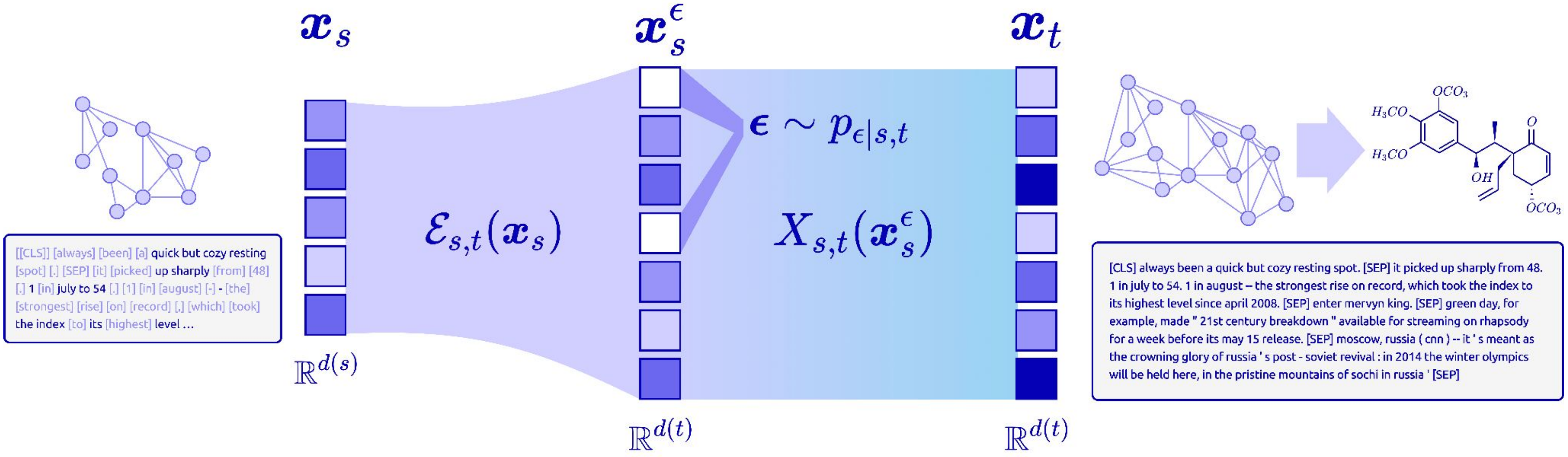
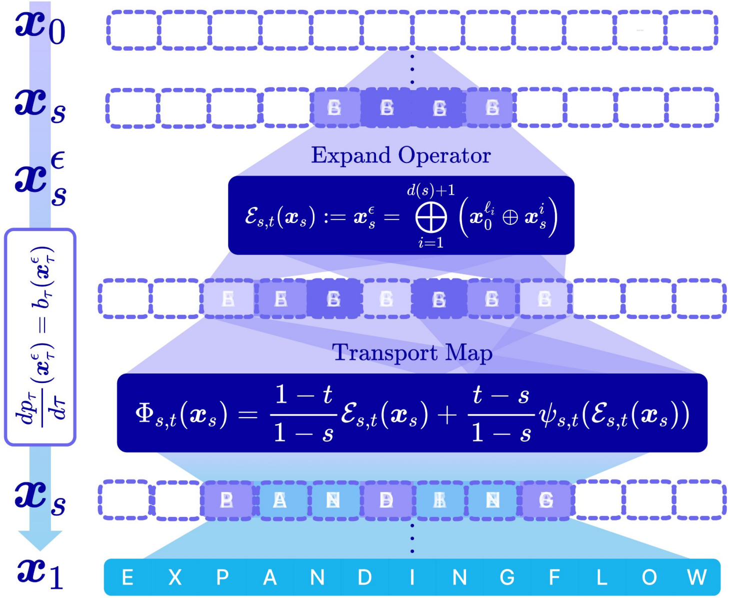
Expanding Flows for Generative Modeling

Sophia Tang | *Microsoft Research New England Generative Modeling & Sampling Seminar* | *July 20, 2026*



Agenda



1. Background: Flow Matching & Flow Maps
2. Motivation: Why a Fixed Canvas Is Limiting
3. Expanding Generative Flows (EFlows)
4. Expanding Flow Maps (EFMs)
5. Discrete Expanding Flow Maps (DEFMs)
6. Experiments
7. Conclusion + Q&A

Background: Flow Matching & Flow Maps

Generative Flows & Stochastic Interpolants

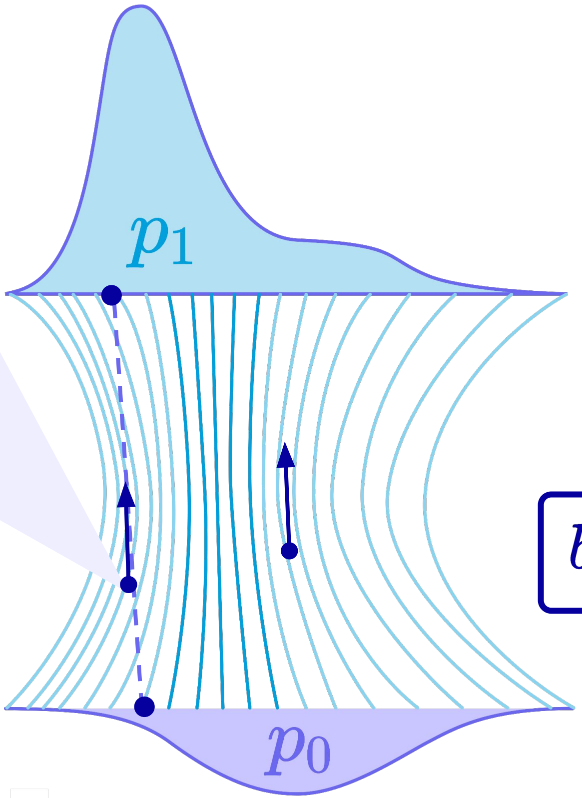
A continuous-time path between a prior and the data distribution

Interpolant

Endpoint-conditioned continuous-time path

$$\boxed{\boldsymbol{x}_t = I_t(\boldsymbol{x}_0, \boldsymbol{x}_1)}$$

$$\boldsymbol{x}_{t=0} = \boldsymbol{x}_0 \sim p_0$$
$$\boldsymbol{x}_{t=1} = \boldsymbol{x}_1 \sim p_1$$
$$\boldsymbol{x}_t \sim p_t = \mathrm{Law}(I_t)$$



Marginal velocity

Expectation of the time derivative over all paths passing x

$$\boxed{b_t(\boldsymbol{x}) = \mathbb{E}[\dot{\boldsymbol{x}}_t | \boldsymbol{x}_t = \boldsymbol{x}]}$$

Conditional Flow Matching

Learning the velocity field from two empirical distributions

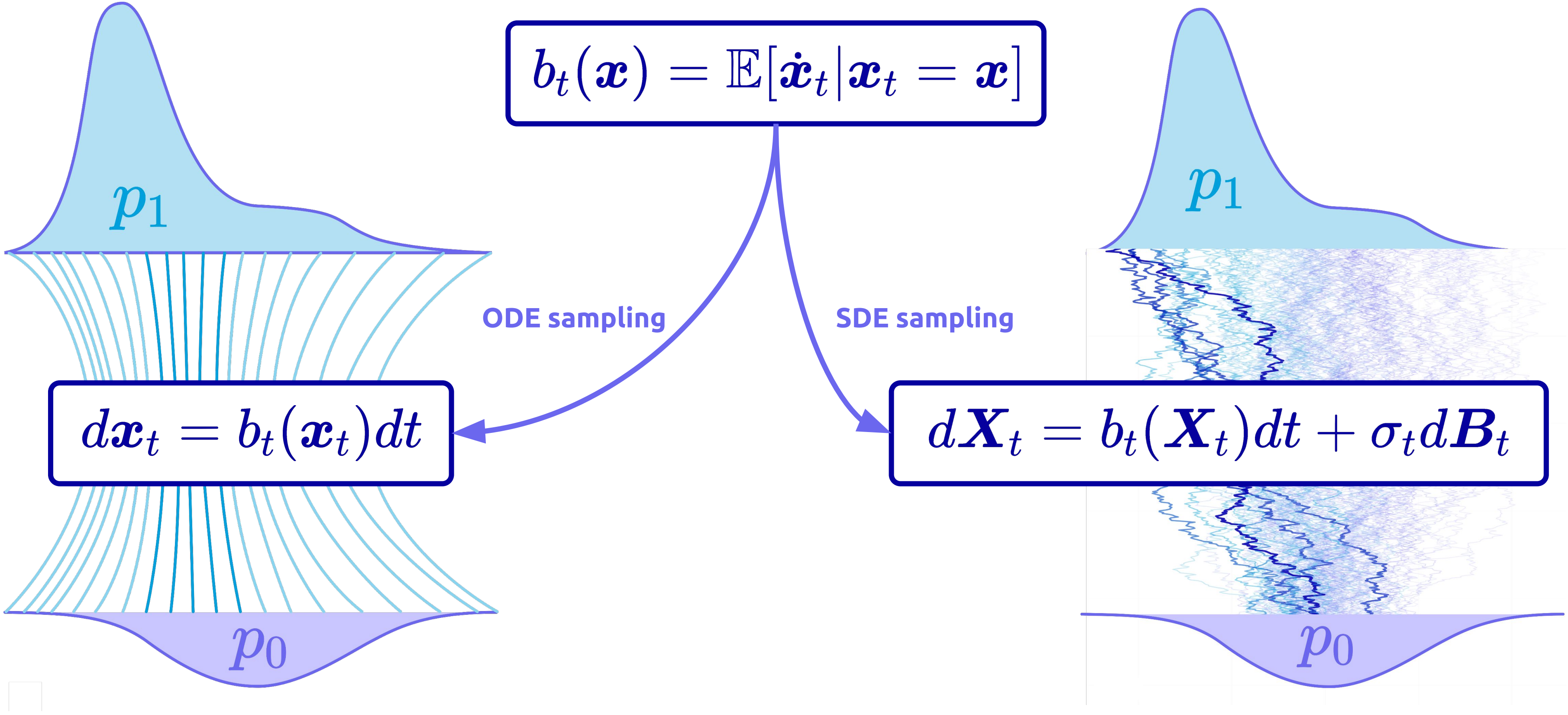
$$b_t(\boldsymbol{x}) = \mathbb{E}[\dot{\boldsymbol{x}}_t | \boldsymbol{x}_t = \boldsymbol{x}]$$

Minimizer of conditional flow matching loss

$$\mathcal{L}_{\mathrm{CFM}}(\hat{b}) := \int_0^1 \mathbb{E}_{\boldsymbol{x}_0, \boldsymbol{x}_1}\left[\left\| \hat{b}_t(I_t(\boldsymbol{x}_0, \boldsymbol{x}_1)) - \partial_t I_t(\boldsymbol{x}_0, \boldsymbol{x}_1) \right\|^2\right] dt$$

Conditional Flow Matching

Learning the velocity field from two empirical distributions

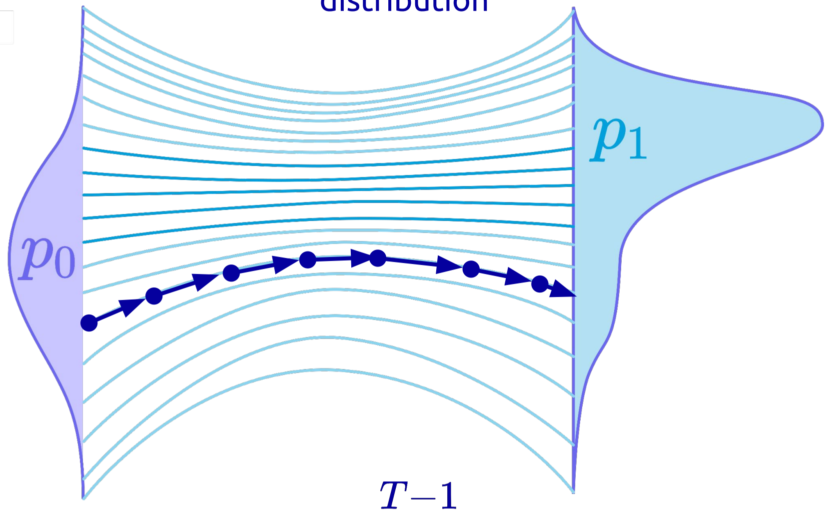


The Cost of Sampling: Many Function Evaluations

Integrating the ODE/SDE requires many small steps

Flow Matching

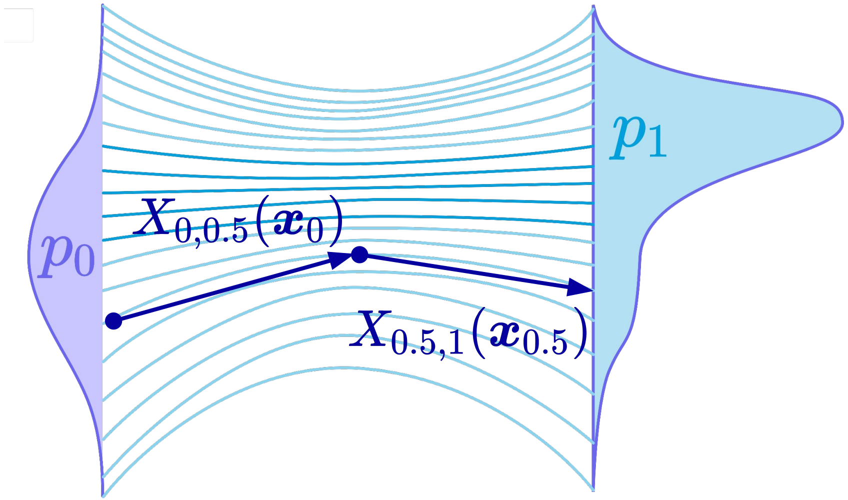
Requires many Euler integration steps to get an accurate sample from the data distribution



$$\hat{\boldsymbol{x}}_1 = \boldsymbol{x}_{t_0} + \sum_{i=0}^{T-1} b_{t_i}(\boldsymbol{x}_{t_0})\Delta t$$

Flow Maps

Large jumps along the ODE trajectory



$$\hat{\boldsymbol{x}}_1 = X_{0.5,1}(X_{0,0.5}(\boldsymbol{x}_0))$$

Flow Maps for Trajectory Distillation

Jumping directly between any two times along the ODE

$$\boxed{X_{s,t}(\boldsymbol{x}_s) = \boldsymbol{x}_t = \boldsymbol{x}_s + (t-s)\underbrace{v_{s,t}(\boldsymbol{x}_s)}_{\text{Mean velocity over interval [s,t]}}} \quad \forall s,t \in [0,1]$$

On the diagonal: $t = s : v_{s,s}(\boldsymbol{x}_s) = b_s(\boldsymbol{x}_s)$ → **Train with CFM loss**

On the off-diagonal: $t \neq s : v_{s,t}(\boldsymbol{x}_s)$ → **Requires consistency objectives**

Three Equivalent Conditions for a Valid Flow Map

Each can be written as a consistency objective

$$\text{Lagrangian:} \quad \partial_t X_{s,t}(\boldsymbol{x}_s) = v_{t,t}(X_{s,t}(\boldsymbol{x}))$$

the map's endpoint follows the instantaneous velocity

$$\text{Eulerian:} \quad \partial_s X_{s,t}(\boldsymbol{x}) + v_{s,s}(\boldsymbol{x}) \cdot \nabla_{\boldsymbol{x}_s} X_{s,t}(\boldsymbol{x}) = 0$$

shifting the start time leaves the output unchanged

$$\text{Semigroup:} \quad X_{u,t}(X_{s,u}(\boldsymbol{x})) = X_{s,t}(\boldsymbol{x})$$

two chained jumps s→u then u→t equals one direct jump s→t

Full objective: $$\mathcal{L}_{\text{FM}}(\hat{v}) = \mathcal{L}_{\text{diag}}(\hat{v}) + \mathcal{L}_{\text{cons}}(\hat{v})$$

set each condition to zero, take squared residual

Boffi et al. 2025

Flow Maps for Discrete Data

Repurposing continuous flow-map objectives on the simplex

Mean denoiser: $\psi_{s,t}(\boldsymbol{x}) = \boldsymbol{x} + (1-s)v_{s,t}(\boldsymbol{x}), \quad v_{s,t}(\boldsymbol{x}) = \dfrac{\psi_{s,t}(\boldsymbol{x}) - \boldsymbol{x}}{1-s}$

$$\boxed{X_{s,t}(\boldsymbol{x}) = \frac{1-t}{1-s}\boldsymbol{x} + \frac{t-s}{1-s}\psi_{s,t}(\boldsymbol{x})} \quad \forall s,t \in [0,1]$$

Mean denoiser over interval [s,t]

$$\boxed{\mathcal{L}_{\text{diag}}(\hat{\psi}) = -\sum_{i=1}^{L} D_s(\boldsymbol{x}_s^i) \cdot \log \hat{\psi}_{s,s}(\boldsymbol{x}_s^i), \quad \mathcal{L}_{\text{cons}}(\hat{\psi}) = -\sum_{i=1}^{L} \bar{\psi}_{s,t} \cdot \log \hat{\psi}_{s,t}(\boldsymbol{x}_s^i)}$$

Defined via consistency identities

Lee et al., Potaptchik et al. 2026

Motivation: The Fixed-Canvas Limitation

Existing flow maps commit to a state-space size at initialization

1. **Continuous flow maps operate on fixed dimension *d*.** The state is fixed in d-dimensions before generation begins and never changes size.
2. **Discrete flows operate on fixed sequence length *L*.** The token grid is committed to up front and the model cannot grow or shrink the sequence.

In many real tasks, the size of the output is itself something to learn and predict at inference

How can we enable few-step generation of continuous, discrete, and multi-modal data with adaptive dimensionality at inference?

Our Solution: Factor the Flow Map into Expand + Transport

A single map that jointly expands and denoises the state

$$\Phi_{s,t}(\boldsymbol{x}_s) := X_{s,t}(\mathcal{E}_{s,t}(\boldsymbol{x}_s))$$

Expand operator inserts new coordinates/tokens, raising the dimension

Transport map moves the expanded state forward along the interpolant

Fixed-dimensional flow maps = the special case where the expand operator is the identity

Expanding Generative Flows

The Expanding Interpolant

Defining a generative flow with increasing dimensionality

Generative Flow with Expanding Dimensionality

$$\mathcal{P}(\mathbb{R}^{d(0)}) \ni p_0 \to \cdots \to p_s \to \cdots \to p_t \to \cdots \to p_1 \in \mathcal{P}(\mathbb{R}^{d(1)})$$

$d(s) \leq d(t)$ **for all** $0 \leq s < t \leq 1$

?

Not a well-defined flow: no diffeomorphism $\mathbb{R}^{d(s)} \to \mathbb{R}^{d(t)}$

no fixed-dimensional velocity field transforms $p_s \to p_t$

The Expanding Interpolant

Defining a generative flow with increasing dimensionality

Generative Flow with Expanding Dimensionality

$$\mathcal{P}(\mathbb{R}^{d(0)}) \ni p_0 \to \cdots \to p_s \to \cdots \to p_t \to \cdots \to p_1 \in \mathcal{P}(\mathbb{R}^{d(1)})$$

Augment the source with extra noise coordinates

$$\boldsymbol{\epsilon} \sim p_{\epsilon|s,t} \in \mathbb{R}^{d(t)-d(s)}$$

to lift the source dimension to the target dimension before transporting (integrating the flow)

Not a well-defined flow: no diffeomorphism $\mathbb{R}^{d(s)} \to \mathbb{R}^{d(t)}$

no fixed-dimensional velocity field transforms $p_s \to p_t$

The Expand Operator

Lifting the source into the target dimensionality

$$\boxed{\mathcal{E}_{s,t}(\boldsymbol{x}_s, \boldsymbol{\epsilon}) : \mathbb{R}^{d(s)} \times \mathbb{R}^{d(t)-d(s)} \to \mathbb{R}^{d(t)}}$$

Combines the existing coordinates of $\boldsymbol{x}_s$ with the new coordinates of ε via a **placement scheme**

1. **Concatenation:** Append the new coordinates to the end of the state. This is the natural instantiation for *autoregressive sequence generation or for appending points to a point cloud*.

$$\mathcal{E}_{s,t}(\boldsymbol{x}_s, \boldsymbol{\epsilon}) = \mathrm{Cat}(\boldsymbol{x}_s, \boldsymbol{\epsilon})$$

2. **Positional Insertion:** Interleaves the entries of ε at a subset of positions within {1, …, d(t)} in the augmented state. *Useful for any-length sequence generation or infilling.*

3. **Child Expansion**: attaches each new coordinate to a parent entry and initializes it relative to that parent. Suitable for *coarse-to-fine generation on structured states*.

The Expand Operator

Lifting the source into the target dimensionality

$$\boxed{\mathcal{E}_{s,t}(\boldsymbol{x}_s, \boldsymbol{\epsilon}) : \mathbb{R}^{d(s)} \times \mathbb{R}^{d(t)-d(s)} \rightarrow \mathbb{R}^{d(t)}}$$

Combines the existing coordinates of $\boldsymbol{x}_s$ with the new coordinates of ε via a **placement scheme**

1. **Concatenation:** Append the new coordinates to the end of the state. This is the natural instantiation for *autoregressive sequence generation or for appending points to a point cloud.*

The augmented noise can be a deterministic function of *s* and *t* (e.g., append i.i.d. Gaussian to the end) **or learned via a neural network conditioned on the current state and time interval**

2. **Positional I** …, d(t)} in the augmented

3. **Child Expan** … relative to that parent. Suitable for *coarse-to-fine generation on structured states.*

The Expand Operator

Lifting the source into the target dimensionality

$$\mathcal{E}_{s,t}(\boldsymbol{x}_s, \boldsymbol{\epsilon}) : \mathbb{R}^{d(s)} \times \mathbb{R}^{d(t)-d(s)} \to \mathbb{R}^{d(t)}$$

Combines the existing coordinates of $\boldsymbol{x}_s$ with the new coordinates of ε via a **placement scheme**

1. **Concatenation:** Append the new coordinates to the end of the state. This is the natural instantiation for *autoregressive sequence generation or for appending points to a point cloud.*

Expansion alone fixes the conditional law of the augmented state…

$$p_{t|s}(\cdot \mid \boldsymbol{x}_s) = (\mathcal{E}_{s,t}(\boldsymbol{x}_s, \cdot))_{\#} p_{\epsilon|s,t}(\cdot \mid \boldsymbol{x}_s)$$

…but it does not yet say how that state is transported to the target time.

2. **Position** … d(t)} in the augment…

3. **Child Ex** … ive to that parent. Suitable for *coarse-to-fine generation on structured states.*

Local Time Coordinate

Every inserted dimension is denoised on a common clock

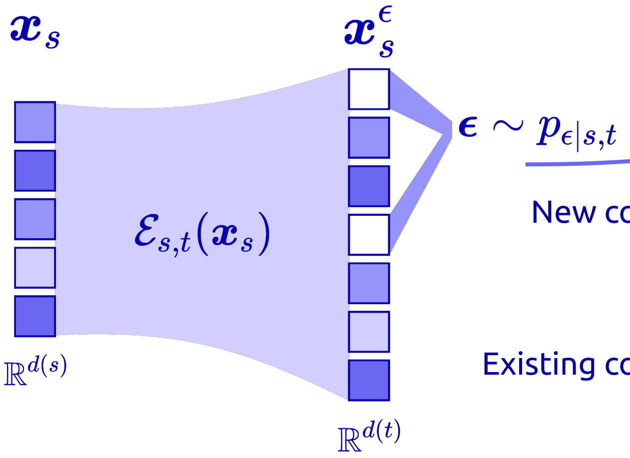
Local clock starts at 0 at the insertion time:

$$t = t_i^{\text{ins}} \implies t_i = 0$$

For all times after insertion, the local time is:

$$t_i = \frac{t - t_i^{\text{ins}}}{1 - t_i^{\text{ins}}} \in [0, 1]$$

Rescales each coordinate's global time onto its own [0,1] interval



Global time s:

New coordinates: $s = t_i^{\text{ins}}$ **and** $s_i = 0$

Existing coordinates: $s > t_i^{\text{ins}}$ **and** $s_i = \dfrac{s - t_i^{\text{ins}}}{1 - t_i^{\text{ins}}}$

Transport Along the Expanding Interpolant

Integrate the standard velocity field over active coordinates

Integrate the ODE in the augmented space:

$$\boldsymbol{x}_t = \boldsymbol{x}_s^{\epsilon} + \int_s^t b_\tau(\boldsymbol{x}_\tau^{\epsilon})d\tau$$

Per coordinate

Local clock:

$$\boldsymbol{x}_t[i] = \boldsymbol{x}_s^{\epsilon}[i] + \int_{s_i}^{t_i} b_{\tau_i}(\boldsymbol{x}_{\tau_i}^{\epsilon}[i])d\tau_i$$

Per coordinate velocity:

$$b_t(\boldsymbol{x}_t^{\epsilon}) = \begin{cases} b_{t_i}(\boldsymbol{x}_{t_i}^{\epsilon}[i]) & t_i^{\text{ins}} \leq t \\ \mathbf{0} & t_i^{\text{ins}} > t \end{cases}$$

Train $\hat{b}_t$ with the CFM objective on active coordinates

Inactive (not-yet-inserted) **coordinates get velocity 0 and stay unchanged**

Expanding Generative Flow (EFlow) Definition

Compose the expand operator with a fixed-dimension transport map

EFlow is the composition of the expand operator and transport map:

$$\Phi_{s,t} := X_{s,t} \circ \mathcal{E}_{s,t} : \mathbb{R}^{d(s)} \to \mathbb{R}^{d(t)}$$

$$X_{s,t} : \mathbb{R}^{d(t)} \to \mathbb{R}^{d(t)}$$

Distribution of the augmented state $\boldsymbol{x}_s^\epsilon := \mathcal{E}_{s,t}(\boldsymbol{x}_s)$

$$p_{s,t}^\epsilon := (\mathcal{E}_{s,t})_\# \left(p_s \otimes p_{\epsilon|s,t}\right) \in \mathcal{P}(\mathbb{R}^{d(t)})$$

Two Conditions:

1. **Marginal consistency:** $(X_{s,t})_\# \, p_{s,t}^\epsilon = p_t$ (target marginal fixed)
2. **Coupling consistency:** $X_{s,t}$ agrees with the interpolant flow on the embedded source coordinates (fixes the coupling, not just the marginal)

EFlows Are Piecewise-Deterministic Markov Processes

Unifying transport and expansion into one well-defined process

Setup: State space $\mathcal{M} := \bigsqcup_{t\in[0,1]} \mathbb{R}^{d(t)}$ (disjoint union of different-dimensional spaces)

Piecewise-deterministic Markov process (PDMP) on $\mathcal{M}$ with extended generator:

$$\mathcal{L}_t f(\boldsymbol{x}) = b_t(\boldsymbol{x}) \cdot \nabla f(\boldsymbol{x}) + \lambda_t(\boldsymbol{x}) \int \left(f(\mathcal{E}_t(\boldsymbol{x}, \boldsymbol{\epsilon})) - f(\boldsymbol{x})\right) p_{\epsilon|t}(d\boldsymbol{\epsilon})$$

Transport (smooth denoising)

Insertion intensity

Jump to higher dimensionality with kernel = $(\mathcal{E}_t(\boldsymbol{x}, \cdot))_{\#} p_{\epsilon|t}$

💡

This characterization is valid over the full interval, even across jump discontinuities

Change-of-Variables Along the Expanding Flow

Density evolves as if already at final dimension

Change of variables formula for the ODE in augmented space $\frac{d}{d\tau}\boldsymbol{x}_\tau^\epsilon = b_\tau(\boldsymbol{x}_\tau^\epsilon)$ **is:**

$$\log p_t(X_{s,t}(\boldsymbol{x}_s^\epsilon)) = \log p_s(\boldsymbol{x}_s^\epsilon) - \int_s^t \nabla \cdot b_\tau(X_{s,\tau}(\boldsymbol{x}_s^\epsilon))d\tau$$

For all $s \leq t$

Divergence sums only over active coordinates:

$$\nabla \cdot b_\tau = \sum_{i:t_i^{\text{ins}} \leq \tau} \partial_{\boldsymbol{x}^i} b_\tau^i$$

💡

Non-inserted coordinates contribute nothing to the density until their insertion time

Expanding Flow Maps

The Expanding Flow Map

Jointly expand and transport in a single learned step

The expanding flow map (EFM) takes the residual form:

$$\Phi_{s,t}(\boldsymbol{x}_s) := X_{s,t}(\mathcal{E}_{s,t}(\boldsymbol{x}_s)) =: \mathcal{E}_{s,t}(\boldsymbol{x}_s) + (t-s)v_{s,t}(\mathcal{E}_{s,t}(\boldsymbol{x}_s))$$

Raises dimension $\mathcal{E}_{s,t} : \mathbb{R}^{d(s)} \to \mathbb{R}^{d(t)}$

Mean velocity transports the expanded state to time t $v_{s,t} : \mathbb{R}^{d(t)} \to \mathbb{R}^{d(t)}$

💡 Standard residual flow-map form, but applied in the augmented space

The Tangent Condition and Diagonal Loss

Anchoring the flow map on the instantaneous diagonal

The tangent condition states that the limit at s → t is the instantaneous velocity in augmented space:

$$\lim_{s \to t} \partial_t \Phi_{s,t}(\boldsymbol{x}) = v_{t,t}(\boldsymbol{x}) = b_t(\boldsymbol{x})$$

Expanding Flow Map

Instantaneous velocity in d(t)

Diagonal Objective

$$\mathcal{L}_{\text{diag}}(\hat{v}) := \int_0^1 \mathbb{E} \left\| \hat{v}_{t,t}(\boldsymbol{x}_t) - \dot{\boldsymbol{x}}_t \right\|^2 dt$$

$$\dot{\boldsymbol{x}}_t = \begin{cases} \boldsymbol{x}_1^{(t)} - \mathcal{E}_{0,t}(\boldsymbol{x}_0) & \text{(self-distillation from data)} \\ \hat{b}_t(\boldsymbol{x}_t) & \text{(distillation from a teacher flow)} \end{cases}$$

intermediate state is $\boldsymbol{x}_t = (1-t)\,\mathcal{E}_{0,t}(\boldsymbol{x}_0) + t\,\boldsymbol{x}_1^{(t)}$ where $\boldsymbol{x}_1^{(t)}$ is the clean sample truncated to the dimensions present at time t

Expanding Flow Map Identities

Consistency constraints extended to the expanding interpolant

Lagrangian: $\partial_t \Phi_{s,t}(\boldsymbol{x}) = v_{t,t}(\Phi_{s,t}(\boldsymbol{x}))$

the map's endpoint follows the instantaneous velocity

Eulerian: $\partial_s \Phi_{s,t}(\boldsymbol{x}) + \nabla_{\boldsymbol{x}} \Phi_{s,t}(\boldsymbol{x}) \tilde{v}_{s,s}(\boldsymbol{x}) = 0$

shifting the start time leaves the output unchanged

Semigroup: $\Phi_{u,t}(\Phi_{s,u}(\boldsymbol{x})) = \Phi_{s,t}(\boldsymbol{x})$

two chained jumps s→u then u→t equals one direct jump s→t

Same as fixed-dimensional identities with the exception that $v_{s,s}$ in the Eulerian identity acts on the expanded state $\mathcal{E}_{s,t}(\boldsymbol{x}) \in \mathbb{R}^{d(t)}$

Consistency Training Objectives

Lagrangian, Eulerian, and semigroup distillation losses

$$\mathcal{L}_{\mathrm{LSD}}(\hat{\mathcal{E}}, \hat{v}) := \int_0^1 \int_0^t \mathbb{E}\left\| \partial_t \hat{\Phi}_{s,t}(\boldsymbol{x}_s) - \mathrm{sg}\Big(\hat{b}_t(\hat{\Phi}_{s,t}(\boldsymbol{x}_s))\Big) \right\|^2 ds dt + \mathcal{L}_{\mathrm{diag}}(\hat{v})$$

squared residual of the Lagrangian identity

$$\mathcal{L}_{\mathrm{ESD}}(\hat{\mathcal{E}}, \hat{v}) := \int_0^1 \int_0^t \mathbb{E}\left\| \partial_s \hat{\Phi}_{s,t}(\boldsymbol{x}_s) + \mathrm{sg}\Big(\nabla_{\boldsymbol{x}} \hat{\Phi}_{s,t}(\boldsymbol{x}_s) \hat{b}_s(\boldsymbol{x}_s)\Big) \right\|^2 ds dt + \mathcal{L}_{\mathrm{diag}}(\hat{v})$$

squared residual of the Eulerian identity

$$\mathcal{L}_{\mathrm{PSD}}(\hat{\mathcal{E}}, \hat{v}) := \int_0^1 \int_0^t \int_s^t \mathbb{E}\left\| \hat{\Phi}_{s,t}(\boldsymbol{x}_s) - \mathrm{sg}(\hat{\Phi}_{u,t}(\hat{\Phi}_{s,u}(\boldsymbol{x}_s))) \right\|^2 du ds dt + \mathcal{L}_{\mathrm{diag}}(\hat{v})$$

squared residual of the semigroup identity

sg(·) = stop-gradient on the self-distillation target

Unlike fixed-dimensional flow maps, $\hat{\Phi}_{s,t}$ is parameterized with both and
$\hat{\mathcal{E}}_{s,t}$ $\hat{v}_{s,t}$

Expanding Flow Maps Are Stochastic Flow Maps

Randomness from conditional noise injection induces a stochastic map

The pushforward of the expanding flow map with augmented noise drawn from the conditional distribution is:

$$p_{t|s}(\cdot|\boldsymbol{x}_s) = (\Phi_{s,t}(\boldsymbol{x}_s, \cdot))_{\#} p_{\epsilon|s,t}$$

Recovers the clean posterior $p_{1|s}(\cdot|\boldsymbol{x}_s)$ at t=1

💡

Captures the full posterior over future states: the expand + denoise composition reconstructs the interpolant at any t > s

Discrete Expanding Flow Maps

Discrete Expanding Interpolant

Growing sequence lengths by inserting tokens

Insertion Schedule α_t :

$$\Pr[t_i^{\text{ins}} \leq t] = \alpha_t \quad \text{where } \alpha_0 = 0, \alpha_1 = 1$$

Probability that token *i* is inserted before time *t*

Per-position state:

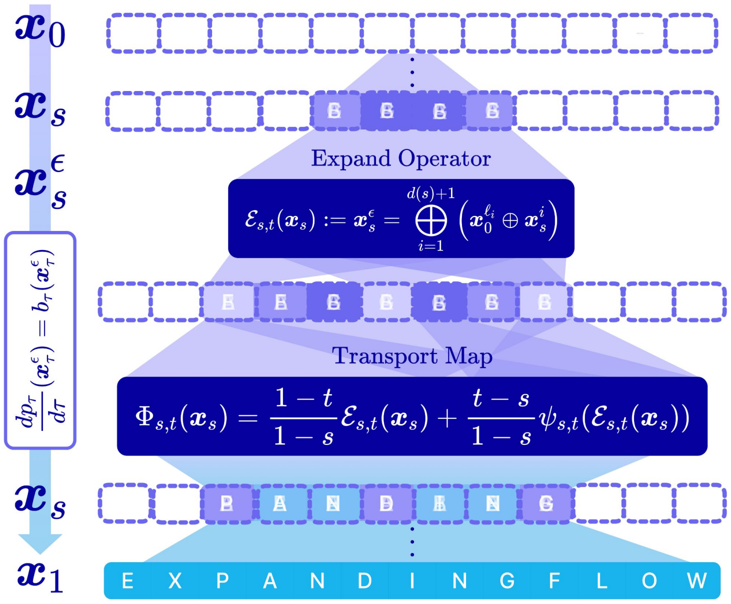
$$\boldsymbol{x}_t^i = \begin{cases} (\text{empty}), & t < t_i^{\text{ins}} \\ (1 - t_i)\boldsymbol{x}_0^i + t_i \boldsymbol{x}_1^i, & t \geq t_i^{\text{ins}} \end{cases}$$

Before insertion

After insertion

Sequence at time t = subsequence of currently inserted tokens:

$$\boldsymbol{x}_t = (\boldsymbol{x}_{t_1}^1, \ldots, \boldsymbol{x}_{t_{d(t)}}^{d(t)}) \in \mathbb{R}^{d(t) \times V}$$



Parameterizing Token Insertions

Predicting how many tokens land in each gap

Two-time insertion expectation:

$$\mathcal{I}_{s,t}(\boldsymbol{x}_s)[i] = \underbrace{\frac{\alpha_t - \alpha_s}{1-\alpha_s}}_{\text{fraction of remaining tokens inserted over (s,t]}} \mathcal{I}_s(\boldsymbol{x}_s)[i]$$

Total gap expectation at time s:

$$\mathcal{I}_s(\boldsymbol{x}_s)[i] := \mathbb{E}\big[\underbrace{g_i(s)}_{\text{count still missing from gap i}} \mid \boldsymbol{x}_s\big]$$

$$g_i(t) := \mathrm{ind}_i(t) - \mathrm{ind}_{i-1}(t) - 1$$

Insertion Loss:

$$\mathcal{L}_{\text{insert}}(\hat{\mathcal{I}}) = \mathbb{E}_t \mathbb{E}_{\boldsymbol{x}_0, \boldsymbol{x}_1}\left[\frac{\dot{\alpha}_t}{1-\alpha_t} \sum_i \phi\big(g_i(t), \hat{\mathcal{I}}_t(\boldsymbol{x}_t)[i]\big)\right] + \mathbb{E}_{s<t}\mathbb{E}_{\boldsymbol{x}_0, \boldsymbol{x}_1}\left[\sum_i \phi\big(\mu_i(s,t), \hat{\mathcal{I}}_{s,t}(\boldsymbol{x}_s)[i]\big)\right]$$

Diagonal term supervises against gap count: $g_i(t)$

Off-diagonal term supervises the true insertion count over (s,t]:

$$\mu_i(s,t) := \#\{j : t_j^{\text{ins}} \in (s,t], j \in \text{gap } i\}$$

$\phi(a,b) := b - a + a \log \frac{a}{b}$ is the Poisson negative log likelihood minimized at $\arg\min_b \mathbb{E}[\phi(A,b)] = \mathbb{E}[A]$

Discrete Expand Operator

Inserting latent tokens per gap

Discrete expand operator inserts latent

$$\mathcal{E}_{s,t}(\boldsymbol{x}_s) := \boldsymbol{x}_s^{\epsilon} = \left[\bigoplus_{i=1}^{d(s)} \left(\boldsymbol{x}_0^{\ell_i} \oplus \boldsymbol{x}_{s_i}^{i}\right)\right] \oplus \boldsymbol{x}_0^{\ell_{d(s)+1}}$$

Insertion law matching predicted mean, capped at max length:

$$\ell_i \sim \mathrm{Binomial}\left(L - n_s,\ \frac{\hat{\mathcal{I}}_{s,t}(\boldsymbol{x}_s)[i]}{L - n_s}\right)$$

Many-step

$$\rho_{s,t} \ll 1$$

Converges to: $\ell_i \sim \mathrm{Poisson}(\hat{\mathcal{I}}_{s,t}(\boldsymbol{x}_s)[i])$

Equal to the infinitesimal jump law of the PDMP with intensity:

$$\lambda_t := \frac{\dot{\alpha}_t}{1-\alpha_t} \sum_i \mathcal{I}_t(\boldsymbol{x}_t)[i]$$

Few-step

$$\rho_{s,t} \to 1$$

As (s,t)→(0,1), the insertion expectation converges to the full length: $\sum_i \hat{\mathcal{I}}_{s,t}[i] \approx L$

Poisson fails (unbounded, high variance), but Binomial(n,p) has variance np(1-p) which vanishes as p→1

💡 **The Binomial law is what makes few-step, whole-sequence generation stable**

The Discrete Expanding Flow Map (DEFM)

Jointly insert and denoise toward the one-hot target

Mean denoiser on expanded state: $\psi_{s,t}(\mathcal{E}_{s,t}(\boldsymbol{x}_s)) := \mathcal{E}_{s,t}(\boldsymbol{x}_s) + (1-s)v_{s,t}(\mathcal{E}_{s,t}(\boldsymbol{x}_s))$

Discrete Expanding Flow Map (DEFM) with mean denoiser:

$$\Phi_{s,t}(\boldsymbol{x}_s) := X_{s,t}(\mathcal{E}_{s,t}(\boldsymbol{x}_s)) = \frac{1-t}{1-s}\mathcal{E}_{s,t}(\boldsymbol{x}_s) + \frac{t-s}{1-s}\psi_{s,t}(\mathcal{E}_{s,t}(\boldsymbol{x}_s))$$

Insert d(t)-d(s) tokens:

$$\mathcal{E}_{s,t}(\boldsymbol{x}_s) : \mathbb{R}^{d(s)\times V} \to \mathbb{R}^{d(t)\times V}$$

Denoise to time *t* with mean denoiser:

$$\psi_{s,t} : \mathbb{R}^{d(t)\times V} \to (\Delta^{V-1})^{d(t)}$$

The mean denoiser on the augmented space lies on the simplex
→ optimize with cross entropy

Cross-Entropy Objective

Semigroup consistency defines the objective for the discrete expanding flow map

Discrete Expanding Flow Map (DEFM) with mean denoiser:

$$\mathcal{L}_{\mathrm{DEFM}}(\hat{\psi}) := \mathbb{E}_{s,u,t}\mathbb{E}_{\boldsymbol{x}_0,\boldsymbol{x}_1}\left[-\sum_{i=1}^{d(t)}(\bar{\psi}_{s,t}\cdot\log\hat{\psi}_{s,t})(\boldsymbol{x}_s^{\epsilon})_i\right] + \mathbb{E}_t\mathbb{E}_{\boldsymbol{x}_0,\boldsymbol{x}_1}\left[-\sum_{i=1}^{d(t)}(D_t\cdot\log\hat{\psi}_{t,t})(\boldsymbol{x}_t^{i})\right]$$

Enforces semigroup consistency by matching the target:

Diagonal loss is standard CFM loss with teacher denoiser or self-distillation from the diagonal

$$\bar{\psi}_{s,t}(\boldsymbol{x}_s^{\epsilon}) := \mathrm{sg}\left[\omega_{s,u,t}\hat{\psi}_{s,u}(\boldsymbol{x}_s^{\epsilon}) + (1-\omega_{s,u,t})\hat{\psi}_{u,t}\left(\hat{\Phi}_{s,u}(\boldsymbol{x}_s^{\epsilon})\right)\right]$$

Augmented state

$$\omega_{s,u,t} := \frac{(u-s)(1-t)}{(t-s)(1-u)}$$

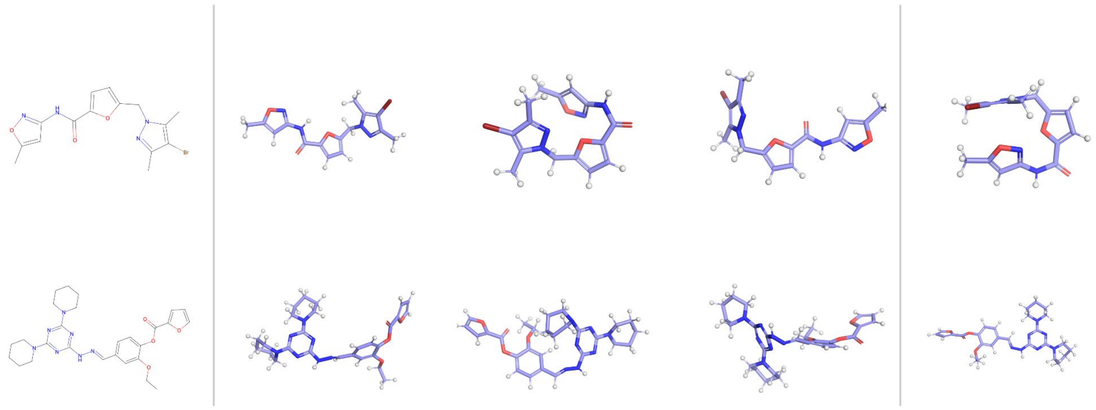
Together with the insertion loss, this trains the mean denoiser and the expand operator for few-step discrete generation

Experiments

Coarse-to-Fine Molecular Conformer Generation

From heavy atoms coordinate denoising to fine-grained hydrogen insertions

Molecular Graphs Generated Conformer Ground Truth



Setup: conformer as a coarse-to-fine continuous expanding interpolant. We first denoise heavy-atom backbone, freeze, then insert and denoise hydrogens around it with a single insertion flow with per-atom local times.

Table 1 *Molecular conformer generation results on GEOM-Drugs ($\delta = 1.25Å$) and GEOM-QM9 ($\delta = 0.5Å$)*

		GEOM-QM9 ($\delta = 0.5$ Å)				GEOM-Drugs ($\delta = 1.25$ Å)			
		COV-R(%)↑	MAT-R(Å)↓	COV-P(%)↑	MAT-P(Å)↓	COV-R(%)↑	MAT-R(Å)↓	COV-P(%)↑	MAT-P(Å)↓
Models	Steps↓	Mean/Median	Mean/Median	Mean/Median	Mean/Median	Mean/Median	Mean/Median	Mean/Median	Mean/Median
GeoDiff	500	87.80/93.66	0.3179/0.3216	46.25/45.02	0.6173/0.5112	64.12/75.56	1.1444/1.1296	43.16/42.02	1.3806/1.3314
SubgDiff	500	**89.40**/94.39	**0.2543**/0.2601	49.21/47.45	0.5030/0.4724	74.30/77.87	1.0003/0.9905	-/-	-/-
MSGEN	1000	88.36/**95.10**	0.2606/**0.2332**	49.54/48.39	0.4934/**0.4624**	85.06/**97.16**	0.9538/0.9478	49.28/48.20	1.3140/1.2719
EFlow	20	88.51/93.01	0.3849/0.3585	53.07/52.27	0.5185/0.4891	81.78/90.91	0.9971/0.9826	55.88/61.58	1.2672/1.2028
EFlow+R	30	88.87/94.05	0.3212/0.2924	54.86/52.38	**0.4876/0.4636**	85.80/93.85	0.9069/0.8855	64.05/70.38	1.1663/1.0997
EFM	1	69.21/69.23	0.4884/0.4635	32.18/29.09	0.6093/0.5794	63.28/72.47	1.1921/1.1721	31.03/26.09	1.4966/1.4454
EFM	2	72.83/74.25	0.4703/0.4436	34.09/31.33	0.6000/0.5654	77.21/88.67	1.0740/1.0499	47.14/49.24	1.3483/1.2966
EFM	4	83.62/86.91	0.4165/0.3954	48.76/45.26	0.5412/0.5159	81.26/91.13	1.0201/1.0029	53.37/56.72	1.2883/1.2354
EFM+R	14	87.79/92.01	0.3271/0.3049	**55.56/53.09**	0.4889/0.4691	**87.97**/95.45	**0.9029/0.8836**	**64.67/73.25**	**1.1575/1.0875**

On GEOM-QM9 and GEOM-Drugs:

- **EFlow:** 20 steps competitive with or surpassing 500–1000-step diffusion baselines
- **EFM (distilled):** accurate conformers in 4, 2, even 1 step

Discrete Molecular Graph Generation

Variable size graph generation by inserting and denoising nodes and edges

On QM9:

- **Multi-step vs. DeFoG:** at 100 steps EFlow matches DeFoG. From 100→10 steps EFlow degrades only 3.4× while DeFoG degrades 12×
- **Few-step vs. CFM:** at 1 step CFM's FCD degrades 4.4× (0.49→2.14) while EFM degrades only 15% (0.46→0.53) and uniqueness rises to 98.0

One framework robust across multi-step and one-step generation of variable-sized graphs

Table 2 ***Discrete molecular graph generation results.*** *Comparison against state-of-the-art multi-step baseline DeFoG (Qin et al., 2024) (top) and flow map baseline CFM (Roos et al., 2026) (bottom) on 10,000 molecules with FCD computed on the test split.*

Multi-Step	**DeFoG**			**EFlow** (ours)		
Steps	**Valid** ↑	**Unique** ↑	**FCD** ↓	**Valid** ↑	**Unique** ↑	**FCD** ↓
100	99.2	96.2	0.134	98.7	96.5	0.130
50	98.8	96.2	0.260	98.7	96.2	0.154
10	88.6	92.8	1.630	97.7	96.4	0.445
4	53.6	63.1	3.008	91.1	93.4	1.826
Few-Step	**CFM**			**EFM** (ours)		
Steps	**Valid** ↑	**Unique** ↑	**FCD** ↓	**Valid** ↑	**Unique** ↑	**FCD** ↓
2	91.1	97.6	0.49	88.9	97.9	0.46
1	90.0	91.8	2.14	87.5	98.0	0.53

Variable-Length Language Modeling

Generating sequences of increasing length

On LM1B:

- **Multi-step vs. FLM:** EFlow improves on the fixed-length FLM at every budget despite solving the harder variable-length problem
- **Few-step vs. distilled baselines:** EFM remains competitive and coherent at 2 steps

The expand–transport decomposition adds variable-length flexibility without sacrificing quality at matched budgets

Table 3 ***EFlow performance on LM1B.*** *Gen. PPL (↓) and entropy across sampling budgets. FLM results are computed by running publicly released checkpoints from Lee et al. (2026).*

	EFlow (ours)		FLM	
Steps	Gen. PPL (↓)	Ent.	Gen. PPL (↓)	Ent.
1024	**103.63**	4.16	111.36	4.30
512	**110.97**	4.16	114.89	4.31
264	**117.69**	4.16	120.19	4.32
128	**126.20**	4.15	129.06	4.34
64	**133.62**	4.14	143.73	4.36

steps = 128

EFlow

Gen. PPL: **54.19**
Entropy: **4.21**

[CLS] is scheduled to be monday. [SEP] they are appealing against two key yesterday. [SEP] i'm never sure how much i mean. [SEP] he said, " a friend has come a little down and has been making the most into again. " [SEP] by the time, last week, some 5, 000 tamil there were living in the, which on have year, an even the s at found. [SEP] the price of gasoline was 2. 7 % compared with the previous year. [SEP] the next morning mr cameron said that he would not be able to give london to for the expense of defence. [SEP] they study the verys so that [SEP]

steps = 2

EFM (Ours)

Gen. PPL: **60.8**
Entropy: **4.01**

[CLS] country. [CLS] but well it will not see after all in the high school of, tenn. [CLS] the most important of his first second in three years which has left over more on u. s. house. [CLS] it's do, and that's a good country in city high points. [CLS] well, one week, as well as he's trying to be up by half two points. [CLS] the 23 - year - old's house, they are more important is the country's own win, and when they are once in the game house because he is to work for the most next, but an [CLS]

Conclusions

1. We introduce **Expanding Generative Flows (EFlows)** and **Expanding Flow Maps (EFMs)** which recast generation as transport along an *expanding interpolant between marginals of increasing dimensionality*.
2. The **expand operator** lifts a state into a higher-dimensional space via *conditional noise augmentation* and the **transport map** pushes the expanded state toward the target. Fixed-dimensional flow maps are the identity-expand special case.
3. By extending continuous- and discrete-space **consistency identities**, EFMs enable efficient few-step generation across dimensionalities within a single formulation, removing the fixed-dimension limitation of flow-based generators.
4. One unified recipe validated on **3D molecular conformers, discrete molecular graphs, and language modeling**, from multi-step down to a single step.

Thanks for Listening! Questions?

Feel free to send questions to sophtang@engineering.upenn.edu!

Penn