Temporal Difference Flows (Farebrother et al., 2025)

Kellen Kanarios

July 12, 2025

Outline

1. Motivation
2. Flow Matching
3. Application to Reinforcement Learning

Variational inference with normalizing flows (Rezende & Mohamed)

Given data $\mathcal{X} \sim p(\mathbf{x})$. Want to learn $p(\mathbf{x})$.

- Assume there exists some joint $p(\mathbf{x}, \mathbf{z})$, $\mathbf{z} \sim p(\mathbf{z})$ is latent (no samples of $\mathbf{z}$). (we choose $p(\mathbf{z})$ i.e. $\mathcal{N}(0, 1)$).
- Maximize log-likelihood $\log p_\theta(\mathbf{x})$ via ELBO:

$$\log p_\theta(\mathbf{x}) \geq \mathbb{D}_{\mathrm{KL}}[q_\phi(\mathbf{z} \mid \mathbf{x}) \mid\mid p(\mathbf{z})] + \mathbb{E}_q[\log p_\theta(\mathbf{x} \mid \mathbf{z})]. \quad \text{(ELBO)}$$

- Simultaneously learn q_ϕ and p_θ

Variational inference with normalizing flows (Rezende & Mohamed)

Given data $\mathcal{X} \sim p(\mathbf{x})$. Want to learn $p(\mathbf{x})$.

- Assume there exists some joint $p(\mathbf{x}, \mathbf{z})$, $\mathbf{z} \sim p(\mathbf{z})$ is latent (no samples of $\mathbf{z}$). (we choose $p(\mathbf{z})$ i.e. $\mathcal{N}(0, 1)$).
- Maximize log-likelihood $\log p_\theta(\mathbf{x})$ via ELBO:

$$\log p_\theta(\mathbf{x}) \geq \mathbb{D}_{\mathrm{KL}}[q_\phi(\mathbf{z} \mid \mathbf{x}) \mid\mid p(\mathbf{z})] + \mathbb{E}_q[\log p_\theta(\mathbf{x} \mid \mathbf{z})]. \quad \text{(ELBO)}$$

- Simultaneously learn q_ϕ and p_θ

Problem. Recall $q_{\phi^*} = \arg\max_\phi \mathrm{ELBO}(\phi) = p_\theta(z \mid x)$.

- Need parametric family q_ϕ to contain $p_\theta(\cdot \mid \mathbf{x})$
- Rarely the case.

Variational inference with normalizing flows (Rezende & Mohamed)

Given data $\mathcal{X} \sim p(\mathbf{x})$. Want to learn $p(\mathbf{x})$.

- Assume there exists some joint $p(\mathbf{x}, \mathbf{z})$, $\mathbf{z} \sim p(\mathbf{z})$ is latent (no samples of $\mathbf{z}$). (we choose $p(\mathbf{z})$ i.e. $\mathcal{N}(0, 1)$).
- Maximize log-likelihood $\log p_\theta(\mathbf{x})$ via ELBO:

$$\log p_\theta(\mathbf{x}) \geq \mathbb{D}_{\mathrm{KL}}[q_\phi(\mathbf{z} \mid \mathbf{x}) \,||\, p(\mathbf{z})] + \mathbb{E}_q[\log p_\theta(\mathbf{x} \mid \mathbf{z})]. \quad \text{(ELBO)}$$

- Simultaneously learn q_ϕ and p_θ

Problem. Recall $q_{\phi^*} = \arg\max_\phi \mathrm{ELBO}(\phi) = p_\theta(z \mid x)$.

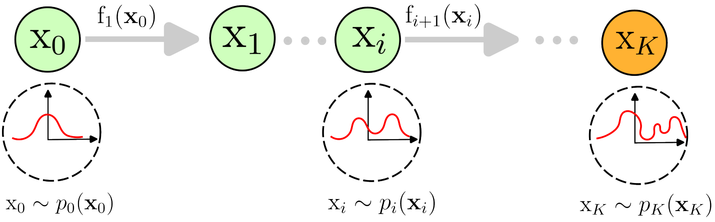
- Need parametric family q_ϕ to contain $p_\theta(\cdot \mid \mathbf{x})$
- Rarely the case.

How do we create a more expressive family that we can still optimize?

Given target distribution $p(\mathbf{x})$. Pick some initial distribution p_0, where we can sample $\mathbf{x}_0 \sim p_0$.

Idea. *Learn sequence of functions $f_1, f_2, \ldots, f_K$, such that*

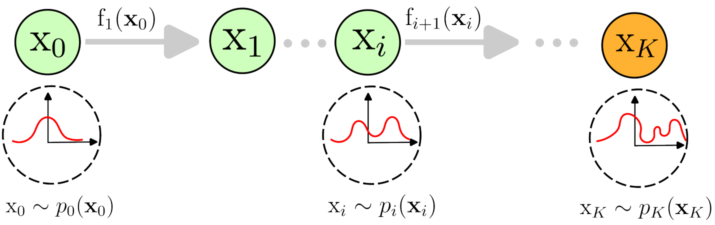
$$(f_K \circ f_{K-1} \cdots \circ f_1)(\mathbf{x}_0) \sim p$$



Given target distribution $p(\mathbf{x})$. Pick some initial distribution p_0, where we can sample $\mathbf{x}_0 \sim p_0$.

Idea. *Learn sequence of functions $f_1, f_2, \ldots, f_K$, such that*

$$(f_K \circ f_{K-1} \cdots \circ f_1)(\mathbf{x}_0) \sim p$$

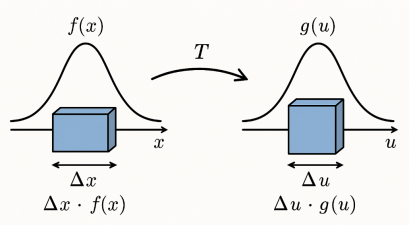


How do we learn such a sequence of functions f_i?

Given **invertible**, **smooth both ways** function $T : \mathbb{R}^d \to \mathbb{R}^d$.

Change of variables theorem. For $X, Y \subset \mathbb{R}^d$, $T : X \to Y$

$$\int_X p_0(\mathbf{x}) d\mathbf{x} = \int_{T^{-1}(Y)} p_0(T^{-1}(\mathbf{y})) \left| \det \frac{\partial T}{\partial y} \right|^{-1} d\mathbf{y}.$$



$$\Rightarrow f_1(\mathbf{x}_0) \sim p_0(\mathbf{x}_0) \left| \det \frac{\partial f_1}{\partial \mathbf{x}} \right|^{-1}$$

$$\Rightarrow \log p_K(\mathbf{x}_k) = \log p_0(\mathbf{x}_0) - \sum_{k=1}^{K} \left| \det \frac{\partial f_k}{\partial \mathbf{x}_{k-1}} \right|^{-1}$$

Re. $\log p_K(\mathbf{x}_k) = \log p_0(\mathbf{x}_0) - \sum_{k=1}^{K} \left|\det \frac{\partial f_k}{\partial \mathbf{x}_{k-1}}\right|^{-1}$

1. Given $\mathbf{x}^{(i)}$ from dataset, compute $\mathbf{x}_0^{(i)} = (f_K \circ \cdots \circ f_1)^{-1}(\mathbf{x}^{(i)})$

Re. $\log p_K(\mathbf{x}_k) = \log p_0(\mathbf{x}_0) - \sum_{k=1}^{K} \left|\det \frac{\partial f_k}{\partial \mathbf{x}_{k-1}}\right|^{-1}$

1. Given $\mathbf{x}^{(i)}$ from dataset, compute $\mathbf{x}_0^{(i)} = (f_K \circ \cdots \circ f_1)^{-1}(\mathbf{x}^{(i)})$
2. Maximize log-likelihood

$$\max_{\theta} \left[\log p_0(\mathbf{x}_0^{(i)}(\theta)) - \sum_{k=1}^{K} \left|\det \frac{\partial f_k(\theta)}{\partial \mathbf{x}_{k-1}^{(i)}}\right|^{-1}\right].$$

Re. $\log p_K(\mathbf{x}_k) = \log p_0(\mathbf{x}_0) - \sum_{k=1}^{K} \left|\det \frac{\partial f_k}{\partial \mathbf{x}_{k-1}}\right|^{-1}$

1. Given $\mathbf{x}^{(i)}$ from dataset, compute $\mathbf{x}_0^{(i)} = (f_K \circ \cdots \circ f_1)^{-1}(\mathbf{x}^{(i)})$
2. Maximize log-likelihood

$O(LD^3)$

$$\max_\theta \left[\log p_0(\mathbf{x}_0^{(i)}(\theta)) - \sum_{k=1}^{K} \boxed{\left|\det \frac{\partial f_k(\theta)}{\partial \mathbf{x}_{k-1}^{(i)}}\right|^{-1}}\right].$$

Re. $\log p_K(\mathbf{x}_k) = \log p_0(\mathbf{x}_0) - \sum_{k=1}^{K} \left|\det \frac{\partial f_k}{\partial \mathbf{x}_{k-1}}\right|^{-1}$

1. Given $\mathbf{x}^{(i)}$ from dataset, compute $\mathbf{x}_0^{(i)} = (f_K \circ \cdots \circ f_1)^{-1}(\mathbf{x}^{(i)})$
2. Maximize log-likelihood

$O(LD^3)$

$$\max_{\theta} \left[\log p_0(\mathbf{x}_0^{(i)}(\theta)) - \sum_{k=1}^{K} \boxed{\left|\det \frac{\partial f_k(\theta)}{\partial \mathbf{x}_{k-1}^{(i)}}\right|^{-1}}\right].$$

Need to pick **simple** transformations i.e.

$$f(\mathbf{x}) = \mathbf{x} + \mathbf{u}h(\mathbf{w}^\top \mathbf{x} + b).$$

Require **very large** K to represent complex distributions.

Re. $\log p_K(\mathbf{x}_k) = \log p_0(\mathbf{x}_0) - \sum_{k=1}^{K} \left|\det \frac{\partial f_k}{\partial \mathbf{x}_{k-1}}\right|^{-1}$

1. Given $\mathbf{x}^{(i)}$ from dataset, compute $\mathbf{x}_0^{(i)} = (f_K \circ \cdots \circ f_1)^{-1}(\mathbf{x}^{(i)})$
2. Maximize log-likelihood

$O(LD^3)$

$$\max_{\theta} \left[\log p_0(\mathbf{x}_0^{(i)}(\theta)) - \sum_{k=1}^{K} \boxed{\left|\det \frac{\partial f_k(\theta)}{\partial \mathbf{x}_{k-1}^{(i)}}\right|^{-1}} \right].$$

Need to pick **simple** transformations i.e.

$$f(\mathbf{x}) = \mathbf{x} + \mathbf{u}h(\mathbf{w}^\top \mathbf{x} + b).$$

Require **very large** K to represent complex distributions.

Why not take $K \to \infty$...

Neural Ordinary Differential Equations (Chen et al.)

Replace $\mathbf{x}_{t+1} = f_t(\mathbf{x}_t)$ with the **ODE**

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = f_t(\mathbf{x}_t).$$

Remark. *Instantaneous COV*

$$\frac{\partial \log p_t(\mathbf{x}_t)}{\partial t} = -\mathrm{tr}\left(\frac{df}{d\mathbf{x}_t}\right)$$

Neural Ordinary Differential Equations (Chen et al.)

Replace $\mathbf{x}_{t+1} = f_t(\mathbf{x}_t)$ with the **ODE**

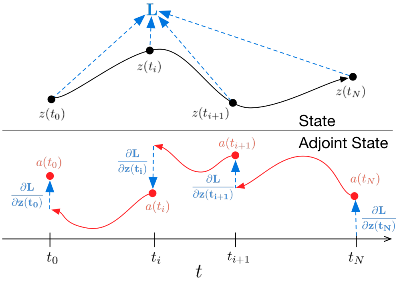
$$\frac{d\mathbf{x}}{dt} = f_t(\mathbf{x}_t).$$

Remark. *Instantaneous COV*

$$\frac{\partial \log p_t(\mathbf{x}_t)}{\partial t} = -\mathrm{tr}\left(\frac{df}{d\mathbf{x}_t}\right)$$

Given $\mathbf{x}^{(i)}$ from dataset, now must compute $\mathbf{x}_0^{(i)} = \int_1^0 f_t^{-1}(\mathbf{x}_t^{(i)})dt$

- Requires **exact** ODE solve for unbiased gradient.



Neural Ordinary Differential Equations (Chen et al.)

Replace $\mathbf{x}_{t+1} = f_t(\mathbf{x}_t)$ with the **ODE**

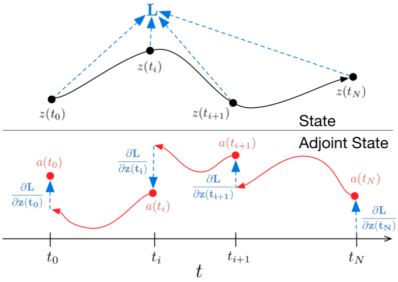
$$\frac{d\mathbf{x}}{dt} = f_t(\mathbf{x}_t).$$

Remark. *Instantaneous COV*

$$\frac{\partial \log p_t(\mathbf{x}_t)}{\partial t} = -\mathrm{tr}\left(\frac{df}{d\mathbf{x}_t}\right)$$

Given $\mathbf{x}^{(i)}$ from dataset, now must compute $\mathbf{x}_0^{(i)} = \int_1^0 f_t^{-1}(\mathbf{x}_t^{(i)})dt$

- Requires **exact** ODE solve for unbiased gradient.

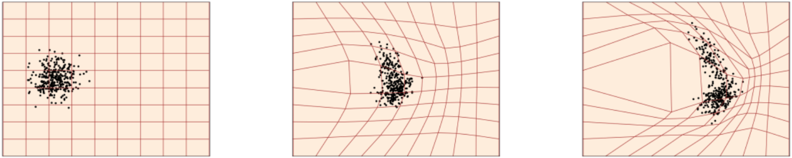


Can we learn f_t without backprop through an ODE?

Disclaimer. f_t will now often be referred to as the *flow*.

Book keeping. We say the *flow* f_t *generates* a probability path p_t if $X_t = f_t(X_0) \sim p_t$. Equivalently,

$$p_t(x) = [f_{t\sharp} p_0](x) \triangleq p_0(f_t^{-1}(y)) \left|\det \partial_y f_t^{-1}(y)\right| .$$



Key Idea. Do **not** learn the *flow* f_t, learn the **velocity** of the flow.

Def. This velocity is a *vector field* $u_t : \mathbb{R}^d \to \mathbb{R}^d$, such that

$$\frac{\mathrm{d}f_t(\mathbf{x})}{\mathrm{d}t} = u_t(f_t(\mathbf{x})).$$

Key Idea. Do **not** learn the *flow* f_t, learn the **velocity** of the flow.

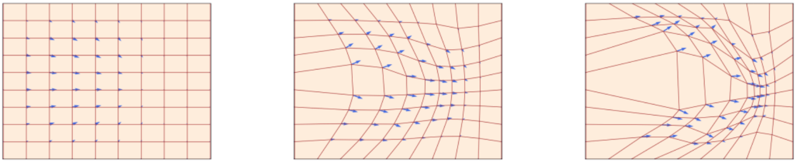
Def. This velocity is a *vector field* $u_t : \mathbb{R}^d \to \mathbb{R}^d$, such that

$$\frac{\mathrm{d}f_t(\mathbf{x})}{\mathrm{d}t} = u_t(f_t(\mathbf{x})).$$

To sample, solve ODE at **sample time** i.e.

$$f_{t+\Delta t}(\mathbf{x}) \approx f_t(\mathbf{x}) + \Delta t \cdot u_t(f_t(\mathbf{x})) \qquad \text{(Euler method)}$$

By uniqueness of ODE, (vector field u_t) $\leftrightarrow$ (flow f_t).



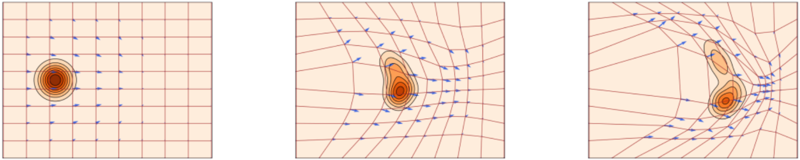
Def. A *vector field* u_t *generates* a probability path p_t if its corresponding flow f_t *generates* p_t.

Thm. A *vector field* u_t *generates* a probability path p_t if and only if it satisfies the *continuity equation*.

Continuity Equation.

$$\frac{\mathrm{d}}{\mathrm{d}t} p_t(x) + \mathrm{div}(p_t u_t)(x) = 0,$$

where $\mathrm{div}(v)(x) = \sum_{i=1}^{d} \partial_{x^i} v^i(x)$, and $v(x) = (v^1(x), \dots, v^d(x))$.



Continuity Equation.

$$\frac{\mathrm{d}}{\mathrm{d}t}p_t(x) + \mathrm{div}(p_t u_t)(x) = 0,$$

where $\mathrm{div}(v)(x) = \sum_{i=1}^{d} \partial_{x^i} v^i(x)$, and $v(x) = (v^1(x), \dots, v^d(x))$.

Proof: (If people care)

$$\frac{d}{dt}\mathbb{E}f(X_t) = \frac{d}{dt}\int_{\mathbb{R}^d} f(x)p_t(x)dx = \int_{\mathbb{R}^d} f(x)\partial_t p_t(x)dx,$$

$$\begin{aligned}
\mathbb{E}\frac{d}{dt}f(X_t) &= \int_{\mathbb{R}^d} \left(\nabla f(x_t)\cdot v_t(x_t)\right) p_t(x_t)dx_t \\
&= 0 - \int_{\mathbb{R}^d} f(x_t)\mathrm{div}(v_t(x_t)p_t(x_t))dx_t \qquad \text{(IBP)} \\
&= \int_{\mathbb{R}^d} -f(x)\mathrm{div}(v_t(x)p_t(x))dx..
\end{aligned}$$

The result follows from fundamental lemma of calculus of variations.

How do we actually learn the vector field u_t?

Flow Model Loss.

$$\mathcal{L}_{\text{FM}}(\theta) = \mathbb{E}_{t, p_t(x)} \|v_t(x;\theta) - u_t(x)\|^2$$

- $p_t : \mathcal{X} \to [0,1]$: probability density path.
- $u_t : \mathbb{R}^d \to \mathbb{R}^d$: vector field that *generates* p_t.

Flow Model Loss.

$$\mathcal{L}_{\text{FM}}(\theta) = \mathbb{E}_{t,p_t(x)} \left\| v_t(x;\theta) - u_t(x) \right\|^2$$

- $p_t : \mathcal{X} \to [0,1]$: probability density path.
- $u_t : \mathbb{R}^d \to \mathbb{R}^d$: vector field that *generates* p_t.

Intractable.

1. How to choose p_t?

Flow Model Loss.

$$\mathcal{L}_{\text{FM}}(\theta) = \mathbb{E}_{t, p_t(x)} \left\| v_t(x; \theta) - u_t(x) \right\|^2$$

- $p_t : \mathcal{X} \to [0, 1]$: probability density path.
- $u_t : \mathbb{R}^d \to \mathbb{R}^d$: vector field that *generates* p_t.

Intractable.

1. How to choose p_t?
2. Given p_t need to solve continuity equation for u_t, which is a **PDE** likely without close-form.

Key Idea. *Given the endpoint X_1, we can easily construct a path between X and X_1.*

Ex: *Consider*

$$f_t(x \mid x_1) = (1 - t)x + tx_1$$

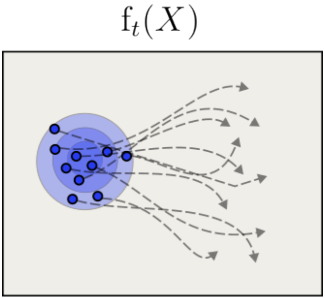


Figure: Without target **how** do we construct p_t, f_t?

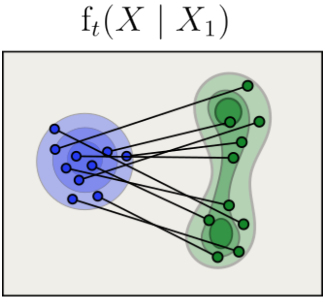


Figure: With target, make f_t **line** to known x_1.

What does the conditional flow being nice have to do with our problem?

Idea 1: Given target data distribution q, approximate q by marginalizing "nice" conditionals by q i.e.

$$p_t(\mathbf{x}) = \int p_t(\mathbf{x} \mid \mathbf{x}_1) q(\mathbf{x}_1) d\mathbf{x}_1. \qquad \text{(Marginal Density)}$$

For $p_1(\mathbf{x} \mid \mathbf{x}_1) \sim N(x_1, \sigma_1)$, with $\sigma_1 \ll 1$, $p_1(\mathbf{x}) \approx q(\mathbf{x})$.

Idea 1: Given target data distribution q, approximate q by marginalizing "nice" conditionals by q i.e.

$$p_t(\mathbf{x}) = \int p_t(\mathbf{x} \mid \mathbf{x}_1) q(\mathbf{x}_1) \mathrm{d}\mathbf{x}_1. \qquad \text{(Marginal Density)}$$

For $p_1(\mathbf{x} \mid \mathbf{x}_1) \sim N(x_1, \sigma_1)$, with $\sigma_1 \ll 1$, $p_1(\mathbf{x}) \approx q(\mathbf{x})$.

Idea 2: Use this to define an approximate vector field

$$u_t(\mathbf{x}) = \int u_t(\mathbf{x} \mid \mathbf{x}_1) \frac{p_t(\mathbf{x} \mid \mathbf{x}_1) q(\mathbf{x}_1)}{p_t(\mathbf{x})} \mathrm{d}\mathbf{x}_1. \qquad \text{(Marginal Vector Field)}$$

Here, $u_t(\mathbf{x} \mid \mathbf{x}_1)$ is the conditional vector field that generates $p_t(\mathbf{x} \mid \mathbf{x}_1)$.

Idea 1: Given target data distribution q, approximate q by marginalizing "nice" conditionals by q i.e.

$$p_t(\mathbf{x}) = \int p_t(\mathbf{x} \mid \mathbf{x}_1) q(\mathbf{x}_1) \mathrm{d}\mathbf{x}_1. \qquad \text{(Marginal Density)}$$

For $p_1(\mathbf{x} \mid \mathbf{x}_1) \sim N(x_1, \sigma_1)$, with $\sigma_1 \ll 1$, $p_1(\mathbf{x}) \approx q(\mathbf{x})$.

Idea 2: Use this to define an approximate vector field

$$u_t(\mathbf{x}) = \int u_t(\mathbf{x} \mid \mathbf{x}_1) \frac{p_t(\mathbf{x} \mid \mathbf{x}_1) q(\mathbf{x}_1)}{p_t(\mathbf{x})} \mathrm{d}\mathbf{x}_1. \quad \text{(Marginal Vector Field)}$$

Here, $u_t(\mathbf{x} \mid \mathbf{x}_1)$ is the conditional vector field that generates $p_t(\mathbf{x} \mid \mathbf{x}_1)$.

Thm. *The marginal vector field u_t generates the marginal probability path p_t.* (Check continuity equation)

Conditional Flow Model Loss.

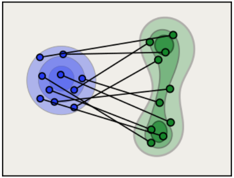
$$\mathcal{L}_{\mathsf{CFM}}(\theta) = \mathbb{E}_{t,q(x_1),p_t(x|x_1)} \left\| v_t(x;\theta) - u_t(x \mid x_1) \right\|^2$$

Conditional Flow Model Loss.

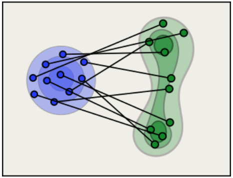
$$\mathcal{L}_{\text{CFM}}(\theta) = \mathbb{E}_{t,q(x_1),p_t(x|x_1)} \left\| v_t(x;\theta) - u_t(x \mid x_1) \right\|^2$$

Thm. *Up to a constant independent of* θ*,* $\mathcal{L}_{FM}(\theta) = \mathcal{L}_{CFM}(\theta)$*. In particular,* $\nabla \mathcal{L}_{FM}(\theta) = \nabla \mathcal{L}_{CFM}(\theta)$.

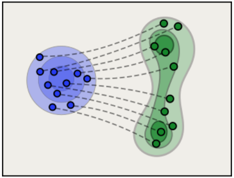
Batch i



Batch j



$\mathbb{E}[$**Batch**$]$



How do we actually use this?

Example: Optimal Transport

For $x \sim \mathcal{N}(0,1)$,

$$f_t(x) = \mu_t(x_1) + x\sigma_t(x_1) \implies f_t(x) \sim \mathcal{N}(\mu_t(x), \sigma_t(x).$$

Consider

$$\mu_t = tx_1, \quad \sigma_t(x) = 1 - (1 - \sigma_{\text{min}})t,$$

such that

Example: Optimal Transport

For $x \sim \mathcal{N}(0, 1)$,

$$f_t(x) = \mu_t(x_1) + x\sigma_t(x_1) \implies f_t(x) \sim \mathcal{N}(\mu_t(x), \sigma_t(x).$$

Consider

$$\mu_t = tx_1, \quad \sigma_t(x) = 1 - (1 - \sigma_{\mathrm{min}})t,$$

such that

$$f_t^{-1}(x) = \frac{x - tx_1}{1 - (1 - \sigma_{\mathrm{min}})t}, \quad \frac{\mathrm{d}}{\mathrm{d}t} f_t(x) = x_1 - (1 - \sigma_{\mathrm{min}})x.$$

Example: Optimal Transport

For $x \sim \mathcal{N}(0,1)$,

$$f_t(x) = \mu_t(x_1) + x\sigma_t(x_1) \implies f_t(x) \sim \mathcal{N}(\mu_t(x), \sigma_t(x)).$$

Consider

$$\mu_t = tx_1, \quad \sigma_t(x) = 1 - (1 - \sigma_{\text{min}})t,$$

such that

$$f_t^{-1}(x) = \frac{x - tx_1}{1 - (1 - \sigma_{\text{min}})t}, \quad \frac{\mathrm{d}}{\mathrm{d}t} f_t(x) = x_1 - (1 - \sigma_{\text{min}})x.$$

Recall

$$\frac{\mathrm{d}}{\mathrm{d}t} f_t(x) = u(f_t(x) \mid x_1) \implies u(x \mid x_1) = \frac{\mathrm{d}}{\mathrm{d}t} f_t(f_t^{-1}(x)).$$

Example: Optimal Transport

$$f_t^{-1}(x) = \frac{x - tx_1}{1 - (1 - \sigma_{\text{min}})t}, \quad \frac{\mathrm{d}}{\mathrm{d}t} f_t(x) = x_1 - (1 - \sigma_{\text{min}})x.$$

Recall

$$\frac{\mathrm{d}}{\mathrm{d}t} f_t(x) = u(f_t(x) \mid x_1) \implies u(x \mid x_1) = \frac{\mathrm{d}}{\mathrm{d}t} f_t(f_t^{-1}(x)).$$

Thus,

$$u_t(x \mid x_1) = \frac{x_1 - (1 - \sigma_{\text{min}})x}{1 - (1 - \sigma_{\text{min}})t}.$$

Example: Optimal Transport

$$f_t^{-1}(x) = \frac{x - tx_1}{1 - (1 - \sigma_{\text{min}})t}, \quad \frac{\mathrm{d}}{\mathrm{d}t} f_t(x) = x_1 - (1 - \sigma_{\text{min}})x.$$

Recall

$$\frac{\mathrm{d}}{\mathrm{d}t} f_t(x) = u(f_t(x) \mid x_1) \implies u(x \mid x_1) = \frac{\mathrm{d}}{\mathrm{d}t} f_t(f_t^{-1}(x)).$$

Thus,

$$u_t(x \mid x_1) = \frac{x_1 - (1 - \sigma_{\text{min}})x}{1 - (1 - \sigma_{\text{min}})t}.$$

$$\boxed{\mathcal{L}_{\text{CFM}}(\theta) = \mathbb{E}_{t, q(x_1), p_t(x|x_1)} \left\| v_t(x; \theta) - \frac{x_1 - (1 - \sigma_{\text{min}})x}{1 - (1 - \sigma_{\text{min}})t} \right\|^2}$$

Diffusion Meets Flow Matching: Two Sides of the Same Coin (Gao et al., 2024)

Recall that instead of an ODE, in diffusion, we have a **SDE**

$$\mathrm{d}\mathbf{x}_t = f_t(\mathbf{x}_t)\mathrm{d}t + \sigma(\mathbf{x}_t)\mathrm{d}B_t.$$

However, for *OU* process, we can actually absorb Brownian motion term and get vector field

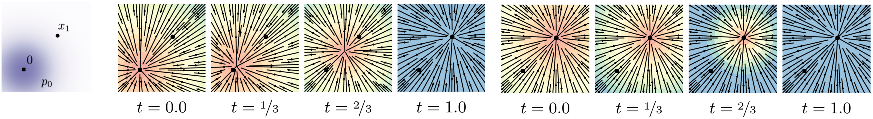
$$u_t(\mathbf{x}_t) = -(\mathbf{x}_t + \nabla \ln p_t(\mathbf{x}_t)).$$

Similarly, learning the score function $\nabla \ln p_t(\mathbf{x}_t)$ can be rewritten as the flow matching objective for certain choices of p_t see (Lipman et al., 2024).

Why Flows?

Anecdotally.

- *The OT path's conditional vector field has constant direction in time and is arguably simpler to fit with a parametric model. (Lipman et al., 2023)*
- *The deterministic nature of ODEs equips flow-matching methods with simpler learning objectives and faster inference speed (Zheng et al., 2025)*

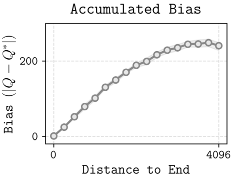


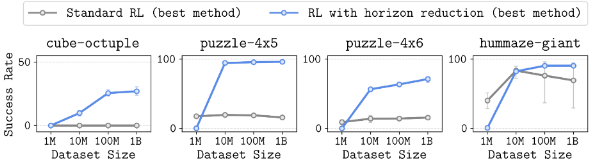
Conditional score Conditional vector field

Q learning is not yet scalable (Park, 2025)

Long horizon problems are hard.

$$\mathbb{E}_{(s,a,r,s')\sim\mathcal{D}}\left[\left(Q_\theta(s,a) - \underbrace{\left(r + \gamma \max_{a'} Q_{\bar{\theta}}(s',a')\right)}_{\texttt{Biased}}\right)^2\right].$$





Temporal Difference Flows (Farebrother et al., 2025)

Idea. Apply flow matching to learn the *successor measure* of an MDP.

Temporal Difference Flows (Farebrother et al., 2025)

Idea. Apply flow matching to learn the *successor measure* of an MDP.

Def. We define the *successor measure* as

$$m^{\pi}(X \mid s, a) = (1-\gamma) \sum_{k=0}^{\infty} \gamma^{k} \Pr(S_{k+1} \in X \mid S_0 = s, A_0 = a, \pi),$$

Temporal Difference Flows (Farebrother et al., 2025)

Idea. Apply flow matching to learn the *successor measure* of an MDP.

Def. We define the *successor measure* as

$$m^\pi(\mathrm{X} \mid s, a) = (1-\gamma) \sum_{k=0}^{\infty} \gamma^k \Pr(S_{k+1} \in \mathrm{X} \mid S_0 = s, A_0 = a, \pi),$$

Recall. The *successor measure* is the unique fix point to the *Bellman* equation

$$\begin{aligned} m^\pi(\cdot \mid s, a) &= (\mathcal{T}^\pi m^\pi)(\cdot \mid s, a) \\ &:= (1-\gamma) P(\cdot \mid s, a) + \gamma (P^\pi m^\pi)(\cdot \mid s, a), \end{aligned}$$

where

$$(P^\pi m)(\mathrm{d}x \mid s, a) = \int_{s'} P(\mathrm{d}s' \mid s, a) m(\mathrm{d}x \mid s', \pi(s'))).$$

The most straightforward idea is to just substitute m^{π} for q in flow matching

The most straightforward idea is to just substitute m^π for q in flow matching i.e.

$$\mathcal{L}_{\text{MC-CFM}}(\theta) = \mathbb{E}_{\rho,t,Z,X_t}\left[\left\|\tilde{v}_t(X_t \mid S, A; \theta) - u_t(X_t \mid Z)\right\|^2\right],$$

where $Z = X_1 \sim m^\pi(\cdot \mid S, A), X_t \sim p_t(\cdot \mid Z)$.

Here we just use the optimal transport conditional vector field and corresponding probability density path.

The most straightforward idea is to just substitute m^π for q in flow matching i.e.

$$\mathcal{L}_{\text{MC-CFM}}(\theta) = \mathbb{E}_{\rho,t,Z,X_t}\left[\left\|\tilde{v}_t(X_t \mid S, A; \theta) - u_t(X_t \mid Z)\right\|^2\right],$$

where $Z = X_1 \sim m^\pi(\cdot \mid S, A), X_t \sim p_t(\cdot \mid Z)$.

Here we just use the optimal transport conditional vector field and corresponding probability density path.

This requires direct access to samples from m^π.

Can we learn from offline one-step transitions (S, A, S')?

The most straightforward idea is to just substitute m^π for q in flow matching i.e.

$$\mathcal{L}_{\text{MC-CFM}}(\theta) = \mathbb{E}_{\rho,t,Z,X_t}\left[\left\|\tilde{v}_t(X_t \mid S, A; \theta) - u_t(X_t \mid Z)\right\|^2\right],$$

where $Z = X_1 \sim m^\pi(\cdot \mid S, A), X_t \sim p_t(\cdot \mid Z)$.

Here we just use the optimal transport conditional vector field and corresponding probability density path.

This requires direct access to samples from m^π.

Can we learn from offline one-step transitions (S, A, S')?

Leverage **recursive** structure of Bellman equation i.e.

$$X_0 \sim p_0$$
$$Z = X_1 \sim (1-\gamma)\delta_{S'} + \gamma\delta_{\tilde{f}_1(X_0 \mid S', \pi(S'))}. \qquad \text{(TD-CFM)}$$

Leverage **recursive** structure of Bellman equation i.e.

$$X_0 \sim p_0$$
$$Z = X_1 \sim (1-\gamma)\delta_{S'} + \gamma\delta_{\tilde{f}_1(X_0|S',\pi(S'))}.$$

- With probability $(1-\gamma)$, $X_1 = S'$
- With probability γ, sample from $\tilde{m}^\pi$ by integrating $\tilde{f}_t$.

Can we do better?

Lemma. Let v_t^1 and v_t^2 be vector fields that generate the probability paths p_t^1 and p_t^2, respectively. Then, for any $\gamma \in [0,1]$, the mixture probability path $p_t = (1-\gamma)p_t^1 + \gamma p_t^2$ is generated by the vector field.

$$v_t := \frac{(1-\gamma)p_t^1 v_t^1 + \gamma p_t^2 v_t^2}{(1-\gamma)p_t^1 + \gamma p_t^2}.$$

Lemma. Let v_t^1 and v_t^2 be vector fields that generate the probability paths p_t^1 and p_t^2, respectively. Then, for any $\gamma \in [0, 1]$, the mixture probability path $p_t = (1-\gamma)p_t^1 + \gamma p_t^2$ is generated by the vector field.

$$v_t := \frac{(1-\gamma)p_t^1 v_t^1 + \gamma p_t^2 v_t^2}{(1-\gamma)p_t^1 + \gamma p_t^2}.$$

Remember. The *successor measure* is the unique fix point to the *Bellman* equation

$$m^\pi(\cdot \mid s, a) = (1-\gamma)\underbrace{P(\cdot \mid s, a)}_{p_t^1} + \gamma \underbrace{(P^\pi m^\pi)(\cdot \mid s, a)}_{p_t^2},$$

Lemma. Let v_t^1 and v_t^2 be vector fields that generate the probability paths p_t^1 and p_t^2, respectively. Then, for any $\gamma \in [0,1]$, the mixture probability path $p_t = (1-\gamma)p_t^1 + \gamma p_t^2$ is generated by the vector field.

$$v_t := \frac{(1-\gamma)p_t^1 v_t^1 + \gamma p_t^2 v_t^2}{(1-\gamma)p_t^1 + \gamma p_t^2}.$$

Remember. The *successor measure* is the unique fix point to the *Bellman* equation

$$m^\pi(\cdot \mid s,a) = (1-\gamma)\underbrace{P(\cdot \mid s,a)}_{p_t^1} + \gamma \underbrace{(P^\pi m^\pi)(\cdot \mid s,a)}_{p_t^2},$$

Idea. Learn v_t for $p_t^1 = P(\cdot \mid S,A), p_t^2 = (P^\pi m)(\cdot \mid S,A)$

Lemma. Let v_t^1 and v_t^2 be vector fields that generate the probability paths p_t^1 and p_t^2, respectively. For $\gamma \in [0, 1]$, the vector field $v_t = \frac{(1-\gamma)p_t^1 v_t^1 + \gamma p_t^2 v_t^2}{(1-\gamma)p_t^1 + \gamma p_t^2}$ satisfies

$$v_t = \underset{v:\mathbb{R}^d \to \mathbb{R}^d}{\arg\min} \Big\{ (1-\gamma)\mathbb{E}_{x_t \sim p_t^1} \left[||v_t(x_t) - v_t^1(x_t)||^2 \right] + \gamma \mathbb{E}_{x_t \sim p_t^2} \left[||v_t(x_t) - v_t^2(x_t)||^2 \right] \Big\}.$$

Lemma. Let v_t^1 and v_t^2 be vector fields that generate the probability paths p_t^1 and p_t^2, respectively. For $\gamma \in [0,1]$, the vector field $v_t = \frac{(1-\gamma)p_t^1 v_t^1 + \gamma p_t^2 v_t^2}{(1-\gamma)p_t^1 + \gamma p_t^2}$ satisfies

$$v_t = \underset{v:\mathbb{R}^d \to \mathbb{R}^d}{\arg\min} \Big\{ (1-\gamma)\mathbb{E}_{x_t \sim p_t^1} \left[||v_t(x_t) - v_t^1(x_t)||^2 \right] + \gamma \mathbb{E}_{x_t \sim p_t^2} \left[||v_t(x_t) - v_t^2(x_t)||^2 \right] \Big\}.$$

Substitute.

$$v_t = \underset{v:\mathbb{R}^d \to \mathbb{R}^d}{\arg\min} \Big\{ (1-\gamma)\overbrace{\mathbb{E}_{x_t \sim P} \left[||v_t(x_t) - v_t^1(x_t)||^2 \right]}^{\mathcal{L}_{\text{OS}}} + \gamma \underbrace{\mathbb{E}_{x_t \sim P^{\pi} m} \left[||v_t(x_t) - v_t^2(x_t)||^2 \right]}_{\mathcal{L}_{\text{GHM}}} \Big\}.$$

Lemma. Let v_t^1 and v_t^2 be vector fields that generate the probability paths p_t^1 and p_t^2, respectively. For $\gamma \in [0,1]$, the vector field $v_t = \frac{(1-\gamma)p_t^1 v_t^1 + \gamma p_t^2 v_t^2}{(1-\gamma)p_t^1 + \gamma p_t^2}$ satisfies

$$v_t = \underset{v:\mathbb{R}^d \to \mathbb{R}^d}{\arg\min} \Big\{ (1-\gamma)\mathbb{E}_{x_t \sim p_t^1}\left[||v_t(x_t) - v_t^1(x_t)||^2\right] + \gamma \mathbb{E}_{x_t \sim p_t^2}\left[||v_t(x_t) - v_t^2(x_t)||^2\right] \Big\}.$$

Substitute.

$$v_t = \underset{v:\mathbb{R}^d \to \mathbb{R}^d}{\arg\min} \Big\{ (1-\gamma)\overbrace{\mathbb{E}_{x_t \sim P}\left[||v_t(x_t) - v_t^1(x_t)||^2\right]}^{\mathcal{L}_{\text{OS}}} + \gamma \underbrace{\mathbb{E}_{x_t \sim P^{\pi}m}\left[||v_t(x_t) - v_t^2(x_t)||^2\right]}_{\mathcal{L}_{\text{GHM}}} \Big\}.$$

How do we get v_t^1 and v_t^2?

Lemma. Let v_t^1 and v_t^2 be vector fields that generate the probability paths p_t^1 and p_t^2, respectively. For $\gamma \in [0,1]$, the vector field $v_t = \frac{(1-\gamma)p_t^1 v_t^1 + \gamma p_t^2 v_t^2}{(1-\gamma)p_t^1 + \gamma p_t^2}$ satisfies

$$v_t = \operatorname*{arg\,min}_{v:\mathbb{R}^d \to \mathbb{R}^d} \Big\{ (1-\gamma)\mathbb{E}_{x_t \sim p_t^1}\left[||v_t(x_t) - v_t^1(x_t)||^2\right] + \gamma \mathbb{E}_{x_t \sim p_t^2}\left[||v_t(x_t) - v_t^2(x_t)||^2\right] \Big\}.$$

Substitute.

$$v_t = \operatorname*{arg\,min}_{v:\mathbb{R}^d \to \mathbb{R}^d} \Big\{ (1-\gamma)\overbrace{\mathbb{E}_{x_t \sim P}\left[||v_t(x_t) - v_t^1(x_t)||^2\right]}^{\mathcal{L}_{\text{OS}}} + \gamma \underbrace{\mathbb{E}_{x_t \sim P^{\pi} m}\left[||v_t(x_t) - v_t^2(x_t)||^2\right]}_{\mathcal{L}_{\text{GHM}}} \Big\}.$$

How do we get v_t^1 and v_t^2? *Conditioning trick* **separately!**

Case 1. $\mathcal{L}_{\text{OS}}, S' \sim P(\cdot \mid S, A)$

1. Our dataset (S, A, S') has plenty of samples $S' \sim P(\cdot \mid S, A)$
2. $\vec{u}_t(\cdot \mid Z), \vec{p}_t(\cdot \mid Z)$ can be chosen as done before in **traditional** flow matching

$$\vec{v}_t(x \mid s, a) = \int \vec{u}_t(x \mid x_1) \frac{\vec{p}_t(x \mid x_1) P(\mathrm{d}x_1 \mid s, a)}{\vec{p}_t(x \mid s, a)},$$

Case 1. $\mathcal{L}_{\text{OS}}$, $S' \sim P(\cdot \mid S, A)$

1. Our dataset (S, A, S') has plenty of samples $S' \sim P(\cdot \mid S, A)$
2. $\vec{u}_t(\cdot \mid Z)$, $\vec{p}_t(\cdot \mid Z)$ can be chosen as done before in **traditional** flow matching

$$\vec{v}_t(x \mid s, a) = \int \vec{u}_t(x \mid x_1) \frac{\vec{p}_t(x \mid x_1) P(\mathrm{d}x_1 \mid s, a)}{\vec{p}_t(x \mid s, a)},$$

$$\mathcal{L}_{COS}(\theta) = \mathbb{E}_{\rho, t, Z, \vec{X}_t}\left[\left\|\tilde{v}_t(\vec{X}_t \mid S, A; \theta) - \vec{u}_t(\vec{X}_t \mid Z)\right\|^2\right],$$

$$\text{where } Z = X_1 \sim P(\cdot \mid S, A),\ \vec{X}_t \sim \vec{p}_t(\cdot \mid Z)$$

Case 2. $\mathcal{L}_{\text{GHM}}$, $S' \sim (P^\pi m)(\cdot \mid S, A)$

Assume. probability path m_t, such that $m_0 = m_0$ and $m_1 = m_1$.

Recall. We want a *marginal* distribution, such that

$$(P^\pi m_t)(\cdot \mid S, A) \approx \int_{S'} p_t(\cdot \mid S', A') q(S' \mid S, A)$$

for some p_t and q.

Case 2. $\mathcal{L}_{\text{GHM}}$, $S' \sim (P^\pi m)(\cdot \mid S, A)$

Assume. probability path m_t, such that $m_0 = m_0$ and $m_1 = m_1$.

Recall. We want a *marginal* distribution, such that

$$(P^\pi m_t)(\cdot \mid S, A) \approx \int_{S'} p_t(\cdot \mid S', A')q(S' \mid S, A)$$

for some p_t and q.

Note. Taking $p_t(\cdot \mid S')$ to be $N(S', \sigma)$ and q to be the desired distribution was just one choice. Need following criteria:

1. $p_t(\cdot \mid S', A')$ can be sampled from,
2. The conditional vector field $u_t(\cdot \mid S', A')$ that generates $p_t(\cdot \mid S', A')$ can be easily computed,
3. Marginal $p_1 \approx P^\pi m_1 = P^\pi m$.

Case 2. $\mathcal{L}_{\text{GHM}}, S' \sim (P^\pi m)(\cdot \mid S, A)$

By definition,

$$(P^\pi m)(\cdot \mid S, A) \approx \int_{S'} m(\cdot \mid S', A') P(S' \mid S, A).$$

In particular,

$$(P^\pi m)(\cdot \mid S, A) \approx \int_{S'} m_1(\cdot \mid S', A') P(S' \mid S, A)$$

and m_t satisfies (1),(2),(3)!

Chicken or the egg: How do we compute the vector field that generates m_t?

Case 2. $\mathcal{L}_{\text{GHM}}$, $S' \sim (P^\pi m)(\cdot \mid S, A)$

Lemma. If $v_t^{(n+1)} = \arg\max \mathcal{L}_{\text{OS+GHM}}(v_t^{(n)})$ then $v_t^{(n+1)}$ generates $m_t^{(n+1)}$ and $m_t^{(n+1)} = \mathcal{T}^\pi m_t^{(n)}$.

Case 2. $\mathcal{L}_{\text{GHM}}$, $S' \sim (P^\pi m)(\cdot \mid S, A)$

Lemma. If $v_t^{(n+1)} = \arg\max \mathcal{L}_{\text{OS+GHM}}(v_t^{(n)})$ then $v_t^{(n+1)}$ generates $m_t^{(n+1)}$ and $m_t^{(n+1)} = \mathcal{T}^\pi m_t^{(n)}$.

Implication. Use $v_t^{(n+1)}(\cdot|S', A')$ in **CFM** loss $\implies$ will converge to vector field that generates $P^\pi m_t$.

Case 2. $\mathcal{L}_{\mathsf{GHM}}$, $S' \sim (P^\pi m)(\cdot \mid S, A)$

Lemma. If $v_t^{(n+1)} = \arg\max \mathcal{L}_{\mathsf{OS+GHM}}(v_t^{(n)})$ then $v_t^{(n+1)}$ generates $m_t^{(n+1)}$ and $m_t^{(n+1)} = \mathcal{T}^\pi m_t^{(n)}$.

Implication. Use $v_t^{(n+1)}(\cdot|S', A')$ in **CFM** loss $\implies$ will converge to vector field that generates $P^\pi m_t$.

$$\mathcal{L}_{CGHM}(\theta) = \mathbb{E}_{\rho,t,\widehat{X}_t}\left[\left\|\tilde{v}_t(\widehat{X}_t \mid S, A; \theta) - \tilde{v}_t^{(n)}(\widehat{X}_t \mid S', \pi(S')\right\|^2\right],$$

where $X_0 \sim p_0$, $S' \sim P(\cdot \mid S, A)$, $\widehat{X}_t = \widetilde{f}_t^{(n)}(X_0 \mid S', \pi(S'))$, Now we just combine the **objectives**!!

$$\mathcal{L}_{COS}(\theta) = \mathbb{E}_{\rho,t,Z,\vec{X}_t}\left[\left\|\tilde{v}_t(\vec{X}_t \mid S, A; \theta) - \vec{u}_t(\vec{X}_t \mid Z)\right\|^2\right],$$

$$\text{where } Z = X_1 \sim P(\cdot \mid S, A),\ \vec{X}_t \sim \vec{p}_t(\cdot \mid Z)$$

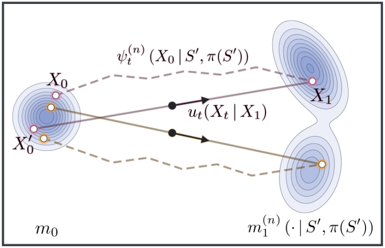
$$+$$

$$\mathcal{L}_{CGHM}(\theta) = \mathbb{E}_{\rho,t,\widehat{X}_t}\left[\left\|\tilde{v}_t(\widehat{X}_t \mid S, A; \theta) - \tilde{v}_t^{(n)}(\widehat{X}_t \mid S', \pi(S'))\right\|^2\right],$$

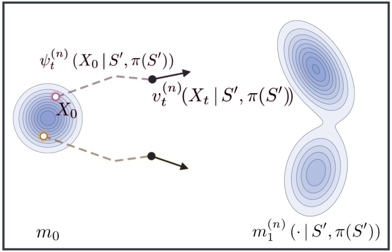
where $X_0 \sim p_0$, $S' \sim P(\cdot \mid S, A)$, $\widehat{X}_t = \widetilde{f}_t^{(n)}(X_0 \mid S', \pi(S'))$,

$$\mathcal{L}_{\mathrm{TD^2-CFM}}(\theta) = (1-\gamma)\mathcal{L}_{COS}(\theta) + \gamma\mathcal{L}_{CGHM}(\theta) \implies \text{Lower } \mathbb{E}\nabla\mathcal{L}^2.$$

TD-CFM

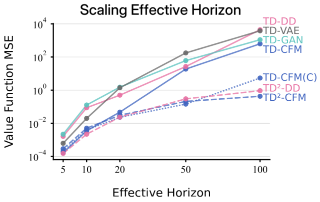


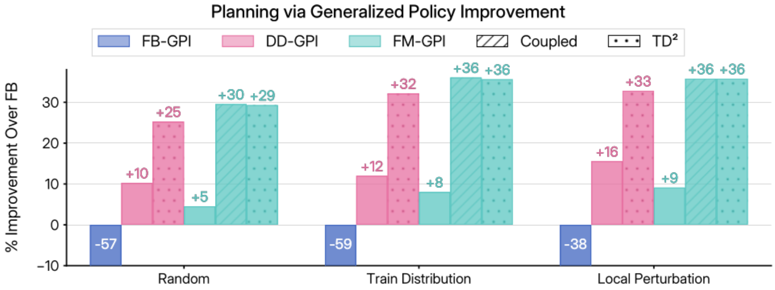
TD²-CFM



GPI:

- Train π_w with Forward backward.
- Do GPI as $w_t \in \underset{w \sim D(W)}{\arg\max} \underbrace{(1-\gamma)^{-1}\mathbb{E}_{X \sim m^{\pi_w}(\cdot|s_t,\pi_w(s_t))}[r(X)]}_{Q^{\pi_w}(s_t,\pi_w(s_t))}$.
- Averaged over 128 samples.





References

Tian Qi Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural Ordinary Differential Equations.

Jesse Farebrother, Matteo Pirotta, Andrea Tirinzoni, Rémi Munos, Alessandro Lazaric, and Ahmed Touati. Temporal Difference Flows, March 2025.

Ruiqi Gao, Emiel Hoogeboom, Jonathan Heek, Valentin De Bortoli, Kevin P. Murphy, and Tim Salimans. Diffusion Meets Flow Matching: Two Sides of the Same Coin, December 2024.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow Matching for Generative Modeling, February 2023.

Yaron Lipman, Marton Havasi, Peter Holderrieth, Neta Shaul, Matt Le, Brian Karrer, Ricky T. Q. Chen, David Lopez-Paz, Heli Ben-Hamu, and Itai Gat. Flow Matching Guide and Code, December 2024.

Seohong Park. Q-learning is not yet scalable, June 2025.

Danilo Jimenez Rezende and Shakir Mohamed. Variational Inference with Normalizing Flows.

Chongyi Zheng, Seohong Park, Sergey Levine, and Benjamin Eysenbach. Intention-Conditioned Flow Occupancy Models, June 2025.