals, as described in Section 3.3, we obtain a tractable closed-form involving only $\mathcal{D} \times |\mathcal{S}|$-dimensional quantities. Our main contributions are as follows:

- We introduce the Average Generator for CTMCs and prove the Discrete Average Generator Identity, the discrete analogue of the MeanFlow identity.
- We derive a tractable training objective based on $\tilde{p}$-prediction and per-coordinate marginal projection, which aligns with existing diffusion language model parameterizations and admits a closed-form expression involving only $\mathcal{D} \times |\mathcal{S}|$-dimensional quantities.
- We validate our approach through simulations, text generation on the OpenWebText dataset, and image generation on the ImageNet dataset. On OpenWebText, our method achieves the lowest generative perplexity among the compared methods for 8 to 64 sampling steps and maintains lower generative perplexity with $16\times$ fewer generation steps against the strongest baseline. On ImageNet, our method achieves comparable FID and Inception Score to existing acceleration methods for diffusion language models.

2 Preliminary

2.1 Discrete Flow Matching

Consider a Continuous-Time Markov Chain (CTMC) defined on the state space $\mathcal{S}^{\mathcal{D}}$ with marginal probability mass functions $\{p_t\}_{t\in[0,1]}$. Let Q_t denote the associated transition rate matrix (i.e., the generator) at time t, which has dimensions $|\mathcal{S}|^{\mathcal{D}} \times |\mathcal{S}|^{\mathcal{D}}$. For any distinct states $x, z \in \mathcal{S}^{\mathcal{D}}$ (where $x \neq z$), the entry $Q_t(x, z)$ represents the instantaneous transition intensity from state x to state z at time t. If $Q_t(x, z)$ satisfies the following rate-properties:

$$Q_t(x, z) \geq 0, \text{ for any } z \neq x, \text{ and } \sum_{z\in\mathcal{S}^{\mathcal{D}}} Q_t(x, z) = 0,$$

and the Kolmogorov forward equation (also known as the continuity equation):

$$\dot{p}_t(x) = \sum_{z\in\mathcal{S}^{\mathcal{D}}} Q_t(z, x)p_t(z) = \sum_{z\neq x} Q_t(z, x)p_t(z) - \sum_{z\neq x} Q_t(x, z)p_t(x), \tag{1}$$

the transition rate Q_t is also the generator of the CTMC $\{p_t\}_{t\in[0,1]}$. In particular, $p_{t+h|t}(x|z) = \delta_z(x) + Q_t(z, x)h + o(h)$ for a small time step h.

To learn a transition rate Q_t that can transport from an initial noise distribution p_0 to a target data distribution p_1, a well-developed method is to learn the conditional expectation of the conditional transition rate $Q_t(x, z|x_1)$ that generates the conditional probability path $p_{t|1}(\cdot|x_1)$. Then $Q_t(x, z) = \mathbb{E}_{p_{1|t}(x_1|x)}[Q_t(x, z|x_1)]$ can generate the target probability path p_t (see Proposition 3.1 of Campbell et al., 2024), i.e., it satisfies the Kolmogorov forward Equation 1. For completeness, we include the proof in Appendix B.2. We are free to define the conditional probability path $p_{t|1}(\cdot|x_1)$ and with the corresponding conditional transition rate as needed.

A common approach is to construct a coordinate-wise conditional probability path and transition rate,

$$p_{t|1}(x|x_1) = \prod_{d=1}^{\mathcal{D}} p^d_{t|1}(x^d|x^d_1), \qquad Q_t(x, z|x_1) = \sum_{d=1}^{\mathcal{D}} \delta_{x^{\backslash d}}(z^{\backslash d})Q^d_t(x^d, z^d|x^d_1), \tag{2}$$

where we use the notation $x^{\backslash d}$ to denote the vector obtained by removing the d-th coordinate of x, i.e., $x^{\backslash d} := (x^1, \ldots, x^{d-1}, x^{d+1}, \ldots, x^{\mathcal{D}})$, $Q^d_t(x^d, z^d|x^d_1)$ is the conditional transition rate that generates the conditional probability path $p^d_{t|1}$, and $\delta_x(z)$ the Kronecker delta satisfying $\delta_x(z) = 1$ if

$x = z$ and $\delta_x(z) = 0$ if $x \neq z$. A popular choice of probability path and the associated conditional transition rate used in the previous works (Campbell et al., 2024; Gat et al., 2024) is

$$p^d_{t|1}(x^d|x^d_1) = (1-\kappa_t)p^d_0(x^d) + \kappa_t \delta_{x^d_1}(x^d), \qquad Q^d_t(x^d, z^d|x^d_1) = \lambda_t\big(\delta_{x^d_1}(z^d) - \delta_{x^d}(z^d)\big), \quad (3)$$

where $\kappa_t : [0,1] \to [0,1]$ is a non-decreasing function satisfying $\kappa_0 = 0$ and $\kappa_1 = 1$, and $\lambda_t \triangleq \frac{\dot\kappa_t}{1-\kappa_t}$. Common sources are the uniform distribution $p^d_0 = \mathrm{Unif}(\mathcal{S})$ and the masked source $p^d_0 = \delta_{\mathbf{m}}$, which puts all mass on a mask token $\mathbf{m}$. After defining the conditional path and rate, the marginal transition rate is given by

$$\begin{aligned} Q_t(x,z) = \sum_{x_1} Q_t(x,z|x_1)p_{1|t}(x_1|x) &= \sum_{d=1}^{\mathcal{D}} \delta_{x^{\setminus d}}(z^{\setminus d}) \sum_{x^d_1} Q^d_t(x^d, z^d|x^d_1)p^d_{1|t}(x^d_1|x) \\ &\triangleq \sum_{d=1}^{\mathcal{D}} \delta_{x^{\setminus d}}(z^{\setminus d}) Q^d_t(x, z^d), \end{aligned}$$

where $p^d_{1|t}(x^d_1|x) = \sum_{x_1^{\setminus d}} p_{1|t}(x_1|x)$ is the posterior. When the coordinate is clear from the argument we drop the superscript and write $p_{1|t}(z^d|x)$ for $p^d_{1|t}(z^d|x)$; for the mixture path, $Q^d_t(x, z^d) = \lambda_t\, p_{1|t}(z^d|x)$ for $z^d \neq x^d$.

Training objective To learn the transition rate, a common approach is to minimize the Bregman divergence between the conditional generator and the estimated generator (Holderrieth et al., 2025; Lipman et al., 2024; Shaul et al., 2025; Wan et al., 2025):

$$\mathbb{E}_{x_1,t,x_t\sim p_{t|1}(x_t|x_1)}\Big[\sum_{z\neq x_t} D_F\Big(Q_t(x_t,z|x_1)\Big\|Q_t(x_t,z)\Big)\Big],$$

where D_F denotes the Bregman divergence induced by $F(x) = x\log x$, $D_F(a\|b) \triangleq F(a) - F(b) - F'(b)(a-b) = a\log\frac{a}{b} - a + b$, which is defined for $a \geq 0$ and $b > 0$ and vanishes if and only if $a = b$. After some derivation in Appendix B.1, it is equivalent to learn the posterior network $p_{1|t}(x^d_1|x_t)$ under the generalized Kullback–Leibler (KL) divergence:

$$\mathbb{E}_{x_1,t,x_t\sim p_{t|1}(x_t|x_1)}\Big[\lambda_t \sum_{d=1}^{\mathcal{D}}\Big\{-\Big(1-\delta_{x^d_1}(x^d_t)\Big)\log p^d_{1|t}(x^d_1|x_t) + \delta_{x^d_1}(x^d_t) - p^d_{1|t}(x^d_t|x_t)\Big\}\Big], \quad (4)$$

which is equivalent to Equation 37 of Shaul et al. (2025). If we use the masked source with the mixture path, then Equation 4 recovers the training objective for masked diffusion models (Shi et al., 2024; Sahoo et al., 2024; Ou et al., 2025; Nie et al., 2025).

2.2 MeanFlow

In MeanFlow Geng et al. (2025a), the authors define the average velocity field in continuous space as the displacement from time t to r divided by $r - t$ ($0 < t < r < 1$) as

$$u\left(x_t, t, r\right) \triangleq \frac{1}{r-t}\int_t^r v\left(x_s, s\right) ds,$$

where $v\left(x_s, s\right)$ denotes the instantaneous velocity, $t = 0$ corresponds to the noise, and $t = 1$ corresponds to the data. Then, the MeanFlow identity can be derived (see Appendix B.3):

$$u(x_t,t,r) = v(x_t,t) - (t-r)[\partial_t u(x_t,t,r) + v(x_t,t)\cdot\nabla_{x_t}u(x_t,t,r)].$$

Therefore, the training objective for MeanFlow can be formulated as the following fixed-point optimization problem:

$$\mathcal{L}(\theta) = \mathbb{E}_{x_1,t,r,x_t\sim p_{t|1}(x_t|x_1)}\left\|u_\theta(x_t,t,r) - \mathrm{sg}\left(u_{\text{tgt}}(x_t,t,r)\right)\right\|_2^2,$$

where $u_{\text{tgt}}(x_t,t,r) = v(x_t,t) - (t-r)\left(v(x_t,t)\cdot\nabla_x u_\theta(x_t,t,r) + \partial_t u_\theta(x_t,t,r)\right)$ and $\mathrm{sg}(\cdot)$ denotes the stop-gradient operator.

v-prediction parameterization In Geng et al. (2025b), the authors identify an underlying limitation of MeanFlow, i.e., the target $u_{\text{tgt}}(x_t, t, r)$ depends on the unknown oracle $u(x_t, t, r)$, and is therefore approximated using the network prediction $u_\theta(x_t, t, r)$. This self-referential construction introduces bias and leads to unstable training.

To address this issue, they propose a velocity-based reparameterization derived from the MeanFlow identity:

$$V(x_t, t) = u_\theta(x_t, t, r) + (t - r)\ \left(v(x_t, t) \cdot \nabla_x u_\theta(x_t, t, r) + \partial_t u_\theta(x_t, t, r)\right). \tag{5}$$

Instead of directly regressing u, the model is trained via a standard velocity regression objective:

$$\mathbb{E}_{x_1, r, t, x_t \sim p_{t|1}(x_t|x_1)} \left\| V(x_t, t) - v^{\text{oracle}}(x_t, t) \right\|_2^2. \tag{6}$$

In practice, $v(x_t, t)$ can be obtained either via the boundary condition $v(x_t, t) = u_\theta(x_t, t, t)$ or via an auxiliary prediction head. This improved reparameterization significantly reduces variance in calculating the Jacobian-vector product, leading to more stable optimization.

x-prediction parameterization Beyond velocity prediction, an alternative parameterization is to directly model the clean data. As observed in Lu et al. (2026), predicting x_1 instead of the velocity field better preserves the data manifold and improves training stability. Therefore, they reparameterize the MeanFlow as

$$u_\theta(x_t, t, r) = \frac{x_\theta(x_t, t, r) - x_t}{1 - t}.$$

Substituting this into Equation 5 and Equation 6, we obtain the training objective for MeanFlow under the x-prediction parameterization. This formulation further improves empirical stability.

3 Proposed Method

Table 1: Conceptual and mathematical comparison between MeanFlow and the proposed Discrete Average Generator.

Property	**MeanFlow**	**Discrete Average Generator**
State Space	Continuous domain ($\mathbb{R}^d$)	Discrete space ($\mathcal{S}^{\mathcal{D}}$) via CTMCs
Instantaneous Measure	Velocity field $v(x_t, t)$	Transition rate generator $Q_t(x, z)$
Average Measure	Average Velocity: $u(x_t, t, r) = \frac{1}{r-t}\int_t^r v(x_s, s)ds$	Average Generator: $U_{t,r} = \frac{P_{t\to r} - I}{r-t}$
Material Derivative	Spatial gradient used: $\frac{d}{dt}u = \partial_t u + v \cdot \nabla_x u$	Dynkin's formula used: $\frac{D}{Dt}U = \partial_t U + Q_t U$
Fixed-Point Identity	$u = v - (t-r)(\partial_t u + v \cdot \nabla_x u)$	$U_{t,r} = Q_t - (t-r)(\partial_t U_{t,r} + Q_t U_{t,r})$
Stable Target Parameterization	x-prediction (predicting clean data x_1)	$\tilde{p}$-prediction (predicting a per-coordinate distribution $\tilde{p}_{t,r}(\cdot\|x_t)$)

3.1 Average Generator

In continuous MeanFlow, acceleration is enabled by integrating the instantaneous velocity field, where a single Euler step of size $r - t$ reproduces the endpoint of the flow exactly. We define the analogue CTMC generator for discrete spaces. Let $P_{t\to r}$ denote the transition kernel of the CTMC from time t to r, i.e., $P_{t\to r}(x, z) = p_{r|t}(z \mid x)$. We define the *average generator* over $[t, r]$ as

$$U_{t,r} \triangleq \frac{P_{t\to r} - I}{r - t}, \qquad \text{and} \qquad U_{t,t} \triangleq Q_t, \tag{7}$$

so that a single Euler step is exact, $\delta_x^\top \left(I + (r-t)U_{t,r}\right) = p_{r|t}(\cdot|x)$ for every x. By Proposition 2 in Appendix C.2, $U_{t,r}$ is itself a valid generator (nonnegative off-diagonal entries, zero row sums) and satisfies $U_{t,r} \to Q_t$ as $r \to t$. We derive the following identity for the transition rate in a discrete state space, with the proof provided in Appendix C.2.

Proposition 1 (Average Generator Identity)**.** *For a sufficiently smooth generator Q_t, the average generator satisfies:*

$$U_{t,r} = Q_t - (t-r)\left(\partial_t U_{t,r} + Q_t U_{t,r}\right). \tag{8}$$

Equation 8 expresses the average generator as a self-consistent correction of the instantaneous generator. We call $Q_t U_{t,r}$ the *transport term*. We compare the proposed framework with its continuous counterpart in Table 1. This identity forms the basis for a fixed-point training objective described in the following section.

3.2 Training Objective of Discrete Average Generator

U-prediction objective. Equation 8 motivates directly regressing $U^\theta_{t,r}$ against the stop-gradient target constructed from the right-hand side, where $\mathrm{sg}(\cdot)$ denotes the stop-gradient operator. With $\widehat{U}^\theta_{t,r}(x_t, z^d) \triangleq Q_t(x_t, z^d) - (t-r)\big(\partial_t U^\theta_{t,r}(x_t, z^d) + (Q_t U^\theta_{t,r})(x_t, z^d)\big)$, the U-prediction objective is

$$\mathcal{L}_U(\theta) = \mathbb{E}_{x_1,(t,r),x_t\sim p_{t|1}(x_t|x_1)}\left[\sum_{d=1}^{\mathcal{D}}\sum_{z^d\neq x_t^d} D_F\Big(\mathrm{sg}\Big(\widehat{U}^\theta_{t,r}(x_t, z^d)\Big)\,\Big\|\,U^\theta_{t,r}(x_t, z^d)\Big)\right], \tag{9}$$

with (t, r), $0 \le t \le r \le 1$, drawn from the design of Appendix D.2. However, analogously to the continuous setting (Geng et al., 2025a), the target in Equation 9 depends on $U^\theta_{t,r}$ itself, introducing a self-referential bias that can destabilize training; moreover, unlike $U_{t,r}(x_t, z^d)$, it need not be nonnegative, as D_F requires.

Q-prediction objective. To resolve this issue, we follow the reparameterization strategy of Geng et al. (2025b) and express Q_t directly in terms of $U^\theta_{t,r}$ via the Average Generator Identity:

$$Q^\theta_{t,r}(x_t, z^d) \triangleq U^\theta_{t,r}(x_t, z^d) + \mathrm{sg}\Big((t-r)\Big(\partial_t U^\theta_{t,r}(x_t, z^d) + \big(Q_t\, U^\theta_{t,r}\big)(x_t, z^d)\Big)\Big). \tag{10}$$

Instead of regressing $U^\theta_{t,r}$ directly, we train by regressing the reparameterized instantaneous generator $Q^\theta_{t,r}$ against the conditional rate $Q_t(x_t, z^d|x_1)$:

$$\mathcal{L}_Q(\theta) = \mathbb{E}_{x_1,(t,r),x_t\sim p_{t|1}(x_t|x_1)}\left[\sum_{d=1}^{\mathcal{D}}\sum_{z^d\neq x_t^d} D_F\Big(Q_t(x_t, z^d|x_1)\,\Big\|\,Q^\theta_{t,r}(x_t, z^d)\Big)\right]. \tag{11}$$

This eliminates the self-referential dependence in the target, which in the continuous case yields markedly more stable optimization (Geng et al., 2025b).

$\tilde{p}$-prediction objective. Analogously to the x-prediction parameterization in the continuous setting (Lu et al., 2026), we further reparameterize $U^\theta_{t,r}$ through a distribution $\tilde{p}_{t,r}(\cdot|x)$ over $\mathcal{S}$ for each coordinate, state x and pair (t, r), whose dependence on θ we leave implicit. Starting from $Q^d_t(x, z^d) = \sum_{x_1^d} p^d_{1|t}(x_1^d|x)\, Q^d_t(x^d, z^d|x_1^d) = \lambda_t\, p_{1|t}(z^d|x)$ for $z^d \neq x^d$ and modeling $U^\theta_{t,r}$ as a generator with the same single-coordinate structure, we set

$$U^\theta_{t,r}(x, z) = \mu_{t,r}\,\tilde{p}_{t,r}(z^d|x), \quad x^{\backslash d} = z^{\backslash d},\ z^d \neq x^d, \qquad \mu_{t,r} \triangleq \frac{\kappa_r - \kappa_t}{(r-t)(1-\kappa_t)}, \tag{12}$$

where κ_r denotes the schedule of Equation 3 evaluated at r. The scale $\mu_{t,r}$ makes the one-step kernel of a coordinate, $\delta_{x^d} + (r-t)U^\theta_{t,r}(x, \cdot)$, equal to $(1-\omega_{t,r})\delta_{x^d} + \omega_{t,r}\tilde{p}^d_{t,r}(\cdot|x)$ with $\omega_{t,r} \triangleq (r-t)\mu_{t,r} = \frac{\kappa_r-\kappa_t}{1-\kappa_t} \in [0, 1]$, the probability that a coordinate jumps to x_1^d on $[t, r]$ under the conditional path.It satisfies $\mu_{t,r} \to \lambda_t$ as $r \to t$, so the boundary condition $U^\theta_{t,t} = Q_t$ holds whenever $\tilde{p}_{t,t} = p_{1|t}$. Directly substituting Equation 12 into Equation 10 formally gives an objective in $\tilde{p}_{t,r}$, but its transport term $(Q_t U^\theta_{t,r})(x_t, z^d)$ poses several obstacles. In Section 3.3, we first project the transport term onto per-coordinate marginals and adopt a closed-form expression and a one-pass unbiased estimator to reduce the computational costs.

3.3 Tractable Training Objective of Discrete Average Generator

Under the coordinate-wise construction Equation 2, $Q_t(x, y) = 0$ unless x and y differ in exactly one coordinate, so Q_t is determined by the $\mathcal{D} \times |\mathcal{S}|$ numbers $\lambda_t\, p_{1|t}(z^d|x)$ per state, which we obtain from a frozen pretrained denoiser, and $U^\theta_{t,r}$ has the same structure by Equation 12. We therefore model $\tilde{p}_{t,r}$ with a network H^θ that maps (x, t, r) to $\mathcal{D}$ distributions over $\mathcal{S}$, $\tilde{p}_{t,r}(z^d|x) = [H^\theta(x, t, r)]_{d,z^d}$, so that a single forward pass gives every entry of $U^\theta_{t,r}$ at the current state.

Per-coordinate projection. Let $x^{d\to s}$ denote the state x with its d-th coordinate replaced by s. For each coordinate d, the *per-coordinate projection* Π^d is the linear map that marginalizes the target state of a matrix onto its d-th coordinate: it takes a matrix M indexed by pairs of states $(x, y) \in \mathcal{S}^\mathcal{D} \times \mathcal{S}^\mathcal{D}$ and returns the matrix $\Pi^d[M]$ indexed by $(x, z^d) \in \mathcal{S}^\mathcal{D} \times \mathcal{S}$,

$$\Pi^d[M](x, z^d) \triangleq \sum_{y \in \mathcal{S}^\mathcal{D}:\ y^d = z^d} M(x, y), \qquad z^d \neq x^d, \tag{13}$$

i.e. $\Pi^d[M] = ME_d$ with $E_d(y, z^d) \triangleq \mathbb{1}[y^d = z^d]$, which sums the row $M(x, \cdot)$ over all target states whose d-th coordinate equals z^d, whatever the other coordinates are. For a transition kernel, $\Pi^d[M](x, \cdot)$ is the law of the d-th coordinate after the transition. For a generator, $\Pi^d[M](x, z^d)$ is the total rate at which coordinate d moves to z^d. For a generator that changes only one coordinate at a time, the d-th coordinate transition is given by a single matrix entry, $\Pi^d[Q_t](x, z^d) = Q_t(x, x^{d\to z^d}) = \lambda_t\, p_{1|t}(z^d|x)$ and $\Pi^d[U^\theta_{t,r}](x, z^d) = \mu_{t,r}\tilde{p}_{t,r}(z^d|x)$. For the exact average generator, however, the d-th coordinate transition is given by the d-th marginal of the jump kernel: $\Pi^d[U_{t,r}](x, z^d) = p^d_{r|t}(z^d|x)/(r - t)$, which is exactly the quantity required by a coordinate-factorized sampler. In contrast, the entry $U_{t,r}(x, x^{d\to z^d})$ is proportional to the probability that coordinate d jumps to z^d while all other coordinates remain unchanged, with proportionality factor $1/(r - t)$. Applying Π^d to Equation 8 gives, for every d and $z^d \neq x^d$, the projected identity underlying Section 3.2,

$$\begin{aligned}\Pi^d[U_{t,r}](x, z^d) &= \Pi^d[Q_t](x, z^d) - (t - r)\Big(\partial_t \Pi^d[U_{t,r}](x, z^d) + \Pi^d[Q_t U_{t,r}](x, z^d)\Big) \\ &= \Pi^d[Q_t](x, z^d) - (t - r)\Big(\partial_t \Pi^d[U_{t,r}](x, z^d) + \sum_{x'} Q_t(x, x')\, \Pi^d[U_{t,r}](x', z^d)\Big).\end{aligned} \tag{14}$$

Here we use that Π^d is linear and commutes with ∂_t, and that $\Pi^d[Q_t M](x, z^d) = \sum_{x'} Q_t(x, x')\, \Pi^d[M](x', z^d)$. The right-hand side involves $U_{t,r}$ only through $\Pi^d[U_{t,r}]$ at x and at its single-coordinate neighbors. The projection therefore gives the transport term a well-defined per-coordinate entry and absorbs the two-coordinate moves by summing over them. We call $\Pi^d[Q_t U_{t,r}]$ the *projected transport term*.

Tractable projected transport term. The projected identity Equation 14 is a *closed* equation for the per-coordinate marginals: the unique solution of Equation 14 is the exact marginal $\Pi^d[U_{t,r}]$, which is proved in Proposition 5. A sufficiently expressive model trained with Equation 11 therefore learns the per-coordinate marginals of the finite-step kernel, using only the instantaneous rate and quantities at the current state, without simulating the kernel over $[t, r]$. It remains to evaluate the projected transport term for the parameterization Equation 12. We show in Proposition 4 that

$$\Pi^d\big[Q_t\, U^\theta_{t,r}\big](x_t, z^d) = \lambda_t\, \mu_{t,r}\Big[\, C^d_\theta(x_t, z^d) - p_{1|t}(z^d \mid x_t)\Big], \tag{15}$$

where

$$C^d_\theta(x_t, z^d) \triangleq \sum_{e=1}^{\mathcal{D}} \Big(\mathbb{E}_{s\sim p^e_{1|t}(\cdot|x_t)}\big[\tilde{p}_{t,r}\big(z^d \mid x_t^{e\to s}\big)\big] - \tilde{p}_{t,r}\big(z^d \mid x_t\big)\Big) \tag{16}$$

measures how the prediction at coordinate d moves when a single coordinate of x_t jumps according to Q_t. Each summand requires evaluating the network at a state that differs from x_t in a single coordinate, producing an output of size $\mathcal{D} \times |\mathcal{S}|$. Thus, evaluating the sum exactly requires one forward pass for each such state, motivating the estimator introduced below. Substituting Equation 15 and $\partial_t \mu_{t,r} = \frac{\mu_{t,r} - \lambda_t}{r - t} + \lambda_t \mu_{t,r}$ into Equation 10, the purely local terms combine and the reparameterized instantaneous rate becomes

$$Q^\theta_{t,r}(x_t, z^d) = \lambda_t\, \tilde{p}_{t,r}(z^d \mid x_t) + \mathrm{sg}\Big\{(t - r)\, \mu_{t,r}\Big[\partial_t \tilde{p}_{t,r}(z^d \mid x_t) + \lambda_t\, \Gamma^d_\theta(x_t, z^d)\Big]\Big\}, \tag{17}$$

Algorithm 1 Training of Discrete Average Generator

Require: Network H^θ, schedule κ_t, data distribution p_1, frozen denoiser H, boundary fraction $1-\rho$

repeat
  Sample $x_1 \sim p_1$, (t, r) with $t \le r$ from the logit-normal; set $r \leftarrow t$ with probability $1-\rho$
  Sample $x_t \sim p_{t|1}(\cdot \mid x_1)$ ▷ mixture path
  Compute $\tilde{p}_{t,r}(\cdot \mid x_t) \leftarrow H^\theta(x_t, t, r)$, $\partial_t \tilde{p}_{t,r}(z^d \mid x_t)$ at *fixed* r ▷ $\mathcal{D} \times |\mathcal{S}|$
  Compute $p_{1|t}(\cdot \mid x_t) \leftarrow H(x_t, t)$ ▷ frozen, stop-gradient
  Sample $e \sim \mathrm{Unif}(\{1, \dots, \mathcal{D}\})$, $s \sim p^e_{1|t}(\cdot \mid x_t)$; evaluate $\tilde{p}_{t,r}(\cdot \mid x_t^{e \to s})$
  Compute $\widehat{C}^d_\theta$ via Equation 19 and Γ^d_θ via Equation 18
  Compute the shifted probability $\hat{q}^d_\theta$ with stop-gradient via Equation 20
  Compute $\mathcal{L}_{\tilde{p}}(\theta)$ via Equation 21
  Update $\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}_{\tilde{p}}(\theta)$
until converged

with

$$\Gamma^d_\theta(x_t, z^d) \;\triangleq\; C^d_\theta(x_t, z^d) + \tilde{p}_{t,r}(z^d \mid x_t) - p_{1|t}(z^d \mid x_t). \tag{18}$$

Writing Equation 17 with $\lambda_t \tilde{p}_{t,r}$ outside the stop-gradient rescales the gradient at each (t, r) by $\lambda_t/\mu_{t,r}$ relative to Equation 10, which leaves the fixed point unchanged. For the linear schedule $\kappa_t = t$ the two coincide, since $\mu_{t,r} = \lambda_t$. The correction therefore consists of exactly two terms: the partial time derivative of the network at fixed r and $\lambda_t \Gamma^d_\theta$, which contains the projected transport term: by Equation 15 and Equation 18, $\lambda_t \mu_{t,r} \Gamma^d_\theta = \Pi^d[Q_t U^\theta_{t,r}](x_t, z^d) + \lambda_t \mu_{t,r} \tilde{p}_{t,r}(z^d \mid x_t)$, where the last term comes from $\partial_t \mu_{t,r}$. At $r = t$ both vanish and Equation 17 reduces to the instantaneous rate $Q^\theta_{t,t}(x_t, z^d) = \lambda_t\, \tilde{p}_{t,t}(z^d \mid x_t)$, which equals $Q_t(x_t, z^d)$ when $\tilde{p}_{t,t} = p_{1|t}$, so the objective is a strict generalization of Equation 4.

C^d_θ is estimated without bias by a single forward pass shared across all coordinates: draw $e \sim \mathrm{Unif}(\{1, \dots, \mathcal{D}\})$ and $s \sim p^e_{1|t}(\cdot|x_t)$, evaluate $\tilde{p}_{t,r}(\cdot \mid x_t^{e \to s})$ once, and set

$$\widehat{C}^d_\theta(x_t, z^d) = \mathcal{D}\left(\tilde{p}_{t,r}\big(z^d \mid x_t^{e \to s}\big) - \tilde{p}_{t,r}\big(z^d \mid x_t\big)\right). \tag{19}$$

Every summand of C^d_θ, as well as $\tilde{p}_{t,r} - p_{1|t}$ and $\partial_t \tilde{p}_{t,r}$ sums to zero over z^d. Therefore, the shifted probability

$$\hat{q}^d_\theta(z^d|x_t) \triangleq \tilde{p}_{t,r}(z^d|x_t) + \mathrm{sg}\{\frac{(t-r)\mu_{t,r}}{\lambda_t}[\partial_t \tilde{p}_{t,r}(z^d|x_t) + \lambda_t \Gamma^d_\theta(x_t, z^d)]\}, \tag{20}$$

defined for all $z^d \in \mathcal{S}$, satisfies $\sum_{z^d \in \mathcal{S}} \hat{q}^d_\theta(z^d \mid x_t) = 1$. Moreover, for $z^d \neq x^d_t$, $\hat{q}^d_\theta(z^d|x_t) = Q^\theta_{t,r}(x_t, z^d)/\lambda_t$. Expanding D_F in Equation 11 (Appendix C.7), its linear part becomes $\lambda_t(1 - \hat{q}^d_\theta(x^d_t|x_t))$, and the objective takes the form of Equation 4 with the posterior replaced by the shifted probability $\hat{q}^d_\theta$,

$$\mathcal{L}_{\tilde{p}}(\theta) = \mathbb{E}_{x_1,(t,r),x_t}\left[\lambda_t \sum_{d=1}^{\mathcal{D}} \Big\{ -\big(1 - \delta_{x_1^d}(x_t^d)\big) \log \hat{q}^d_\theta(x_1^d|x_t) - \hat{q}^d_\theta(x_t^d|x_t) \Big\}\right] + \mathrm{const.} \tag{21}$$

Algorithm 1 summarizes training algorithm. The instantaneous posterior $p_{1|t}$ is supplied by a frozen pretrained denoiser, which is also the model the network is initialized from; the boundary condition $U^\theta_{t,t} = Q_t$ gives the same quantity when no pretrained model is available, since a fraction $1-\rho$ of the training pairs are drawn with $r = t$.

Sampling. Sampling takes one Euler step of the learned per-coordinate average generator $\Pi^d[U^\theta_{t,r}]$ independently for every coordinate per interval. On a grid $0 = \tau_0 < \cdots < \tau_K = 1$, each step queries the network once at $(x_t, t, r) = (x_{\tau_k}, \tau_k, \tau_{k+1})$ and, independently for every coordinate, replaces it by a draw from $\tilde{p}^d_{t,r}(\cdot \mid x_t)$ with probability $\omega_{t,r} = (r-t)\mu_{t,r} = \frac{\kappa_r - \kappa_t}{1 - \kappa_t}$. The detailed sampling procedure is given in Algorithm 2.

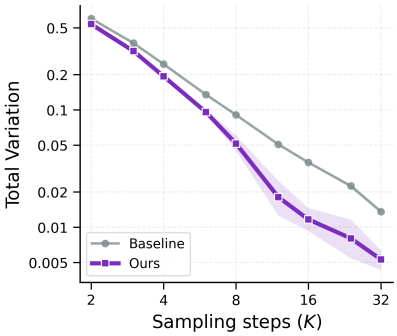

Figure 1: Exact total variation between the target and the law of the K-step sampler on the Potts simulation. Values in Table 2.

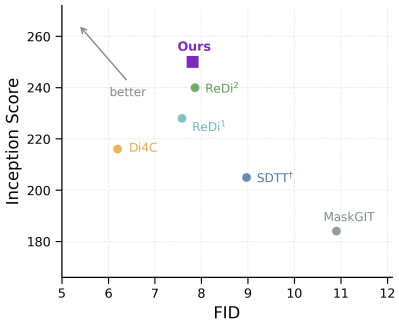

Figure 2: Experimental results on the ImageNet dataset for image generation. Values in Table 4.

4 Experimental Results

4.1 Experimental Setup

Datasets. We evaluate our method on simulated data and two standard benchmarks: OpenWebText (Gokaslan & Cohen, 2019) for text generation and ImageNet (Deng et al., 2009) for image generation. For the simulations, we use a Potts target on $\mathcal{S}^{\mathcal{D}}$ with $|\mathcal{S}| = \mathcal{D} = 4$, a state space small enough to be enumerated exactly. This allows us to evaluate the sampler against the exact ground truth in terms of total variation distance. Details of the data generation procedure are provided in Appendix D.1. For text generation, we use sequences of length 1024, tokenized with the GPT-2 tokenizer (Radford et al., 2019), which has a vocabulary size of 50,257. For image generation, following prior work (Chang et al., 2022; Hayakawa et al., 2025), we tokenize 256×256 images into 16×16 discrete latent codes using a pretrained VQGAN (Esser et al., 2020), resulting in sequences of length 256 with a vocabulary size of 1,024.

Baselines. We compare with SDTT (Deschenaux & Gulcehre, 2025) and Di4C (Hayakawa et al., 2025), which distill a multi-step teacher and, for Di4C, model the correlations between dimensions, and with ReDi (Yoo et al., 2025), which iteratively rectifies the source–data coupling (ReDi^k: k rounds; Di4C+ReDi^1 combines both). Our method does not distill a student from a teacher trajectory.

Implementation. On OpenWebText, we train the model initialized from Di4C+ReDi^1 with Equation 11 under the parameterization Equation 34 and the estimator of Equation 35 (Appendix C.6) for 1k iterations. On ImageNet, we train the model initialized from ReDi^2 with Equation 31 and the single-draw estimator Equation 30 (Appendix C.4) for 3 epochs. Further details can be found in Appendix D.2.

Metrics. For simulated data, we report the total variation distance between the generated distribution and the ground-truth distribution. For text generation, we measure generative perplexity using LLaMA 3.1-8B (Grattafiori et al., 2024) and report entropy to assess diversity and degeneration. For image generation, we generate 50k samples and report Fréchet Inception Distance (FID) (Heusel et al., 2017) and Inception Score (IS) (Salimans et al., 2016).

4.2 Experimental Results

Simulations Figure 1 compares a network trained with the baseline objective Equation 4 with one trained with Equation 21. The correction reduces the total variation for every number of sampling steps K, by 21% at $K = 4$, 43% at $K = 8$ and 67% at $K = 16$, with the same ordering for all three seeds.

Text Generation Figure 3 reports generative perplexity and entropy on OpenWebText. From $K = 8$ on, our method improves on the strongest baseline, Di4C+ReDi^1: 27.59 vs. 30.92 at $K = 8$,

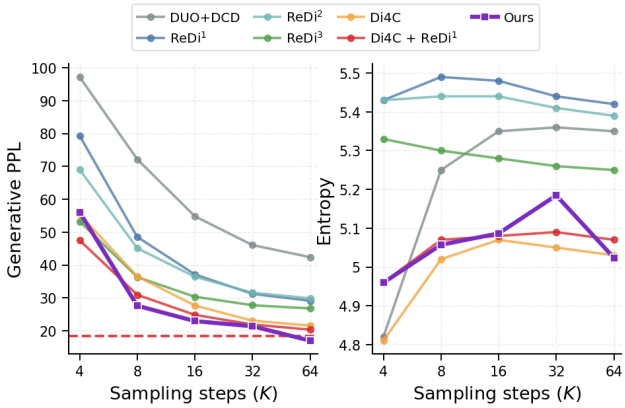

Figure 3: Generative perplexity and entropy on OpenWebText. The dashed horizontal line indicates the PPL of Di4C + ReDi^1 at $K = 1024$ (18.44). Values in Table 3.

22.92 vs. 24.81 at $K = 16$, 21.35 vs. 21.88 at $K = 32$, and 16.97 vs. 20.32 at $K = 64$. Most importantly, our method achieves a lower generative perplexity with $K = 64$ than Di4C+ReDi^1 with $K = 1024$, demonstrating a $16\times$ acceleration. Qualitative inspection of unconditional samples generated at $K = 64$ (Appendix E) further suggests that our method produces fluent, topically coherent text at this reduced step budget, without the repetitive degeneration that can mask a low generative perplexity.

Image Generation Figure 2 compares the methods on ImageNet 256×256 with four network evaluations. Our method attains the highest Inception Score, 250, 4% above the next best (ReDi^2, 240), at a comparable FID (7.81 vs. 7.86). The ablation of classifier-free guidance can be found in Appendix E.1 and qualitative comparisons are provided in Appendix E.2.

5 Related Work

In continuous spaces, sampling is accelerated by higher-order solvers (Song et al., 2020; Lu et al., 2022), caching (Ma et al., 2024), distillation (Salimans & Ho, 2022; Zhou et al., 2024), consistency models (Song et al., 2023), rectified flow (Liu et al., 2022), and methods that learn the solution operator over an interval, including shortcut models (Frans et al., 2025), MeanFlow (Geng et al., 2025a;b; Lu et al., 2026). For discrete diffusion, SDTT (Deschenaux & Gulcehre, 2025) and Di4C (Hayakawa et al., 2025) distill multi-step samplers, ReDi (Yoo et al., 2025) rectifies the coupling, Di[M]O (Zhu et al., 2025) distills a one-step generator by distribution matching, and Fast-dLLM (Wu et al., 2026) and Jacobi Forcing (Hu et al., 2026) accelerate inference. Closest to our work, Categorical Flow Maps (Roos et al., 2026) and Discrete Flow Maps (Potaptchik et al., 2026) learn flow maps of a *continuous* interpolant on the probability simplex. We instead stay on the discrete state space and learn the average generator of a CTMC, so that the transport term is left multiplication by the generator rather than a Jacobian–vector product along a drift, and the construction applies to any CTMC with coordinate-wise rates. Appendix A provides a more detailed discussion of related work.

6 Conclusion

We have introduced the Discrete Average Generator, a principled extension of MeanFlow to discrete state spaces governed by CTMCs. By defining the average generator as the normalized increment of the transition kernel, we derived the Discrete Average Generator Identity and the corresponding training objective. Projected onto per-coordinate marginals, the identity becomes a closed equation whose transport term can be estimated without bias with one additional forward pass. Our method reduces the total variation of the K-step sampler by up to 67% in simulations, attains the lowest generative perplexity of all compared methods on OpenWebText for every budget of 8 to 64 steps with $16\times$ acceleration, and on ImageNet with the highest Inception Score.

AI USE STATEMENT

In this work, we used generative AI tools for designing or providing feedback on research methodology or experiments, assisting in the writing of proofs, and implementing methods. We have not used generative AI tools for generating synthetic data sets, helping develop theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, proposing or refining hypotheses, supporting qualitative and thematic data analysis, interpreting results, and assisting with translation, cleaning and reformat dataset are not applicable to this work. Additionally, we used generative AI tools for suggesting experimental parameters. We have reviewed all AI-assisted work. We checked LLM-generated code and tested for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The code is provided in the supplementary material. The full derivation of the training objective is provided in Appendices B.1 and C.

REFERENCES

Andrew Campbell, Jason Yim, Regina Barzilay, Tom Rainforth, and Tommi Jaakkola. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. In *International Conference on Machine Learning*, 2024.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. Maskgit: Masked generative image transformer. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Justin Deschenaux and Caglar Gulcehre. Beyond autoregression: Fast llms via self-distillation through time. In *International Conference on Learning Representations*, 2025.

Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12868–12878, 2020.

Kevin Frans, Danijar Hafner, Sergey Levine, and Pieter Abbeel. One step diffusion via shortcut models. *International Conference on Learning Representations*, 2025.

Itai Gat, Tal Remez, Neta Shaul, Felix Kreuk, Ricky TQ Chen, Gabriel Synnaeve, Yossi Adi, and Yaron Lipman. Discrete flow matching. In *Advances in Neural Information Processing Systems*, volume 37, pp. 133345–133385, 2024.

Zhengyang Geng, Mingyang Deng, Xingjian Bai, J. Zico Kolter, and Kaiming He. Mean flows for one-step generative modeling. *ArXiv*, abs/2505.13447, 2025a.

Zhengyang Geng, Yiyang Lu, Zongze Wu, Eli Shechtman, J. Zico Kolter, and Kaiming He. Improved mean flows: On the challenges of fastforward generative models. *ArXiv*, abs/2512.02012, 2025b.

Aaron Gokaslan and Vanya Cohen. Openwebtext corpus. `http://Skylion007.github.io/OpenWebTextCorpus`, 2019.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song,

Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca

Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models. *arXiv preprint arXiv: 2407.21783*, 2024.

Satoshi Hayakawa, Yuhta Takida, Masaaki Imaizumi, Hiromi Wakaki, and Yuki Mitsufuji. Distillation of discrete diffusion through dimensional correlations. In *Proceedings of the 42nd International Conference on Machine Learning*, pp. 22259–22297, 2025.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, 2017.

Peter Holderrieth, Marton Havasi, Jason Yim, Neta Shaul, Itai Gat, Tommi Jaakkola, Brian Karrer, Ricky TQ Chen, and Yaron Lipman. Generator matching: Generative modeling with arbitrary markov processes. In *International Conference on Learning Representations*, 2025.

Lanxiang Hu, Siqi Kou, Yichao Fu, Samyam Rajbhandari, Tajana Simunic, Yuxiong He, Zhijie Deng, and Hao Zhang. Fast and accurate causal parallel decoding using jacobi forcing. *International Conference on Machine Learning*, 2026.

Yaron Lipman, Marton Havasi, Peter Holderrieth, Neta Shaul, Matt Le, Brian Karrer, Ricky TQ Chen, David Lopez-Paz, Heli Ben-Hamu, and Itai Gat. Flow matching guide and code. *arXiv preprint arXiv:2412.06264*, 2024.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *ArXiv*, abs/2209.03003, 2022.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *ArXiv*, abs/2206.00927, 2022.

Yiyang Lu, Susie Lu, Qiao Sun, Hanhong Zhao, Zhicheng Jiang, Xianbang Wang, Tianhong Li, Zhengyang Geng, and Kaiming He. One-step latent-free image generation with pixel mean flows. 2026.

Xinyin Ma, Gongfan Fang, and Xinchao Wang. Deepcache: Accelerating diffusion models for free. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. In *International Conference on Learning Representations*, 2025.

Jingyang Ou, Shen Nie, Kaiwen Xue, Fengqi Zhu, Jiacheng Sun, Zhenguo Li, and Chongxuan Li. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. In *International Conference on Learning Representations*, 2025.

Peter Potaptchik, Jason Yim, Adhithya Saravanan, Peter Holderrieth, Eric Vanden-Eijnden, and Michael S. Albergo. Discrete flow maps. *ArXiv*, 2026.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

Daan Roos, Oscar Davis, Floor Eijkelboom, Michael M. Bronstein, Max Welling, .Ismail .Ilkan Ceylan, Luca Ambrogioni, and Jan-Willem van de Meent. Categorical flow maps. *ICML*, 2026.

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

Subham Sekhar Sahoo, Justin Deschenaux, Aaron Gokaslan, Guanghan Wang, Justin T Chiu, and Volodymyr Kuleshov. The diffusion duality. In *International Conference on Machine Learning*, 2025.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

Tim Salimans, Ian J. Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, 2016.

Neta Shaul, Itai Gat, Marton Havasi, Daniel Severo, Anuroop Sriram, Peter Holderrieth, Brian Karrer, Yaron Lipman, and Ricky T. Q. Chen. Flow matching with general discrete paths: a kinetic-optimal perspective. In *International Conference on Learning Representations*, 2025.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis Titsias. Simplified and generalized masked diffusion for discrete data. In *Advances in Neural Information Processing Systems*, volume 37, pp. 103131–103167, 2024.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *ArXiv*, abs/2010.02502, 2020.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *International Conference on Machine Learning*, 2023.

Zhengyan Wan, Yidong Ouyang, Qiang Yao, Liyan Xie, Fang Fang, Hongyuan Zha, and Guang Cheng. Error analysis of discrete flow with generator matching. *arXiv preprint arXiv:2509.21906*, 2025.

Chengyue Wu, Hao Zhang, Shuchen Xue, Zhijian Liu, Shizhe Diao, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Fast-dllm: Training-free acceleration of diffusion llm by enabling kv cache and parallel decoding. *International Conference on Learning Representations*, 2026.

Jaehoon Yoo, Wonjung Kim, and Seunghoon Hong. Redi: Rectified discrete flow. *Advances in Neural Information Processing Systems*, 2025.

Mingyuan Zhou, Huangjie Zheng, Zhendong Wang, Mingzhang Yin, and Hai Huang. Score identity distillation: Exponentially fast distillation of pretrained diffusion models for one-step generation. In *International Conference on Machine Learning*, 2024.

Yuanzhi Zhu, Xi Wang, Stéphane Lathuilière, and Vicky Kalogeiton. Di[m]o: Distilling masked diffusion models into one-step generator. *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18606–18618, 2025.

A Detailed Discussion of Related Work

Acceleration for Continuous Diffusion Models and Flow Matching. DDIM (Song et al., 2020) and DPM-Solver (Lu et al., 2022) leverage deterministic or higher-order ODE solvers to speed up sampling of diffusion models and flow matching. Caching-based methods, such as DeepCache (Ma et al., 2024), reuse intermediate computations across timesteps to further improve efficiency. Another line of approaches focuses on distilling the original multi-step generation process into fewer steps. Progressive distillation (Salimans & Ho, 2022) iteratively compresses the sampling trajectory, while score identity distillation (Zhou et al., 2024) and consistency models (Song et al., 2023) enforce consistency across different noise levels to enable one- or few-step generation. Rectified flow (Liu et al., 2022) provides another perspective by reformulating the generative process as a straightened flow, which naturally reduces the complexity of sampling. Shortcut models (Frans et al., 2025) and MeanFlow (Geng et al., 2025a) introduce tailored modifications to the flow formulation to achieve faster generation. In particular, MeanFlow defines an average velocity field over time intervals and derives a self-consistent fixed-point identity, which is the direct inspiration for our work. Improved MeanFlow (Geng et al., 2025b) further addresses training instability through a velocity-based reparameterization, and pixel MeanFlow (Lu et al., 2026) proposes an x-prediction parameterization that improves empirical stability by directly supervising the clean data prediction.

Acceleration of Diffusion Language Models. Extending acceleration techniques to discrete diffusion models and diffusion language models has attracted increasing attention. SDTT (Deschenaux & Gulcehre, 2025) and Di4C (Hayakawa et al., 2025) accelerate discrete diffusion by distilling multi-step trajectories into fewer steps, where Di4C additionally exploits dimensional correlations to improve distillation quality. ReDi (Yoo et al., 2025) iteratively rectifies the coupling between source and target distributions to reduce the factorization error of discrete flow models, enabling efficient few-step generation. Di[M]O (Zhu et al., 2025) proposes to distill masked diffusion models into a one-step generator via token-level distribution matching in an on-policy framework, and introduces a token initialization strategy to inject stochasticity into the otherwise deterministic initial distribution. On the inference side, Fast-dLLM (Wu et al., 2026) accelerates diffusion LLM inference without additional training by introducing a block-wise approximate KV cache mechanism for bidirectional diffusion models and a confidence-aware parallel decoding strategy. Jacobi Forcing (Hu et al., 2026) takes a complementary approach by converting autoregressive LLMs into causal parallel decoders through progressive distillation on the model's own Jacobi decoding trajectories, preserving the causal backbone and KV cache compatibility while achieving significant wall-clock speedup. In contrast to these approaches, our method directly learns an average generator for CTMCs without multi-step distillation, and addresses the discrete analogue of the MeanFlow fixed-point identity at the level of the transition rate operator.

B Theoretical Details for Section 2

B.1 Proof of Training Objective for Discrete Flow Matching

Proof. We show how the Bregman divergence-based training objective

$$\mathbb{E}\Big[\sum_{z\neq x_t} D_F\big(Q_t(x_t,z|x_1)\,\|\,Q_t(x_t,z)\big)\Big],\quad F(x)=x\log x,$$

reduces to a cross-entropy loss with respect to the posterior $p_{1|t}$ when using the coordinate-wise mixture path.

Step 0: Off-diagonal restriction. The generator $Q_t(x,z|x_1)$ contains a negative diagonal term $-\delta_x(z)$ and is therefore a signed measure. Since the objective only sums over $z\neq x$, we restrict both conditional and marginal generators to their off-diagonal parts:

$$Q_t^{\text{off}}(x,z|x_1) := Q_t(x,z|x_1)\mathbf{1}_{z\neq x}.$$

Under this restriction, all quantities are non-negative and the Bregman divergence is well-defined.

Step 1: Bregman divergence. For $F(x)=x\log x$, the Bregman divergence is

$$D_F(a\|b) = a\log\frac{a}{b} - a + b.$$

Step 2: Conditional and oracle rates. From the coordinate-wise mixture path, for a single coordinate d,

$$Q_{\text{cond}}(z^d) = Q_t^d(x_t^d, z^d|x_1^d)\mathbf{1}_{z^d \neq x_t^d} = \lambda_t \mathbf{1}_{\{x_1^d \neq x_t^d\}}\, \delta_{x_1^d}(z^d),$$

where we used that the original generator contains an additional diagonal term $-\delta_{x_t^d}(z^d)$, which vanishes after restricting to $z^d \neq x_t^d$. The corresponding oracle rate is

$$Q_{\text{oracle}}(z^d) = Q_t^d(x_t^d, z^d) = \lambda_t\, p_{1|t}^d(z^d|x_t), \quad z^d \neq x_t^d.$$

Step 3: Support of the conditional rate. The conditional rate has support only at $z^d = x_1^d$ when $x_1^d \neq x_t^d$, and is identically zero otherwise. Hence the analysis splits into two cases.

Step 4: KL term at the target jump ($x_1^d \neq x_t^d$). For $z^d = x_1^d$:

$$Q_{\text{cond}}(x_1^d) \log \frac{Q_{\text{cond}}(x_1^d)}{Q_{\text{oracle}}(x_1^d)} = \lambda_t \log \frac{\lambda_t}{\lambda_t\, p_{1|t}^d(x_1^d|x_t)} = \lambda_t \big(-\log p_{1|t}^d(x_1^d|x_t)\big).$$

Step 5: Linear terms. The remaining terms in D_F give

$$-Q_{\text{cond}}(x_1^d) + \sum_{z^d \neq x_t^d} Q_{\text{oracle}}(z^d) = -\lambda_t + \lambda_t\big(1 - p_{1|t}^d(x_t^d|x_t)\big).$$

Step 6: Case $x_1^d = x_t^d$. In this case, $Q_{\text{cond}}(z^d) = 0$ for all $z^d \neq x_t^d$. Hence

$$D_F(0\|Q_{\text{oracle}}(z^d)) = Q_{\text{oracle}}(z^d),$$

and summing over $z^d \neq x_t^d$ yields

$$\lambda_t(1 - p_{1|t}^d(x_t^d|x_t)).$$

Step 7: Combine terms. Combining both cases using the indicator $\delta_{x_1^d}(x_t^d)$, the Bregman divergence reduces to

$$\lambda_t \Big\{ -(1 - \delta_{x_1^d}(x_t^d)) \log p_{1|t}^d(x_1^d|x_t) + \delta_{x_1^d}(x_t^d) - p_{1|t}^d(x_t^d|x_t) \Big\}.$$

Summing over all coordinates $d = 1, \ldots, \mathcal{D}$, we obtain exactly the training objective

$$\mathbb{E}\Big[\lambda_t \sum_{d=1}^{\mathcal{D}} \Big\{ -(1 - \delta_{x_1^d}(x_t^d)) \log p_{1|t}^d(x_1^d|x_t) + \delta_{x_1^d}(x_t^d) - p_{1|t}^d(x_t^d|x_t) \Big\}\Big].$$

This completes the proof. □

B.2 Proof of Proposition 3.1 of Campbell et al. (2024)

We include the proof of Proposition 3.1 of Campbell et al. (2024) here for completeness. Let $Q_t(x, z|x_1)$ denote the conditional transition rate from state x to z that generates the conditional probability path $p_{t|1}(\cdot|x_1)$, i.e., it satisfies the Kolmogorov forward equation

$$\partial_t p_{t|1}(x|x_1) = \sum_z Q_t(z, x|x_1)\, p_{t|1}(z|x_1).$$

Taking expectation over $x_1 \sim p_1$, we obtain

$$\begin{aligned}
\partial_t p_t(x) &= \mathbb{E}_{x_1 \sim p_1(x_1)}\big[\partial_t p_{t|1}(x|x_1)\big] \quad \text{(definition of } p_t\text{)} \\
&= \mathbb{E}_{x_1 \sim p_1(x_1)}\big[\sum_z Q_t(z, x|x_1)\, p_{t|1}(z|x_1)\big] \\
&= \sum_z \sum_{x_1} p_1(x_1)\, p_{t|1}(z|x_1)\, Q_t(z, x|x_1) \\
&= \sum_z \sum_{x_1} p_t(z)\, p_{1|t}(x_1|z)\, Q_t(z, x|x_1) \quad \text{(Bayes' rule)} \\
&= \sum_z \mathbb{E}_{x_1 \sim p_{1|t}(\cdot|z)}\big[Q_t(z, x|x_1)\big]\, p_t(z).
\end{aligned}$$

Comparing with the Kolmogorov forward equation

$$\partial_t p_t(x) = \sum_z Q_t(z, x)\, p_t(z),$$

we identify the marginal transition rate as

$$Q_t(z, x) = \mathbb{E}_{x_1 \sim p_{1|t}(\cdot|z)}\big[Q_t(z, x|x_1)\big].$$

Equivalently, reparameterizing the indices to express the transition rate from x to z, we obtain

$$Q_t(x, z) = \mathbb{E}_{x_1 \sim p_{1|t}(\cdot|x)}\big[Q_t(x, z|x_1)\big].$$

This shows that $Q_t(x, z)$ generates the marginal probability path $p_t(x)$.

B.3 Proof of the MeanFlow Identity

The MeanFlow identity was originally proved by Geng et al. (2025a). We reproduce the proof here for completeness.

Let x_t follow the flow defined by $\frac{d}{dt}x_t = v(x_t, t)$. By definition, the MeanFlow velocity is

$$u(x_t, t, r) = \frac{1}{r-t}\int_t^r v(x_s, s)\, ds.$$

Multiplying both sides by $(r - t)$ gives

$$(r-t)\, u(x_t, t, r) = \int_t^r v(x_s, s)\, ds.$$

Taking the total derivative with respect to t and applying the fundamental theorem of calculus yields

$$\frac{d}{dt}\big[(t-r)\, u(x_t, t, r)\big] = v(x_t, t).$$

Expanding the left-hand side using the product rule, we obtain

$$(t-r)\, \frac{d}{dt}u(x_t, t, r) + u(x_t, t, r) = v(x_t, t).$$

Rearranging terms leads to the MeanFlow identity

$$u(x_t, t, r) = v(x_t, t) - (t-r)\, \frac{d}{dt}u(x_t, t, r).$$

Here, the total derivative is given by

$$\frac{d}{dt}u(x_t, t, r) = \partial_t u(x_t, t, r) + v(x_t, t) \cdot \nabla_{x_t} u(x_t, t, r).$$

C Theoretical Details

C.1 Notation and the material derivative of a CTMC

Throughout this appendix, $x^{d\to s}$ denotes the state obtained from x by replacing its d-th coordinate with s, matrices are indexed by states and act on the right on row-vectors of probabilities, and $P_{t\to r}$ denotes the propagator of the CTMC, $P_{t\to r}(x, y) = p_{r|t}(y|x)$, which is the unique solution of

$$\underbrace{\partial_r P_{t\to r} = P_{t\to r}\, Q_r}_{\text{forward}}, \qquad \underbrace{\partial_t P_{t\to r} = -\, Q_t\, P_{t\to r}}_{\text{backward}}, \qquad P_{t\to t} = I. \tag{22}$$

For each coordinate e define the row-stochastic matrix P_t^e that resamples coordinate e from the posterior, $P_t^e(x, x^{e\to s}) \triangleq p_{1|t}^e(s|x)$ for all $s \in \mathcal{S}$ (the diagonal $s = x^e$ included) and $P_t^e(x, y) = 0$ otherwise. The coordinate-wise marginal generator of Section 3 is then exactly

$$Q_t = \lambda_t \sum_{e=1}^{\mathcal{D}} \big(P_t^e - I\big), \qquad U_{t,r}^\theta = \mu_{t,r} \sum_{d=1}^{\mathcal{D}} \big(\tilde{P}_{t,r}^{d,\theta} - I\big), \tag{23}$$

with $\tilde{P}^{d,\theta}_{t,r}(x, x^{d\to s}) = \tilde{p}^{d}_{t,r}(s|x)$. Indeed, for $z^d \neq x^d$ the $(x, x^{d\to z^d})$ entry of Equation 23 is $\lambda_t p_{1|t}(z^d|x)$, the diagonal is $\lambda_t \sum_e (p^e_{1|t}(x^e|x) - 1)$, and all entries at Hamming distance ≥ 2 vanish; in particular Q_t is *single-site supported*.

MeanFlow's identity is obtained by differentiating $(t-r)u(x_t, t, r)$ *along the flow*, which produces the material derivative $\frac{d}{dt} = \partial_t + v \cdot \nabla_x$. The correct discrete counterpart is supplied by Dynkin's formula. Let (X_t) be the CTMC with generator Q_t and let f_t be a time-dependent function on $\mathcal{S}^{\mathcal{D}}$, viewed as a column vector, which is C^1 in t. Then

$$M_t \triangleq f_t(X_t) - \int_0^t \big(\partial_s f_s + Q_s f_s\big)(X_s)\, ds$$

is a martingale, and consequently $\frac{d}{dt}\mathbb{E}[f_t(X_t)] = \mathbb{E}[(\partial_t f_t + Q_t f_t)(X_t)]$. We therefore define the *material derivative* along the CTMC as

$$\frac{D}{Dt} \triangleq \partial_t + Q_t \cdot, \tag{24}$$

where $Q_t \cdot$ denotes left multiplication (applied columnwise for matrix-valued fields). This is the discrete analogue of $v \cdot \nabla_x$: the generator *is* the directional derivative along the process.

C.2 Proof of Proposition 1

We first record the basic properties of $U_{t,r} = \frac{P_{t\to r} - I}{r-t}$ defined in Equation 7.

Proposition 2 (Basic properties of the average generator)**.** *For $0 \leq t < r \leq 1$:* (a) $U_{t,r}(x, z) \geq 0$ *for all $z \neq x$ and $\sum_z U_{t,r}(x, z) = 0$, i.e. $U_{t,r}$ is itself a valid generator;* (b) $\delta_x^\top (I + (r-t)U_{t,r}) = p_{r|t}(\cdot|x)$ *for every x, i.e. a single Euler step of size $r - t$ reproduces the exact transition law;* (c) $\lim_{r\to t} U_{t,r} = Q_t$.

Proof. (a) $P_{t\to r}$ is row-stochastic, so its off-diagonal entries are nonnegative and its rows sum to one; subtracting I and dividing by $r - t > 0$ preserves nonnegativity off the diagonal and yields zero row sums. (b) Immediate from $\delta_x^\top P_{t\to r} = p_{r|t}(\cdot|x)$. (c) $\lim_{r\to t} \frac{P_{t\to r} - I}{r-t} = \partial_r P_{t\to r}|_{r=t} = Q_t$ by Equation 22. □

Proof of Proposition 1. Let $\Omega_{t,r} \triangleq (t-r)\, U_{t,r} = I - P_{t\to r}$. By the backward equation in Equation 22,

$$\partial_t \Omega_{t,r} = -\partial_t P_{t\to r} = Q_t\, P_{t\to r} = Q_t\big(I - \Omega_{t,r}\big) = Q_t - Q_t\, \Omega_{t,r},$$

that is $\partial_t \Omega_{t,r} + Q_t \Omega_{t,r} = Q_t$, which is exactly $\frac{D}{Dt}[(t-r)U_{t,r}] = Q_t$ with the material derivative Equation 24. Expanding the left-hand side with the product rule,

$$\partial_t \Omega_{t,r} = U_{t,r} + (t-r)\, \partial_t U_{t,r}, \qquad Q_t \Omega_{t,r} = (t-r)\, Q_t U_{t,r},$$

and substituting gives $U_{t,r} + (t-r)(\partial_t U_{t,r} + Q_t U_{t,r}) = Q_t$, i.e. Equation 8. The boundary case $r = t$ reduces to $U_{t,t} = Q_t$, consistent with Proposition 2(c). □

C.3 Marginal projection and the tractable projected transport term

The exact $U_{t,r}$ of Equation 7 is *not* single-site supported (a finite-time propagator moves many coordinates at once), whereas the model Equation 23 is. We therefore enforce Equation 8 after projecting onto per-coordinate marginals with Π^d of Equation 13, which is precisely the information a coordinate-factorized sampler consumes.

Proposition 3 (Properties of Π^d)**.** *Let $z^d \neq x^d$. Then* (a) Π^d *is linear and commutes with ∂_t;* (b) *if M is single-site supported then $\Pi^d[M](x, z^d) = M(x, x^{d\to z^d})$, in particular $\Pi^d[Q_t](x, z^d) = \lambda_t p_{1|t}(z^d|x)$ and $\Pi^d[U^\theta_{t,r}](x, z^d) = \mu_{t,r}\tilde{p}_{t,r}(z^d|x)$;* (c) $\Pi^d[I](x, z^d) = 0$ *and $\Pi^d[P^e_t](x, z^d) = \delta_{ed}\, p_{1|t}(z^d|x)$;* (d) $\Pi^d[U_{t,r}](x, z^d) = p^d_{r|t}(z^d|x)/(r-t)$*, so that matching $\Pi^d[U^\theta_{t,r}] = \Pi^d[U_{t,r}]$ is exactly the condition that the coordinate-factorized one-step sampler reproduce the correct per-coordinate marginals of the true jump kernel.*

Proof. (a) is immediate. (b) The only y within Hamming distance one of x with $y^d = z^d \neq x^d$ is $y = x^{d \to z^d}$. (c) Apply (b) to I (supported on $y = x$, which has $y^d = x^d \neq z^d$) and to P_t^e (supported on $y = x^{e \to s}$, which requires $e = d$ and $s = z^d$). (d) Follows from Equation 7 and Equation 13. □

Applying Π^d to Equation 8 and using Proposition 3(a)–(b) gives the projected identity we train against: for every d and every $z^d \neq x^d$,

$$\Pi^d\big[U_{t,r}\big](x, z^d) = \lambda_t \, p_{1|t}(z^d|x) - (t - r)\Big(\partial_t \, \Pi^d\big[U_{t,r}\big](x, z^d) + \Pi^d\big[Q_t U_{t,r}\big](x, z^d)\Big). \tag{25}$$

Only the projected transport term couples different coordinates. This is not because Π^d fails to commute with Q_t: since $\Pi^d[M] = ME_d$ acts on the target state, it commutes with left multiplication, $\Pi^d[Q_t M] = Q_t \, \Pi^d[M]$, as made explicit in Equation 26. Rather, Q_t acts on the source state, so $Q_t \, \Pi^d[U_{t,r}]$ evaluates $\Pi^d[U_{t,r}]$ at the states $x^{e \to s}$ reached by a jump of any coordinate e, not only of d. The next proposition evaluates it in closed-form for the single-site model, which is what makes the objective tractable and proves Equation 15.

Proposition 4 (Tractable projected transport term)**.** *Let Q_t and $U_{t,r}^\theta$ be as in Equation 23. Then for every d and every $z^d \neq x^d$,*

$$\Pi^d\big[Q_t U_{t,r}^\theta\big](x, z^d) = \lambda_t \, \mu_{t,r}\Big[\, C_\theta^d(x, z^d) - p_{1|t}(z^d \mid x)\Big],$$

with C_θ^d as in Equation 16.

Proof. Expanding the product of the two expressions in Equation 23,

$$Q_t U_{t,r}^\theta = \lambda_t \mu_{t,r} \sum_{e=1}^{\mathcal{D}} \sum_{d'=1}^{\mathcal{D}} \Big(P^e \tilde{P}^{d'} - P^e - \tilde{P}^{d'} + I\Big),$$

where we drop the sub/superscripts t, r, θ for readability. Fix d and $z^d \neq x^d$ and apply Π^d term by term.

Step 1. By Proposition 3(c), $\Pi^d[I] = 0$, $\sum_{e,d'} \Pi^d[P^e](x, z^d) = \mathcal{D} \, p_{1|t}(z^d|x)$ and $\sum_{e,d'} \Pi^d[\tilde{P}^{d'}](x, z^d) = \mathcal{D} \, \tilde{p}_{t,r}(z^d|x)$, each sum contributing a factor $\mathcal{D}$ from the free index.

Step 2. For the products, write $(P^e \tilde{P}^{d'})(x, y) = \sum_s P^e(x, x^{e \to s}) \, \tilde{P}^{d'}(x^{e \to s}, y)$, so that y must be of the form $y = x^{e \to s,\, d' \to s'}$. Requiring $y^d = z^d \neq x^d$ forces $d \in \{e, d'\}$ with the corresponding new symbol equal to z^d. Two cases: *(i)* $d' = d$: then $s' = z^d$ and the remaining index s is free and summed over by Π^d, giving, for every e (including $e = d$), $\sum_s p_{1|t}^e(s|x) \tilde{p}_{t,r}(z^d|x^{e \to s}) = \mathbb{E}_{s \sim p_{1|t}^e(\cdot|x)}[\tilde{p}_{t,r}(z^d|x^{e \to s})]$. *(ii)* $d' \neq d$: then coordinate d can only be set by the first factor, so $e = d$ and $s = z^d$; the index s' is free and summed by Π^d, and since $\tilde{P}^{d'}$ is row-stochastic, $\sum_{s'} \tilde{p}_{t,r}(s'|x^{d \to z^d}) = 1$. Each of the $\mathcal{D} - 1$ values of d' contributes $p_{1|t}(z^d|x)$. Summing,

$$\sum_{e,d'} \Pi^d\big[P^e \tilde{P}^{d'}\big](x, z^d) = \sum_e \mathbb{E}_{s \sim p_{1|t}^e(\cdot|x)}\big[\tilde{p}_{t,r}(z^d|x^{e \to s})\big] + (\mathcal{D} - 1) \, p_{1|t}(z^d|x).$$

Step 3. Combining Steps 1 and 2,

$$\Pi^d\big[Q_t U^\theta\big](x, z^d) = \lambda_t \mu_{t,r}\Big[\sum_e \mathbb{E}_{s \sim p_{1|t}^e(\cdot|x)}\big[\tilde{p}_{t,r}(z^d|x^{e \to s})\big] - \mathcal{D} \, \tilde{p}_{t,r}(z^d|x) - p_{1|t}(z^d|x)\Big],$$

which is the claim after grouping the $\mathcal{D}$ copies of $\tilde{p}_{t,r}(z^d|x)$ into the $\mathcal{D}$ summands of Equation 16. □

What the objective learns. The projected identity Equation 25 is closed in the per-coordinate marginals. For a matrix M with zero row sums, extend $\Pi^d[M](x, \cdot)$ to $z^d = x^d$ by $\Pi^d[M](x, x^d) \triangleq \sum_{y:\, y^d = x^d} M(x, y) = -\sum_{s \neq x^d} \Pi^d[M](x, s)$. Exchanging the sums in Equation 13 gives

$$\Pi^d\big[Q_t M\big](x, z^d) = \sum_{y:\, y^d = z^d} \sum_{x'} Q_t(x, x') \, M(x', y) = \sum_{x'} Q_t(x, x') \, \Pi^d[M](x', z^d), \tag{26}$$

so the projected transport term depends on M only through its off-diagonal per-coordinate marginals at the states x' reachable from x in one jump; Equation 35 is Equation 26 written out for a single-site M. In particular $\Pi^d[Q_t U^\theta_{t,r}] = \Pi^d[Q_t U_{t,r}]$ whenever $\Pi^d[U^\theta_{t,r}] = \Pi^d[U_{t,r}]$ for all d and all states, so replacing $U_{t,r}$ by the single-site model in the projected transport term is not an approximation.

Proposition 5 (Fixed point of the projected identity)**.** *Fix r and assume that $t \mapsto Q_t$ is continuous on $[t_0, r]$. Let $V_t(x, z^d)$, $z^d \neq x^d$, be per-coordinate rates that are C^1 in t and bounded on $[t_0, r)$, extended to $z^d = x^d$ by zero row sums, and suppose that for all $t \in [t_0, r)$, x, d and $z^d \neq x^d$,*

$$V_t(x, z^d) = \Pi^d[Q_t](x, z^d) - (t - r)\Big(\partial_t V_t(x, z^d) + \sum_{x'} Q_t(x, x')\, V_t(x', z^d)\Big). \tag{27}$$

Then $V_t = \Pi^d[U_{t,r}]$, i.e. $V_t(x, z^d) = p^d_{r|t}(z^d|x)/(r - t)$.

Proof. By Equation 26, Equation 25 states that $\Pi^d[U_{t,r}]$, which is bounded by Proposition 2(c), satisfies Equation 27. For uniqueness, let $W_t \triangleq (t - r)V_t$. Since $\partial_t W_t = V_t + (t - r)\partial_t V_t$, Equation 27 is equivalent to the linear system $\partial_t W_t = \Pi^d[Q_t] - \mathcal{Q}_t W_t$ with $(\mathcal{Q}_t W)(x, z^d) \triangleq \sum_{x'} Q_t(x, x')W(x', z^d)$, in the finitely many off-diagonal entries of W_t (the diagonal ones being fixed by zero row sums). Boundedness of V_t gives $W_t \to 0$ as $t \to r$. A linear ODE with continuous coefficients has a unique solution with a prescribed terminal value, so W_t, and hence V_t, is unique. □

At the population optimum of Equation 11, a model that can match the marginal rate satisfies $Q^\theta_{t,r}(x_t, z^d) = \Pi^d[Q_t](x_t, z^d)$, the minimizer of the Bregman divergence to the conditional rate. By Equation 10 and Equation 26 this is Equation 27 for $V_t = \Pi^d[U^\theta_{t,r}]$, so $\Pi^d[U^\theta_{t,r}] = \Pi^d[U_{t,r}]$: the objective learns the per-coordinate marginals of the exact finite-step kernel. Regressing $\Pi^d[U^\theta_{t,r}]$ directly onto $\Pi^d[U_{t,r}]$ has the same solution, but requires samples of the kernel over $[t, r]$, i.e. multi-step trajectories of a teacher; the projected identity only requires the instantaneous rate and single-coordinate perturbations of the current state.

C.4 The masked path

For the masked source $p^d_0 = \delta_{\mathbf{m}}$, with $\mathbf{m}$ the mask token and $\mathcal{M}(x)$ the set of masked coordinates of x, the posterior and the network satisfy the carry-over property $p^e_{1|t}(\cdot|x) = \tilde{p}^e_{t,r}(\cdot|x) = \delta_{x^e}$ for $e \notin \mathcal{M}(x)$, and neither predicts $\mathbf{m}$. If $e \notin \mathcal{M}(x)$ then $x^{e\to s} = x$ almost surely and the e-th summand of Equation 16 vanishes; if $e = d \in \mathcal{M}(x)$, carry-over gives $\tilde{p}^d_{t,r}(\cdot|x^{d\to s}) = \delta_s$ and $\mathbb{E}_s[\delta_s(z^d)] = p_{1|t}(z^d|x)$. Hence

$$C^d_\theta(x_t, z^d) = \big(p_{1|t}(z^d|x_t) - \tilde{p}_{t,r}(z^d|x_t)\big) + C^d_{\text{cross}}(x_t, z^d), \tag{28}$$

with $C^d_{\text{cross}}(x_t, z^d) \triangleq \sum_{e\in\mathcal{M}(x_t)\setminus\{d\}} \big(\mathbb{E}_{s\sim p^e_{1|t}(\cdot|x_t)}[\tilde{p}_{t,r}(z^d|x^{e\to s}_t)] - \tilde{p}_{t,r}(z^d|x_t)\big)$, so that Γ^d_θ of Equation 18, which contains the projected transport term, reduces to C^d_{cross} and Equation 17 becomes

$$Q^\theta_{t,r}(x_t, z^d) = \lambda_t\, \tilde{p}_{t,r}(z^d \mid x_t) + \text{sg}\left\{(t - r)\,\mu_{t,r}\Big[\partial_t \tilde{p}_{t,r}(z^d \mid x_t) + \lambda_t\, C^d_{\text{cross}}(x_t, z^d)\Big]\right\}. \tag{29}$$

Since unmasked coordinates contribute nothing, C^d_{cross} is estimated with a lower variance than Equation 19 by drawing $e \sim \text{Unif}(\mathcal{M}(x_t))$ and $s \sim p^e_{1|t}(\cdot|x_t)$ and setting

$$\widehat{C}^d_{\text{cross}}(x_t, z^d) = \big|\mathcal{M}(x_t)\big| \cdot \mathbb{1}[e \neq d] \cdot \Big(\tilde{p}_{t,r}\big(z^d \mid x^{e\to s}_t\big) - \tilde{p}_{t,r}\big(z^d \mid x_t\big)\Big). \tag{30}$$

For an unmasked d the shifted probability is $\hat{q}^d_\theta = \delta_{x^d_t}$, and for a masked d it satisfies $\hat{q}^d_\theta(\mathbf{m}|x_t) = 0$, so Equation 21 reduces to a weighted cross-entropy over the masked coordinates,

$$\mathcal{L}_{\tilde{p}}(\theta) = \mathbb{E}_{x_1,(t,r),x_t}\left[\lambda_t \sum_{d\in\mathcal{M}(x_t)} -\log\left(\tilde{p}_{t,r}(x^d_1 \mid x_t) + \text{sg}\left\{\tfrac{(t-r)\mu_{t,r}}{\lambda_t}\Big[\partial_t \tilde{p}_{t,r}(x^d_1 \mid x_t) + \lambda_t\, C^d_{\text{cross}}(x_t, x^d_1)\Big]\right\}\right)\right] + \text{const.} \tag{31}$$

This is the objective we use on ImageNet.

Algorithm 2 Sampling with the Discrete Average Generator

Require: Network H^θ, schedule κ_t, step count K, grid $0 = \tau_0 < \tau_1 < \cdots < \tau_K = 1$

$x_{\tau_0} \sim p_0$ ▷ uniform noise; all-mask state for the masked source
for $k = 0, \dots, K-1$ **do**
  $t \leftarrow \tau_k, \quad r \leftarrow \tau_{k+1}$ ▷ a single jump of size $r - t$
  $\tilde{p}_{t,r}(\cdot \mid x_t) \leftarrow H^\theta(x_t, t, r)$ ▷ one NFE, $\mathcal{D} \times |\mathcal{S}|$
  $\omega_{t,r} \leftarrow (r-t)\,\mu_{t,r} = \dfrac{\kappa_r - \kappa_t}{1 - \kappa_t}$ ▷ $\omega_{t,r} \in [0,1]$ by monotonicity of κ
  for each coordinate $d = 1, \dots, \mathcal{D}$ **independently do**
    $x_r^d \leftarrow z^d \sim \tilde{p}_{t,r}^d(\cdot \mid x_t)$ with probability $\omega_{t,r}$, else $x_r^d \leftarrow x_t^d$ ▷ masked source: unmasked x_t^d carried over
  end for
end for
return x_{τ_K}

C.5 Sampling with the Discrete Average Generator

Sampling. Algorithm 2 takes, independently for every coordinate, one Euler step of the learned per-coordinate average generator; its one-step kernel is the product

$$K_{t,r}^\theta(x_t, y) \;=\; \prod_{d=1}^{\mathcal{D}} \Big[\delta_{x_t^d}(y^d) + (r-t)\,\Pi^d\big[U_{t,r}^\theta\big](x_t, y^d)\Big], \tag{32}$$

whose d-th factor is a distribution over $\mathcal{S}$, its entry at $y^d = x_t^d$ being fixed by the zero row sum of $\Pi^d[U_{t,r}^\theta]$. This is not the full Euler step $\delta_{x_t}^\top(I + (r-t)U_{t,r}^\theta)$: since $U_{t,r}^\theta$ is single-site supported, the latter moves at most one coordinate and, for large $\omega_{t,r}$, has a negative diagonal entry. In Equation 32 the per-coordinate jump probability is $(r-t)\,\Pi^d[U_{t,r}^\theta](x_t, z^d) = (r-t)\,\mu_{t,r}\,\tilde{p}_{t,r}(z^d|x_t) = \omega_{t,r}\,\tilde{p}_{t,r}(z^d|x_t)$. Three properties follow directly from Section 3 and are worth making explicit.

(i) Exact marginals. By Proposition 3(d), if $\Pi^d[U_{t,r}^\theta] = \Pi^d[U_{t,r}]$ for every d, then the d-th factor of Equation 32 is exactly the d-th marginal of $p_{r|t}(\cdot|x_t)$, for any step size: no discretisation error is incurred by the per-coordinate jump. The remaining error of a single step is the dependence between the coordinates under $p_{r|t}(\cdot|x_t)$, which a product kernel cannot represent. Proposition 5 shows that these exact marginals are the population target of our objective.

(ii) Train/test consistency. The factor $\mu_{t,r}$ appearing in the sampler is the same one used to parameterise $U_{t,r}^\theta$ during training, so the jump probability $\omega_{t,r}$ is by construction the quantity the training objective was written for. Had we instead kept the instantaneous scale, $U^\theta = \lambda_t \tilde{p}_{1|t}$, the sampler would use $(r-t)\lambda_t$ while the conditional path jumps with probability $\frac{\kappa_r - \kappa_t}{1-\kappa_t}$; these agree only for the linear schedule $\kappa_t = t$, and diverge as $O((r-t)^2 \ddot{\kappa}_t)$ otherwise. Carrying $\mu_{t,r}$ explicitly removes this mismatch for any schedule.

(iii) Where the learning shows up. The product in Equation 32 is over coordinates, so a single step cannot reproduce the correlations of the true kernel $P_{t\to r}$. What the objective can improve is therefore the per-coordinate marginals. We do not claim a guarantee on how much exact marginals reduce the error of a K-step sampler; we measure this exactly in Section 4.2. Setting $r = t$ makes $\omega_{t,r} \to 0$ and recovers the instantaneous rate, so Algorithm 2 degenerates gracefully to the standard ancestral sampler as K grows.

Finally, the network must be queried with the same (t, r) conditioning it saw in training: a K-step run with a uniform grid only ever presents step sizes $r - t = 1/K$, so the training distribution over (t, r) has to place mass on the step sizes that will actually be used at test time.

C.6 The uniform-state (DUO) process

Everything above is stated for the mixture path of Equation 2, whose conditional rate $Q_t^d(x^d, z^d|x_1^d) = \lambda_t(\delta_{x_1^d}(z^d) - \delta_{x^d}(z^d))$ does not depend on the source p_0. For a uniform source this is the CTMC in which a single noise sample x_0 is drawn once and each coordinate either still shows x_0 or has already

switched to x_1. Uniform-state models in the DUO family (Sahoo et al., 2025) use a *different* CTMC with the same marginals $p_{t|1}$: the noise is re-randomised at every step, as in D3PM. This appendix records the parameterization we use on top of the uniform-state checkpoints of our text experiments.

Throughout this subsection we follow the convention of DUO, in which α_t decreases with t, $t = 0$ is data, a step goes from t to $r < t$, and $c' \triangleq (1 - \alpha_t)/|\mathcal{S}|$. Write $\alpha_{t|s} = \alpha_t/\alpha_s$.

Conditional and marginal rates. The conditional rate is obtained by differentiating the bridge $q(x_s|x_t, x_1) \propto \big[\alpha_{t|s}\delta_{x_s}(x_t) + \frac{1-\alpha_{t|s}}{|\mathcal{S}|}\big]\big[\alpha_s\delta_{x_1}(x_s) + \frac{1-\alpha_s}{|\mathcal{S}|}\big]$ at $s = t$:

$$Q_t(x_t, z \,|\, x_1) = \frac{-\dot\alpha_t}{\alpha_t|\mathcal{S}|} \cdot \frac{\alpha_t \mathbb{1}[z = x_1] + c'}{\alpha_t \mathbb{1}[x_1 = x_t] + c'}, \qquad z \neq x_t. \tag{33}$$

Averaging Equation 33 over $x_1 \sim p_{1|t}(\cdot|x_t)$ — which by Proposition 3.1 of Campbell et al. (2024) is the marginal generator — splits on $x_1 \in \{z, x_t, \text{other}\}$ and yields the closed-form $\frac{-\dot\alpha_t}{\alpha_t|\mathcal{S}|}\big[p_z\frac{\alpha_t+c'}{c'} + p_{x_t}\frac{c'}{\alpha_t+c'} + (1 - p_z - p_{x_t})\big]$. Unlike on the mixture path, this rate is not proportional to the posterior mass p_z of z, so the parameterization $U^\theta_{t,r} = \mu_{t,r}\tilde p_{t,r}$ does not carry over.

Parameterisation. We take $U^\theta_{t,r}$ from the exact marginalization of the bridge,

$$\Pi^d\big[U^\theta_{t,r}\big](x_t, z^d) = \frac{1}{t-r}\,\mathbb{E}_{v\sim\tilde p^d_{t,r}(\cdot|x_t)}\big[q(x^d_r = z^d \,|\, x^d_t,\, x^d_1 = v)\big], \qquad z^d \neq x^d_t, \tag{34}$$

which, because the bridge is a ratio of two functions affine in the one-hot encoding of v whose denominator takes only two values, has a closed-form costing $O(|\mathcal{S}|)$ per coordinate. The sampler draws each coordinate from the same kernel, so a single per-coordinate Euler step of Equation 34 is exactly the sampler step.

The projected transport term in general form. Proposition 4 is a special case of an identity that holds for *any* pair of single-site matrices. Let $q^e(x, \cdot)$ and $u^d(x, \cdot)$ denote the coordinate-e row of Q_t and the coordinate-d row of $U^\theta_{t,r}$, both with zero row sum. Then

$$\begin{aligned}\Pi^d\big[Q_tU^\theta_{t,r}\big](x, z^d) &= \sum_{e=1}^{\mathcal{D}}\sum_s q^e(x, s)\, u^d\big(x^{e\to s}, z^d\big)\\ &= \sum_{e=1}^{\mathcal{D}}\Lambda^e(x)\Big(\mathbb{E}_{s\sim\nu^e}\big[u^d(x^{e\to s}, z^d)\big] - u^d(x, z^d)\Big),\end{aligned} \tag{35}$$

with $\Lambda^e(x) = \sum_{s\neq x^e} q^e(x, s)$ the total outflow rate at coordinate e and $\nu^e = q^e/\Lambda^e$ the normalised jump law. The terms with $d' \neq d$ in the expansion vanish because $u^{d'}$ has zero row sum. Substituting the mixture-path rows recovers $\lambda_t\mu_{t,r}(C^d_\theta - p_{1|t})$ of Equation 15; with the uniform-state rows, one draw $e \sim \mathrm{Unif}(\{1, \dots, \mathcal{D}\})$, $s \sim \nu^e$ gives the unbiased estimate $\mathcal{D}\,\Lambda^e(x)\big(u^d(x^{e\to s}, z^d) - u^d(x, z^d)\big)$ at the cost of a single forward pass. The objective is the Bregman divergence Equation 11 between Equation 33 and the reparameterized rate, summed over all $z^d \neq x^d_t$.

Sampling. The sampler draws each coordinate from the kernel Equation 34, i.e. it takes one per-coordinate Euler step of the learned average generator. Two conditioning details matter for a model trained this way. First, the final noise-removal step sets $\alpha_r = 1$, i.e. $r = 0$, so its effective step size is the whole of t rather than the grid spacing; passing no step size at all there queries a conditioning path the network never sees in training. Second, a K-step run on a uniform grid only ever presents $t - r = 1/K$, so the training distribution over (t, r) must place mass there; the logit-normal design over (t, r) used in our experiments places mass on the step sizes $1/K$ of all budgets $K \leq 64$ that we evaluate.

Finally, nucleus filtering and softmax temperature must be left at their neutral settings: both reshape the kernel after the fact and would break the calibration that Equation 34 relies on.

C.7 Proof of Training Objective for Discrete Average Generator

Proof. We show how the objective

$$\mathcal{L}(\theta) = \mathbb{E}\left[\sum_{d=1}^{\mathcal{D}} \sum_{z^d \neq x_t^d} D_F\Big(Q_t(x_t, z^d|x_1) \,\Big\|\, \Pi^d\big[U_{t,r}^\theta\big](x_t, z^d) + \mathrm{sg}\Big((t-r)\Big(\partial_t \Pi^d\big[U_{t,r}^\theta\big](x_t, z^d) + \Pi^d\big[Q_t\, U_{t,r}^\theta\big](x_t, z^d)\Big)\Big)\Big)\right],$$

with $\Pi^d[U_{t,r}^\theta](x_t, z^d) = \mu_{t,r}\, \tilde{p}_{t,r}(z^d|x_t)$, reduces to Equation 36 and then to Equation 21.

Step 1: Bregman divergence. For $F(x) = x \log x$, the Bregman divergence is

$$D_F(a\|b) = a \log \frac{a}{b} - a + b.$$

Step 2: Conditional and target rates. For a single coordinate d, restricting to off-diagonal entries $z^d \neq x_t^d$, the conditional rate is

$$Q_{\mathrm{cond}}(z^d) = Q_t^d(x_t^d, z^d|x_1^d)\, \mathbf{1}_{z^d \neq x_t^d} = \lambda_t\, \mathbf{1}_{\{x_1^d \neq x_t^d\}}\, \delta_{x_1^d}(z^d).$$

The target rate is

$$Q_\theta(z^d) = \Pi^d\big[U_{t,r}^\theta\big](x_t, z^d) + A_t(x_t, z^d) = \mu_{t,r}\, \tilde{p}_{t,r}(z^d|x_t) + A_t(x_t, z^d),$$

where

$$A_t(x_t, z^d) \triangleq \mathrm{sg}\Big((t-r)\Big(\partial_t \Pi^d\big[U_{t,r}^\theta\big](x_t, z^d) + \Pi^d\big[Q_t\, U_{t,r}^\theta\big](x_t, z^d)\Big)\Big).$$

Step 3: Support of the conditional rate. The conditional rate has support only at $z^d = x_1^d$ when $x_1^d \neq x_t^d$, and is identically zero otherwise. The analysis splits into two cases.

Step 4: Log term ($x_1^d \neq x_t^d$). At $z^d = x_1^d$:

$$Q_{\mathrm{cond}}(x_1^d) \log \frac{Q_{\mathrm{cond}}(x_1^d)}{Q_\theta(x_1^d)} = \lambda_t \log \lambda_t - \lambda_t \log Q_\theta(x_1^d),$$

whose first term does not depend on θ.

Step 5: Linear terms ($x_1^d \neq x_t^d$). The linear terms from D_F at $z^d = x_1^d$ are

$$-Q_{\mathrm{cond}}(x_1^d) + Q_\theta(x_1^d) = -\lambda_t + Q_\theta(x_1^d).$$

For all $z^d \neq x_1^d$ (and $z^d \neq x_t^d$), $Q_{\mathrm{cond}}(z^d) = 0$, so $D_F(0\|Q_\theta(z^d)) = Q_\theta(z^d)$. Summing over all $z^d \neq x_t^d$ gives

$$\sum_{z^d \neq x_t^d} Q_\theta(z^d) = \mu_{t,r}\big(1 - \tilde{p}_{t,r}(x_t^d|x_t)\big) + \sum_{z^d \neq x_t^d} A_t(x_t, z^d).$$

The last term cannot in general be rewritten as $-A_t(x_t, x_t^d)$: that step would require $\sum_{z^d} A_t(x_t, z^d) = 0$, whereas $\sum_{z^d} \partial_t \Pi^d[U_{t,r}^\theta](x_t, z^d) = \partial_t \mu_{t,r} \neq 0$. We therefore keep the linear contribution as $\sum_{z^d \neq x_t^d} Q_\theta(z^d)$. For the rate Equation 17 with a normalized $\tilde{p}_{t,r}$ it takes a simple form: since $\sum_{z^d} \tilde{p}_{t,r}(z^d|x_t) = 1$ for all t implies $\sum_{z^d} \partial_t \tilde{p}_{t,r}(z^d|x_t) = 0$, and each summand of C_θ^d, as well as $\tilde{p}_{t,r} - p_{1|t}$, is a difference of two normalized distributions, the correction sums to zero over $z^d \in \mathcal{S}$ and

$$\sum_{z^d \neq x_t^d} Q_\theta(z^d) = \lambda_t \sum_{z^d \neq x_t^d} \hat{q}_\theta^d(z^d|x_t) = \lambda_t\big(1 - \hat{q}_\theta^d(x_t^d|x_t)\big),$$

with $\hat{q}_\theta^d$ the shifted probability of Section 3.3. On the masked path, $\hat{q}_\theta^d(\mathbf{m}|x_t) = 0$ for masked d, so this is the constant λ_t.

Combining the linear contributions in this case gives

$$-Q_{\mathrm{cond}}(x_1^d) + \sum_{z^d \neq x_t^d} Q_\theta(z^d) = -\lambda_t + \sum_{z^d \neq x_t^d} Q_\theta(z^d),$$

where the leading $-\lambda_t$ is independent of θ and is absorbed into the additive constant of the loss.

Step 6: Case $x_1^d = x_t^d$. In this case $Q_{\text{cond}}(z^d) = 0$ for all $z^d \neq x_t^d$, so $D_F(0\|Q_\theta(z^d)) = Q_\theta(z^d)$ and the total contribution is $\sum_{z^d \neq x_t^d} Q_\theta(z^d)$, i.e. the same linear expression as in Step 5 with the logarithmic term absent and without the $-\lambda_t$ offset.

Step 7: Combine terms. Combining both cases using the indicator $\delta_{x_1^d}(x_t^d)$, dropping the θ-independent terms (the $a \log a$ part of D_F and the offset $-\lambda_t(1 - \delta_{x_1^d}(x_t^d))$), and summing over all coordinates $d = 1, \ldots, \mathcal{D}$, we obtain

$$\mathcal{L}_Q(\theta) = \mathbb{E}_{x_1,(t,r),x_t}\left[\sum_{d=1}^{\mathcal{D}}\left\{ -\big(1 - \delta_{x_1^d}(x_t^d)\big)\,\lambda_t\,\log Q_{t,r}^\theta(x_t, x_1^d) + \sum_{z^d \neq x_t^d} Q_{t,r}^\theta(x_t, z^d)\right\}\right] + \text{const}, \tag{36}$$

where $Q_{t,r}^\theta(x_t, z^d) = Q_\theta(z^d)$ is the reparameterized instantaneous rate of Equation 10. Substituting Equation 17, writing $\log Q_{t,r}^\theta(x_t, x_1^d) = \log \lambda_t + \log \hat{q}_\theta^d(x_1^d|x_t)$ and using the linear term above yields Equation 21, and on the masked path Equation 31. This completes the proof. □

D Experimental Details

D.1 Experimental Setup of Simulations

Data distribution We consider a data distribution in which every pair of coordinates is coupled, i.e. we take the Potts law

$$q(x) \;\propto\; \exp\Big(\beta \sum_{1 \leq i < j \leq \mathcal{D}} \mathbb{1}\big[x^i = x^j\big] \;+\; \varepsilon_x\Big), \qquad \varepsilon_x \sim 0.3\,\mathcal{N}(0,1),$$

where $\beta \geq 0$ controls how strongly the coordinates are coupled, and the perturbations ε_x are drawn once and then held fixed so as to break the permutation symmetry of the energy. We use $\beta = 1.5$ and dimension $\mathcal{D} = 4$. The vocabulary size is 4.

Experimental setup and implementation details. The state space has $|\mathcal{S}|^{\mathcal{D}} = 256$ elements, so the target q, the posterior $p_{1|t}$, the projected transport term Equation 15 and the law of the K-step sampler are all obtained by enumeration, i.e., no quantity we report is estimated from samples. We parameterize the network as MLPs with SiLU activation functions, consisting of four hidden layers with hidden dimension 256, conditioned on Fourier features of t, r, $r - t$ and $-\log(1 - t)$. We first train the base denoiser with Equation 4 using the Adam optimizer with a learning rate of 2e-3 for 1.2e4 steps, and then train only the correction head with Equation 21 for 3e3 steps with the base frozen, so that every objective we compare shares an identical base denoiser. The pairs (t, r) are drawn from the evaluation grids, with 25% of the draws at $r = t$; the instantaneous posterior $p_{1|t}^e$ is supplied by the frozen base denoiser. In the sampling stage, we evaluate the trained network at every state and propagate the law of the sampler exactly, for a number of steps $K \in \{2, 3, 4, 6, 8, 12, 16, 24, 32\}$, and record the total variation between q and that law. We run each setting with 3 seeds and report the mean.

D.2 Implementation Details of OpenWebText and ImageNet

The network is conditioned on t and on the step size $r - t$. On OpenWebText, τ_t and τ_r are the larger and the smaller of two logit-normal draws $\sigma(\mathcal{N}(P_{\text{mean}}, P_{\text{std}}^2))$ with $(P_{\text{mean}}, P_{\text{std}}) = (0, 1.5)$ and $(-1.0, 1.5)$; on ImageNet, τ_t is the arccos mask ratio and τ_r is a logit-normal draw with $(-3.0, 2.0)$ clipped to $\tau_r \leq \tau_t$, which places more mass on the large steps of the 1–4 step regime. A fraction $1 - \rho = 0.25$ of the pairs is drawn with $r = t$, where the correction vanishes and the objective reduces to Equation 4; this is what keeps the boundary calibrated. On ImageNet the sampling hyperparameters of every method, ours and the baselines, are selected by a sweep and we report the best configuration of each method; on OpenWebText the baseline numbers are taken from Yoo et al. (2025).

D.3 Detailed values for Section 4

Detailed values for simulations, OpenWebText, and ImageNet can be found in Tables 2, 3, and 4, respectively.

Table 2: Total variation Potts model simulations. Mean over three seeds. These are the values plotted in Figure 1.

K	2	3	4	6	8	12	16	24	32
Baseline	0.6020	0.3707	0.2456	0.1351	0.0907	0.0508	0.0356	0.0225	0.0136
Ours	**0.5381**	**0.3170**	**0.1934**	**0.0957**	**0.0517**	**0.0181**	**0.0116**	**0.0080**	**0.0053**

Table 3: OpenWebText results for Figure 3: generative perplexity measured with LLaMA 3.1-8B and entropy, as a function of the number of sampling steps K. Baseline numbers are taken from Yoo et al. (2025). Best perplexity per column in bold.

	Sampling steps K				
Model	4	8	16	32	64
Generative perplexity					
DUO+DCD	97.22	72.18	54.82	46.05	42.38
ReDi[1]	79.35	48.53	37.13	31.21	29.12
ReDi[2]	69.01	45.11	36.42	31.56	29.85
ReDi[3]	53.24	36.33	30.34	27.75	26.78
Di4C	55.42	36.52	27.66	23.04	21.54
Di4C + ReDi[1]	**47.50**	30.92	24.81	21.88	20.32
Ours	56.08	**27.59**	**22.92**	**21.35**	**16.97**
Entropy					
DUO+DCD	4.82	5.25	5.35	5.36	5.35
ReDi[1]	5.43	5.49	5.48	5.44	5.42
ReDi[2]	5.43	5.44	5.44	5.41	5.39
ReDi[3]	5.33	5.30	5.28	5.26	5.25
Di4C	4.81	5.02	5.07	5.05	5.03
Di4C + ReDi[1]	4.96	5.07	5.08	5.09	5.07
Ours	4.96	5.06	5.09	5.18	5.02

Table 4: ImageNet 256×256 with four network evaluations. Every method is reported at the best sampling configuration found by a sweep. Best value per column in bold.

Model	FID ↓	IS ↑
MaskGIT (Chang et al., 2022)	10.90	184
SDTT (Deschenaux & Gulcehre, 2025)	8.97	205
Di4C (Hayakawa et al., 2025)	**6.20**	216
ReDi[1] (Yoo et al., 2025)	7.58	228
ReDi[2] (Yoo et al., 2025)	7.86	240
Ours	7.81	**250**

E Unconditional Text Generation Samples ($K = 64$)

Below we provide unconditional 1,024-token text samples generated by our OpenWebText model using the exact-marginalization sampler of Equation 34–Equation 32 at $K = 64$ sampling steps and temperature $T = 1.0$. Across the evaluation batch, this configuration achieves the generative perplexity and entropy reported for "Ours" at $K = 64$ in Table 3 (LLaMA-3.1-8B generative perplexity 16.97, entropy 5.02).

Sample 1: Energy Policy & Regulatory Hearing ($K = 64$, $T = 1.0$)

<|endoftext|> statement that he was "deeply disappointed" that the company's actions were "not in line with our public policy" during the regulatory process.

"Everyone have a right to have concerns about energy security, and we do not believe that actions like this are in the vital national interest," Price said in the statement.

Pamper E. Price said that the firm paid roughly $14.1 million to lobbying groups in 2013, but a lawsuit filed by the Institute for Energy Policy Studies, the Washington-based Ceres' think group, that it won't be any more attention from the government.

The company is expected to go before the House Committee on Commerce and Transportation before joining the House of Representatives this fall and hold a hearing. The White House said in a statement on Friday that it would continue to get the company's representatives to investigate its activities.

"We are very disappointed this firm played an important role in efforts to implement critical energy security in our region and look forward to continuing to improve the decision being made to the American people of our region," the company said in the statement.

The firms ties to politics have raised questions about the effectiveness of the Obama administration's political attacks on the industry, including the closure of a variety of plants in the United States and the hiring of about 1,200 employees from 35 individuals and nonprofit groups, including fossil fuel firms and environmental groups. The industry is heavily backed by special interests.

One August, the firm's actions were part of the efforts to block the construction of the Dakota Access pipeline through Minnesota, the project being planned up in North Dakota carrying coal and crude oil.

The Trump administration said the actions were related a March 2015 rule, which temporarily halted efforts to save large sections of the proposed pipeline. Protesters repeatedly blocked a portion of the proposed pipeline.

\We look forward to hearing before the House today whether a few of these actions are in line with our national policy," Price said. \There is no evidence, in fact, that these actions involving Goldstone Perkins are politically motivated and are not the policies we should be guided by."<|endoftext|>

Sample 2: Public Health Study & International News ($K = 64, T = 1.0$)

<|endoftext|> by reducing their daily intake of nutrients, such as fruits, vegetables, and nuts, could contribute to a large decrease in the number of displaced in China.

The report, on the issue of food insecurity in China during the growing season, was authored by researchers from the Federal Institute of Public Affairs and the Department of Public Health of the University of Shanghai. The team, led by professors Zhou, U.S. Department of Agriculture, and Liu, also focused on the issue of the National Food Standards, which are mostly designed to combat the problem of hunger and food insecurity in China.

A Professor of Public Health at the University, Wei Mao, who not involved as a senior author of the report, said that the administration decided not to enforce the National Standards "due to the misinformation or distortion of information that is currently available. Our results that excessively high-fat foods such as fruits, vegetables, can lead to high levels of blood sugar. These foods have increase the risk of heart disease and increase the risk of diabetes."

The consumption of sugary drinks contributes to a high risk of food insecurity. Tens of thousand of the people currently without food in China is associated with the risk of diabetes mellitus which has nearly doubled in the past 25 years. And according to a UN monitor, the majority of the citizens of the world, Southeast Asia and East Asia, live on poor quality food conditions.

Earlier this month, Shanghai Municipal Commission for Food Safety and Integrity, maker of the Standards, said that the government was carrying out an effort to improve the use of fresh vegetables in Beijing during the growing season. It's critical to understand different varieties of vegetables offer, such as fresh varieties, to encourage people to eat more vegetables, according to the report.

The authors also explained that the global problem of food insecurity is that the government's largely neglected for decades, and which has hampered the efforts to combat the situation in China. As a result, they said, the need to reduce over-consumption and use of fresh food is a top priority in China.<|endoftext|>

By James Mayr
London - World News staff

British military and intelligence agencies on Friday called on the international community to prosecute the extremist organisations and anti-government forces to defeat in the name of peace and stability.

The calls were made by at least two thirds of the leading members of British military, for the restoration of peace talks between Turkey and the rebels in northern Syria, and for an end to the war in Syria, which has left 250,000 people fleeing the country from abroad.<|endoftext|>

Sample 3: International Diplomacy & Economic Aid ($K = 64$, $T = 1.0$)

<|endoftext|>United States has spoken out against what critics say is a poverty-ridden, expansionary and precarious economic situation in Egypt. Deputy Foreign Minister Rafa Mansour-Hahallah told reporters on Saturday the United States was \aware of a serious crisis with the country's dire situation." \The U.S. is actively in the process to overcome this humanitarian crisis," she said, after releasing a report saying that the Egyptian economy is in a \grave state of stagnation in the country."

The report said that the focus was on strengthening the security of the people and distributing programs to boost growth to end financial hardships for those living below the poverty level. She added that the situation of country is in a difficult position that any growth in activity could jeopardizing its security. Total oil production was cut by 5.2 million people in 2009 to just 3.1 million, according to the report.

To mitigate the effects of the cuts, the federal subsidy is to freeze until the end of March. The proposed cuts, which will be needed to remove 3.5 million from the budget, where the central government has reserves, are only temporary. However, according to the report, the important thing is to improve the quality of life in the United States and \to restore job security and life changing services through the government's ambitious infrastructure plan," administration officials said on Saturday.

The plan is focused on creating jobs by 2011 and is meant to provide relief to the tens of millions of women, children and people living past the retirement age by bringing in full-time jobs. The State Department spokesman, Jay Carney, said that the federal government was focusing mainly to poverty-era programs and measures to help the affected people, most of whom are still facing job insecurity. \The government working hard on words, speaking by themselves." Carney said, referring to comments published in The Washington Post. The U.S. government is determined to play a leading role in addressing this humanitarian situation, the official said. In addition to the call on the central government to take no further restrictions on oil and gas production, the United States however made it clear that the government would not place further restrictions. \The United States continues engaging with the people, and encouraging businesses to do as much as they can help restore the security of the people, as soon as now." the United States Department said in a statement.<|endoftext|>

Sample 4: Internet Openness, Privacy Law & Climate Study ($K = 64, T = 1.0$)

<|endoftext|> on June 7, 2011, in the city of Seattle, the US and Russia agreed to collaborate on three projects proposed by Russia, with the goal of protecting personal information and then they stole this information from the United States for the past 15 years. The most pressing thing they have to do is put pressure on the US government to pay for these projects because their practices have changed, and the peasants affected by this pipeline project are now before the eyes of the world," he said<|endoftext|>

Turkey

In January 2015, a citizen of Iran is six months pregnant in Turkey allegedly find themselves attempting to collect personal information of the Turkish government during a meeting of parliament of Turkey March 7, 2015. The Iranian was attempting to collect information about patient behavior, health and safety, and national security, breaking Turkish privacy laws and attempting to co-operate with Turkish authorities. No criminal charge was committed. The fine imposed on the person was determined by international law.

Germany

The United States, the United Kingdom, and the Netherlands support a variety of scientific, technological and social programs which benefit the people and environment of the world, and serve to be the engine of the global economy. This support is supported by the National Institutes of Health Sciences, the National Institutes of Health, the National Institute of Science and Technology, the Medical School of the University of Maryland, the School of Medicine and Public Health, the Massachusetts Institute of Technology, Energy and Environment, and Harvard University's School of Science, Space and Public Policy, and the Center for Environmental Science and Data Analysis at the University of Maryland.

United States

The United Nations Security Council reached an agreement on the openness of the Internet on July 11, 2011. One of the goals was to protecting national security and the openness of the Internet. On a global level, the Internet is probably one of the largest ecosystems in the world today. For example, the size of the web today is about 2.5 billion, the nature of the content has slightly through a variety of events, including melting Arctic sea ice. But over the course of the year, the surface extent of Arctic sea ice, increased steadily worldwide. A new study, shows glaciers now for about 55 percent of the world's surface ice, and about one-to-third of it is sea ice. (Changes to address climate change adaptation were made by three international experts, from the Institute for Media, Policy and Media Research, and from the World Economic Forum–WHO.)

In June 2011, under Barack Obama, the United States and Russia reached a series of agreement which is designed to promote greater online openness on the Internet. The elements of this agreement: an international commitment to Internet openness, every online form of communication, full Web-wide Internet access, formal reviews, and certifying Internet providers for computers, mobile phones, and other forms of information.<|endoftext|>

Sample 5: Sports Journalism: NHL & Preseason Coverage ($K = 64, T = 1.0$)

<|endoftext|>Yes.

Muthyomun is an important member of the Chicago side which went through the very well and ended up in a third position in the W-L playoffs.

The club had a favorable relationship with the Philadelphia Eagles and took over under the direction of Muthyomun in January. At the same time, the club now looking to reach at least 10th in W-L playoffs.

Chicago head coach Antoine Bouchard is expected to visit the Chicago Blackhawks and make his preseason debut Wednesday, March 14 at the Chicago Blackhawks. In the past few months, he has been able to play as a key part of the lineup for the Chicago Blackhawks, according to the English-language website.

Antoine Brouwer is a key part of the lineup for the Blackhawks and has already signed with the club most recently from the Anaheim Ducks. Anaheim had a great team last season to make the Stanley Cup Playoffs, and he is expected to be a part of the lineup for the first time since his injury in the Eastern Conference Playoffs.

The Chicago Blackhawks had a great 2016 season and have a real chance at making another playoff run in the Eastern Conference. The club finished 23-16 in the regular season and was able to earn a third place spot in the WHLs first bid for the Stanley Cup by making a powerful shot that passed of his hand and his right foot bounced to the back of the net. Following hot start to the season he then scored 2 beautiful third and 2 goals in the third period against the Dallas Stars (4-2-1) and he was able to capitalize on it.

He is considered one of the best players in the league and he has developed a good amount of confidence in his game. His right hand gives him good opportunities to make plays and boost his acceleration, for him to be able to hold on to his feet. While playing with his left foot down, he will have to make sure to find some movement at the end of the season with the Chicago Blackhawks.

Brouwer returned to the Blackhawks after playing well in the American Hockey League. He was named the American Hockey League Most Outstanding Player of the Year and played a key part in the Chicago Blackhawks' victory over the Los Angeles Kings in the playoffs.<|endoftext|>

Sample 6: US Politics & Election Coverage ($K = 64$, $T = 1.0$)

<|endoftext|> the most dangerous people in the federal government in 2016.

The FBI confirmed they only interviewed Flynn on January 20.

Washington (CNN) Matt Flynn reportedly not been interviewed by the FBI after the call was held, Director Comey reportedly chose not to answer all questions about it.

Senate Democratic leaders said the action was the effective result of the official action and viewed the actions of President Trump as "a coup."

"In order to keep Russia out of our country, we must continue to rely on the resolve of all members of Congress and support the American people," said Baucus, D-Mont., chairman of the council of The Church of Jesus Christ of Latter-day Saints.

Vice President Joe Biden, at the center of hacking, said in a State of the Union address that he didn't immediately follow the department's emergency proclamation but it was an "appropriate decision."

The accusations made their way through White House channels Saturday, said Frank Reich, a senior fellow at the liberal Center for American Progress and a member of the White House under President Donald Trump.

"We have repeatedly repeated in the last days about the Trump campaign's contradictory statements about the use of chemical weapons against civilians," Reich told The New York Times. "Now, Mr. Biden's e-mails are a very damaging line of attack for the incoming administration."

Senate Majority Whip Max Baucus, D-Mont., declined to comment about the allegations. On Saturday, he said, "We just have to see the results of the investigation conducted by prosecutors in our community."

Former Attorney General Scott Baquet, who served under Robert M. Mueller as attorney general of the White House, has condemned the accusations.

"It's difficult to say whether there's nothing else, but they certainly are there, and it's highly subjectively concluded that there's no merit in this case," he said.<|endoftext|>

E.1 CFG Ablation on ImageNet

Ablation Studies Sweeping classifier-free guidance (CFG) weight $w \in \{1.5, 2.0, 3.0, 4.0, 5.0\}$ at 4 and 8 sampling steps (Figure 4), we find that precision increases and recall decreases monotonically with w at both step budgets, that the best 4-step FID (7.81) is attained at $w = 3.0$, and that the optimal guidance weight shifts down to $w = 1.5$ (FID 6.90) once step budget increases to 8, where iterative refinement already improves fidelity so a lower guidance weight suffices to preserve diversity. Figure 5 shows the corresponding 4-step samples across w, with the random seed held fixed per class.

E.2 Additional Qualitative Samples on ImageNet

Figure 6 demonstrates the 4-step qualitative comparison to four ImageNet classes (goldfinch, sea anemone, cheeseburger, spotted salamander) following Yoo et al. (2025).

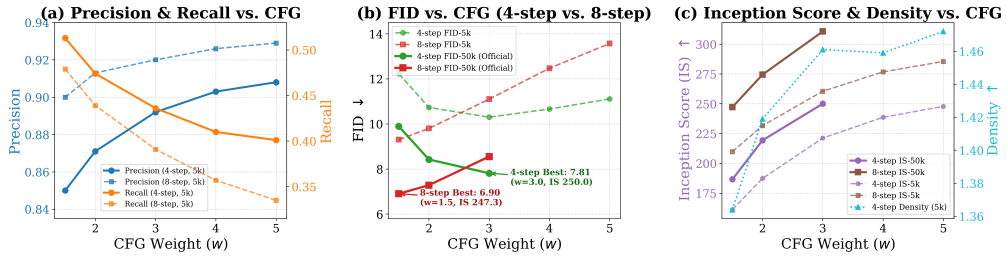

Figure 4: Ablation on the classifier-free guidance weight w used to sample the training pairs, evaluated at 4- and 8-step generation on ImageNet 256×256 (50k samples). (a) Precision and recall vs. w. (b) FID vs. w, with the best configuration at each step budget annotated. (c) Inception score and density vs. w.

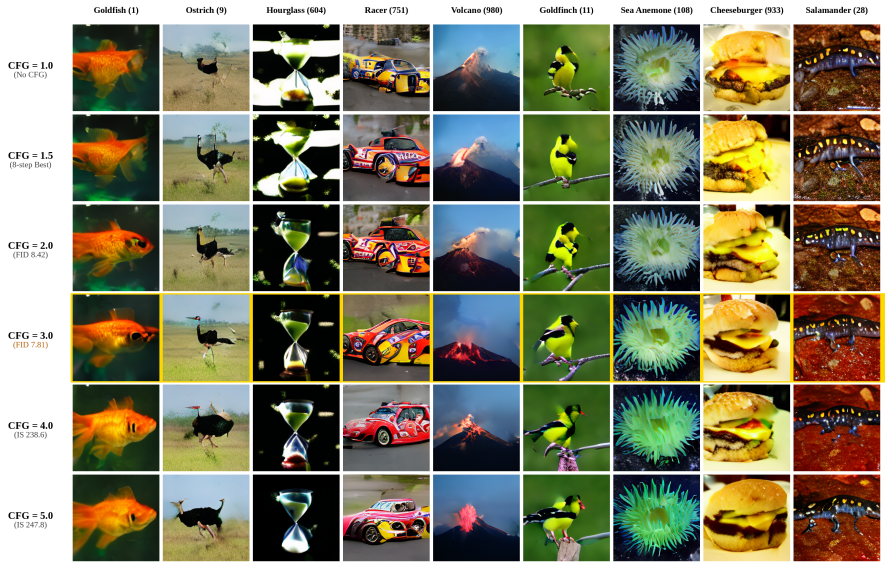

Figure 5: Qualitative effect of the CFG weight w used to sample the training pairs, at 4-step generation on ImageNet 256×256, for nine classes: goldfish, ostrich, hourglass, racer, volcano, goldfinch, sea anemone, cheeseburger and spotted salamander. The random seed is held fixed per class across $w \in \{1.0, 1.5, 2.0, 3.0, 4.0, 5.0\}$, so only the effect of w varies; the optimal 4-step setting ($w = 3.0$, FID 7.81) is highlighted.

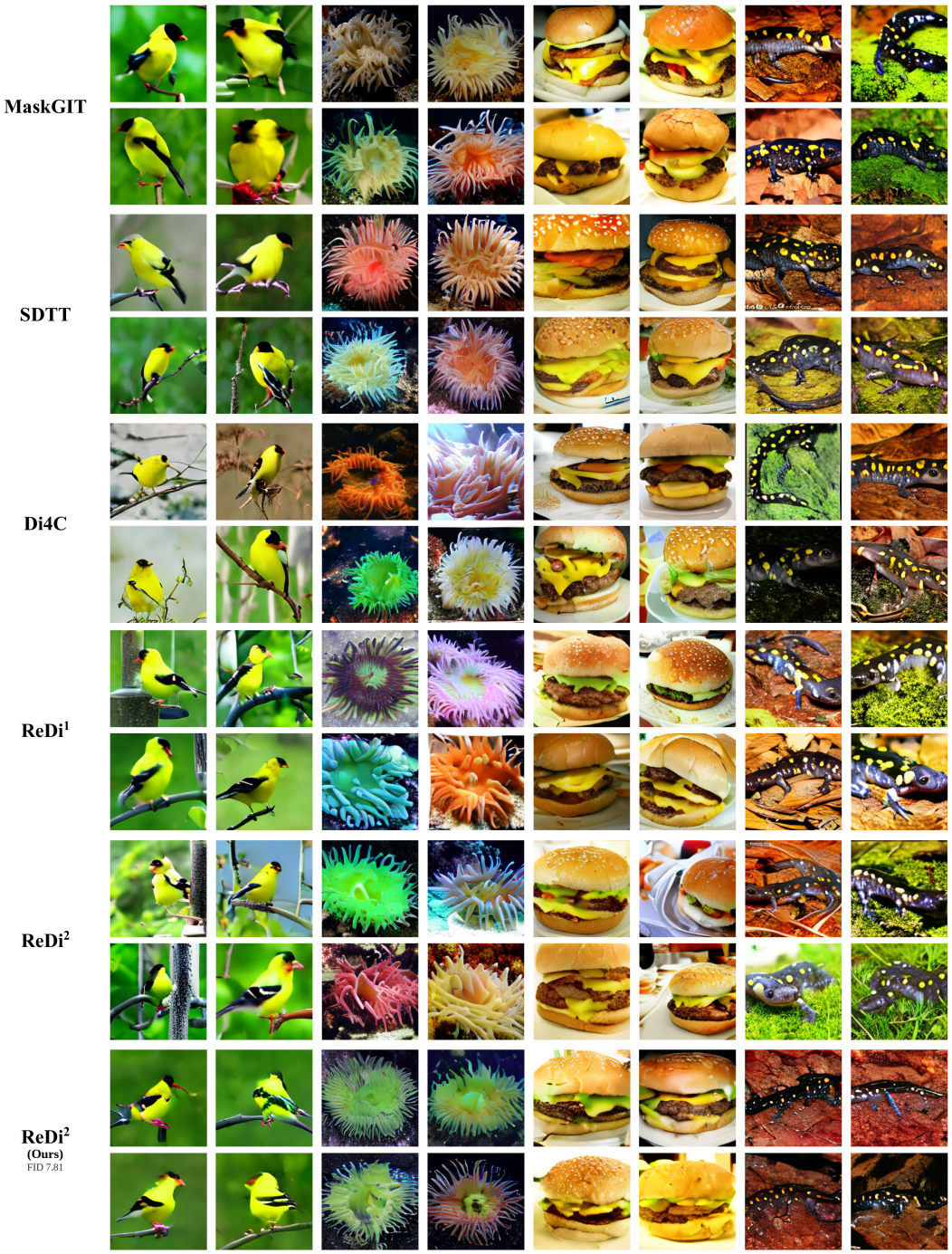

Figure 6: 4-step generation results on ImageNet (classes goldfinch, sea anemone, cheeseburger, spotted salamander), two samples per class and model. Our method (bottom row, FID 7.81) is appended below the baselines.