Advanced Flow Matching



DMQA Open Seminar (26. 11. 27)

Data Mining & Quality Analytics Lab.

송하영

발표자 소개



❖ 송하영 (Hayoung Song)

- 고려대학교 산업경영공학과 석사과정 (2026.03 ~ Present)
- Data Mining & Quality Analytics Labs. (김성범 교수님)

❖ Research Interest

- Generative Models
- Diffusion Models

❖ Contact

- thdgk1245@korea.ac.kr

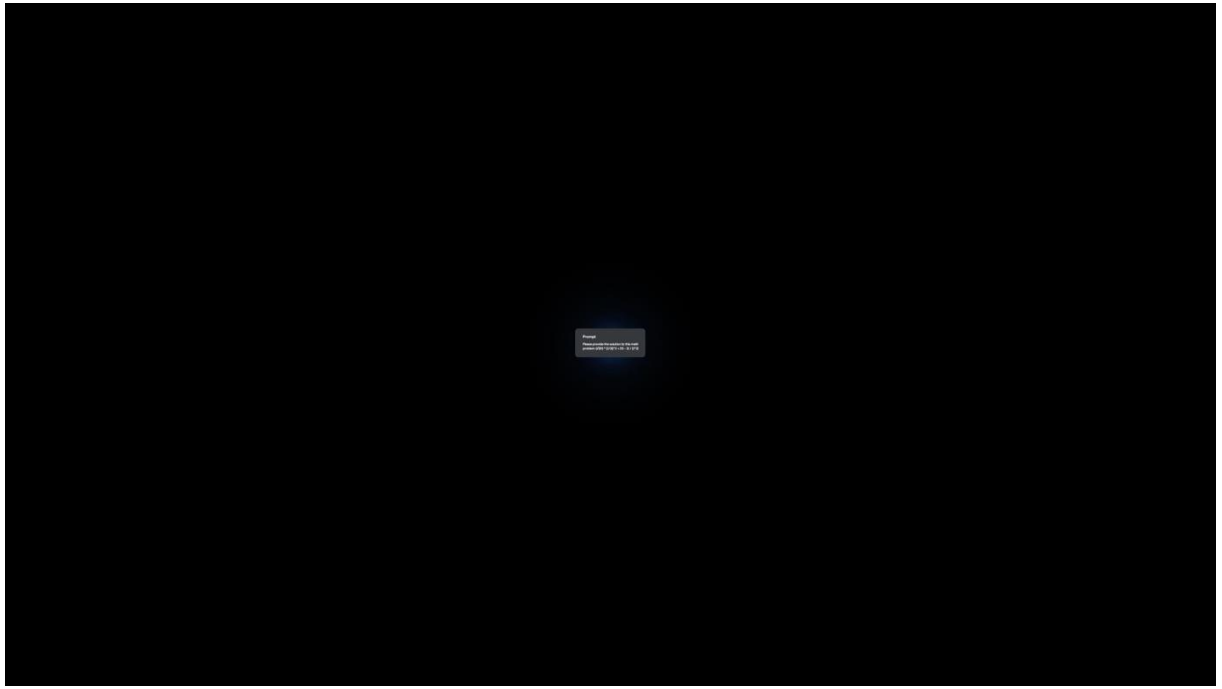
Introduction

Advanced Flow Matching

❖ Generative Models



Flux3 (Black Forest Labs)



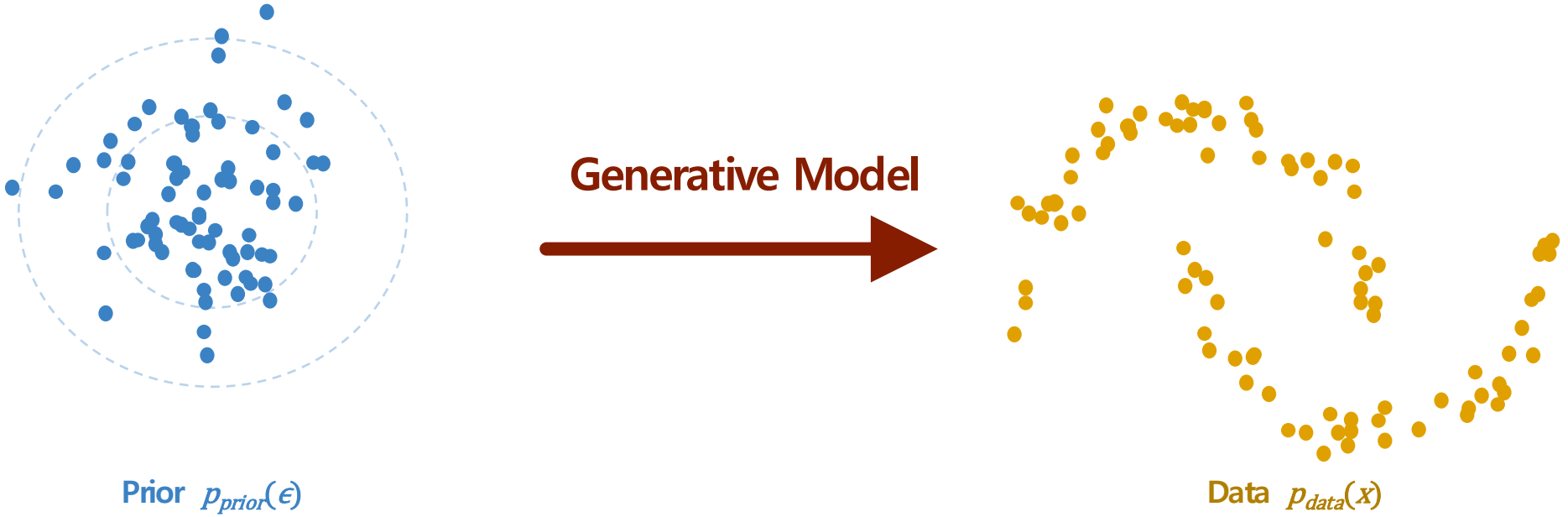
Gemini Diffusion (Google)

Introduction

Advanced Flow Matching

❖ Principle of Generative Models

- **Prior** $p_{prior}(\epsilon)$: Gaussian처럼 샘플링이 쉬운 분포
- **Data** $p_{data}(x)$: 이미지처럼 형태를 모르는 분포
- **생성 모델**: 샘플링이 **쉬운 분포에서 뽑은 데이터를 실제 데이터 분포로 옮기는 변환 과정**을 학습

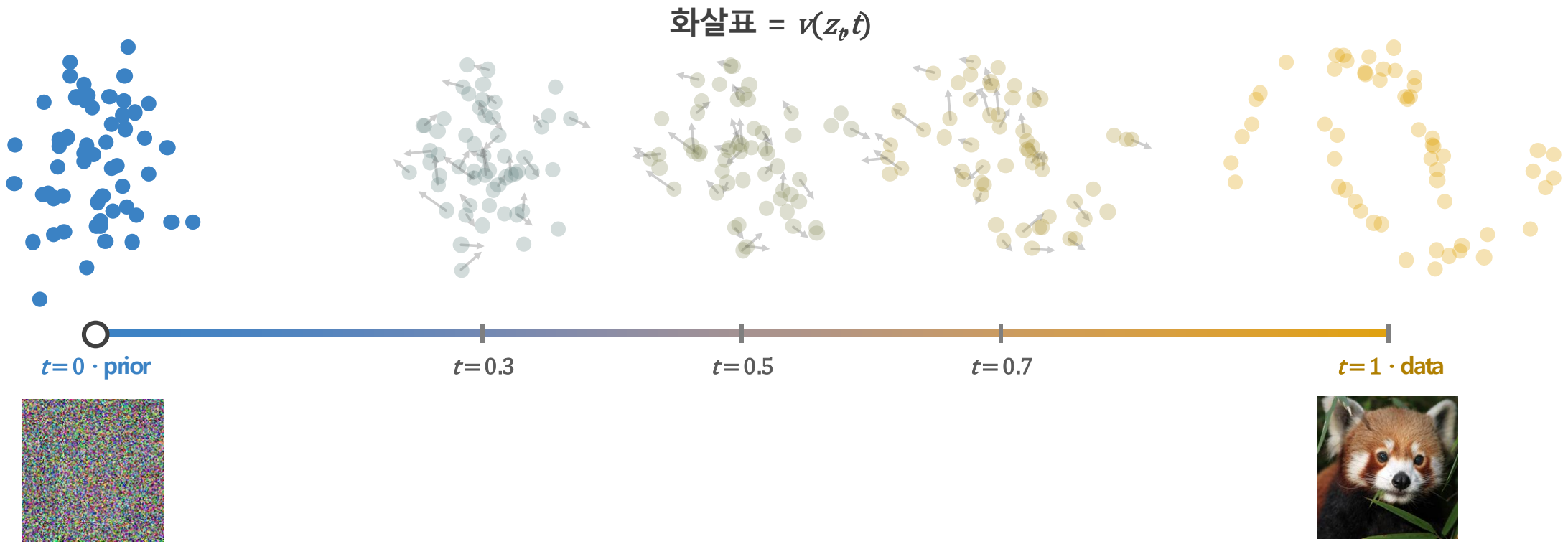


Introduction

Advanced Flow Matching

❖ 생성형 모델은 어떻게 동작할까?

- 시간 t가 0 → 1로 흐르며 prior가 data로 변환된다
- 각 위치·시간의 **velocity** $v(z_t, t)$ — 샘플 전체가 학습된 velocity를 따라 흘러 분포 전체가 이동

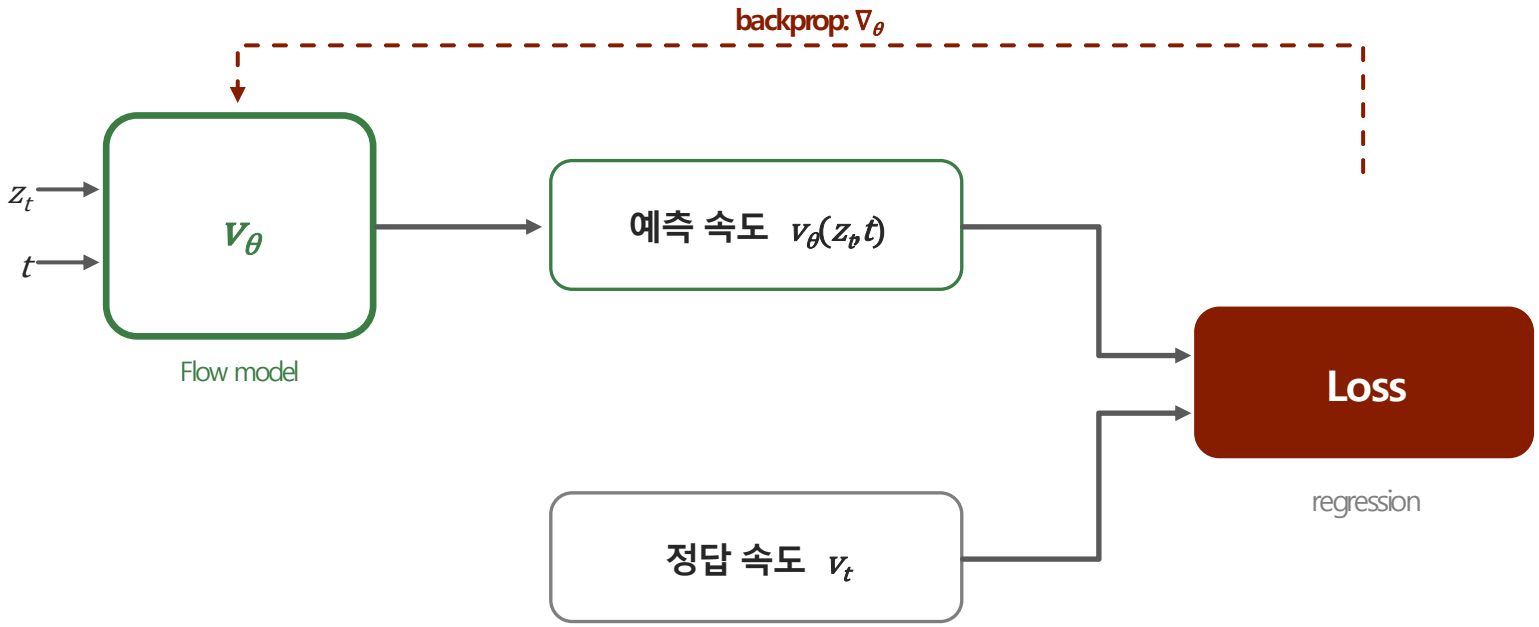


Introduction

Advanced Flow Matching

❖ Velocity field는 어떻게 학습될까?

- 입력: 현재 위치 z_t와 시간 $t \rightarrow$ 출력: 예측 속도 $v_\theta(z_t, t)$
- 정답 속도 v_t와의 차이를 줄이는 **regression**으로 학습



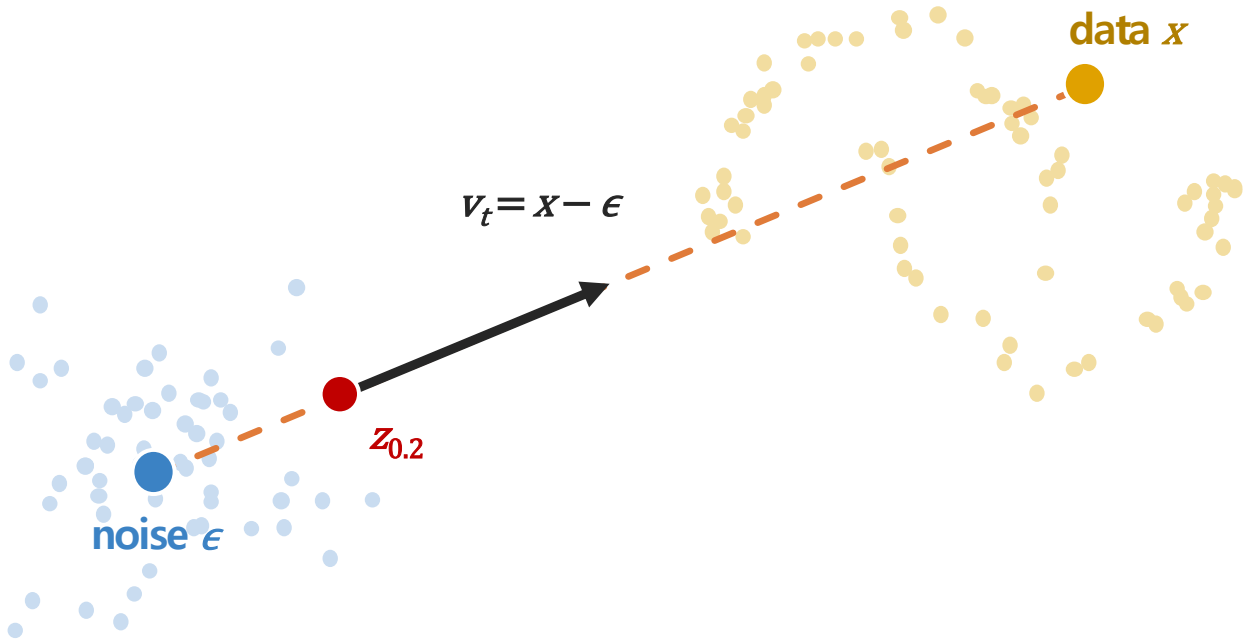
$$\min_\theta \mathbb{E}\|v_\theta(z_t, t) - v_t\|^2$$

Introduction

Advanced Flow Matching

❖ **정답 velocity를 모른다 → 샘플 한 쌍으로 조건부 정답을 만든다**

- 전체 분포의 marginal 속도는 data 분포를 알아야 해서 **intractable**
- 대신 한 쌍 (ϵ, x)을 잇는 직선 $z_t = (1-t)\epsilon + tx$ 사용 → 정답 속도 $v_t = x - \epsilon$ **바로 계산**

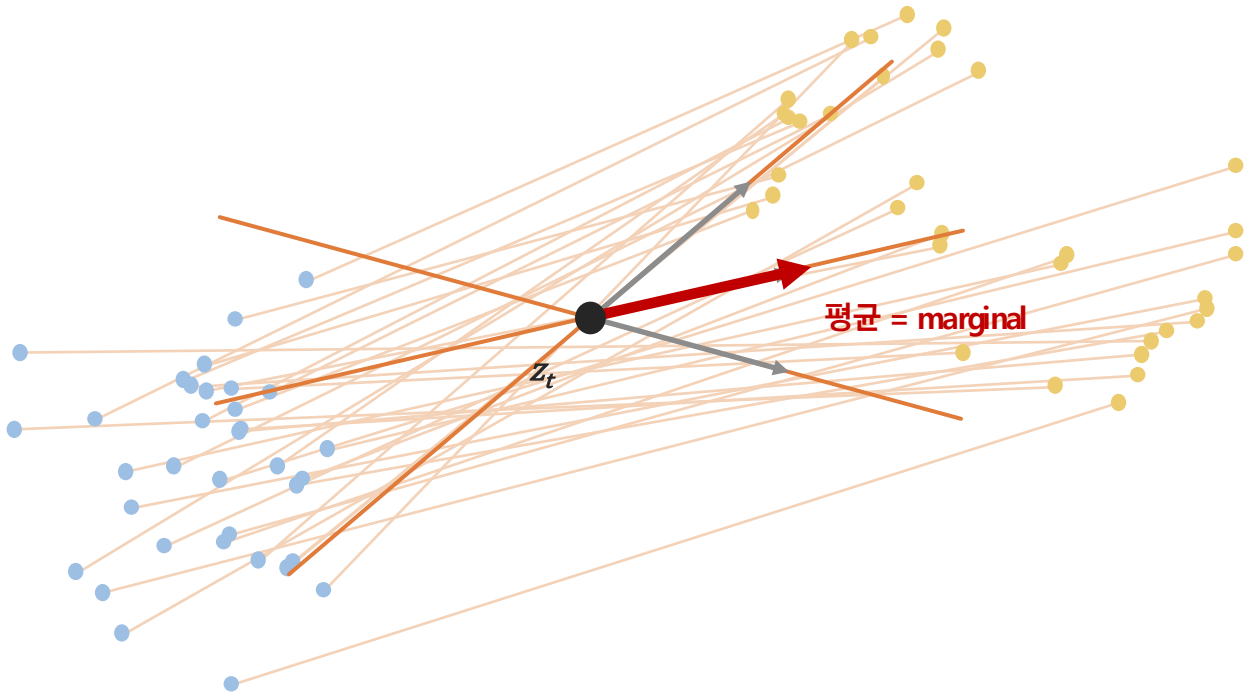


Introduction

Advanced Flow Matching

❖ 수많은 조건부 직선의 평균 = Marginal flow (Flow Matching)

- 같은 z_t를 여러 직선이 지나감 → 쌍마다 다른 조건부 속도
- network는 그 평균 $\mathbb{E}[v_t|z_t]$ = **marginal velocity**를 학습 — gradient가 원래 FM loss와 동일 (Lipman et al., 2023)
- 조건부 정답을 **<u>무수히 많이 구하면 marginal velocity</u>**가 학습됨

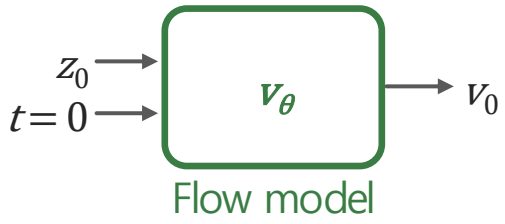


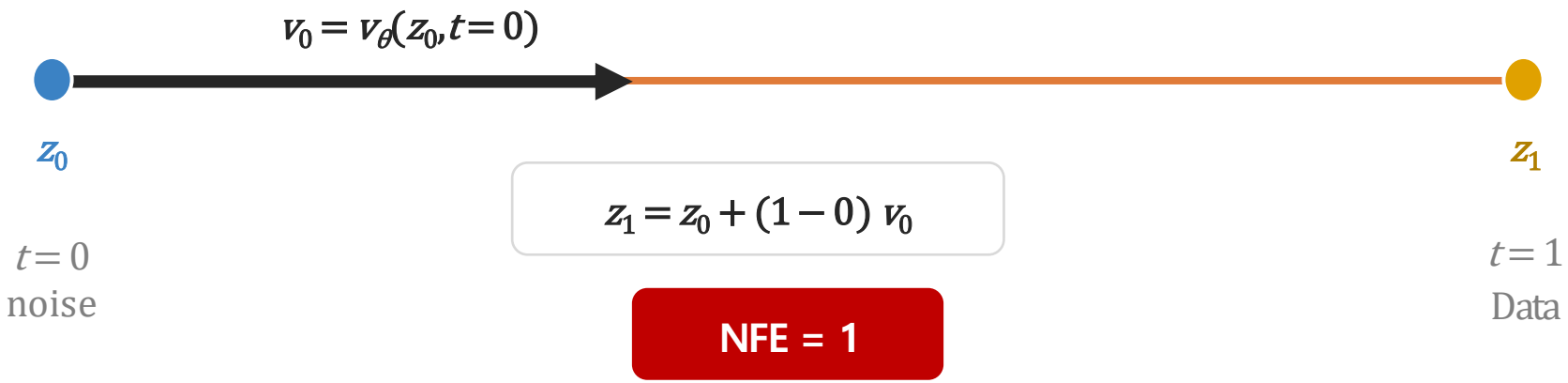
Introduction

Advanced Flow Matching

❖ 경로가 직선이라면 velocity 한 번으로 바로 도착하지 않을까?

- 직선 경로에서는 출발점의 속도 v_0가 끝까지 변하지 않음
- $z_1 = z_0 + (1-0)\, v_0$ → network **1회 호출 (Number of Function Evaluation (NFE) = 1)**



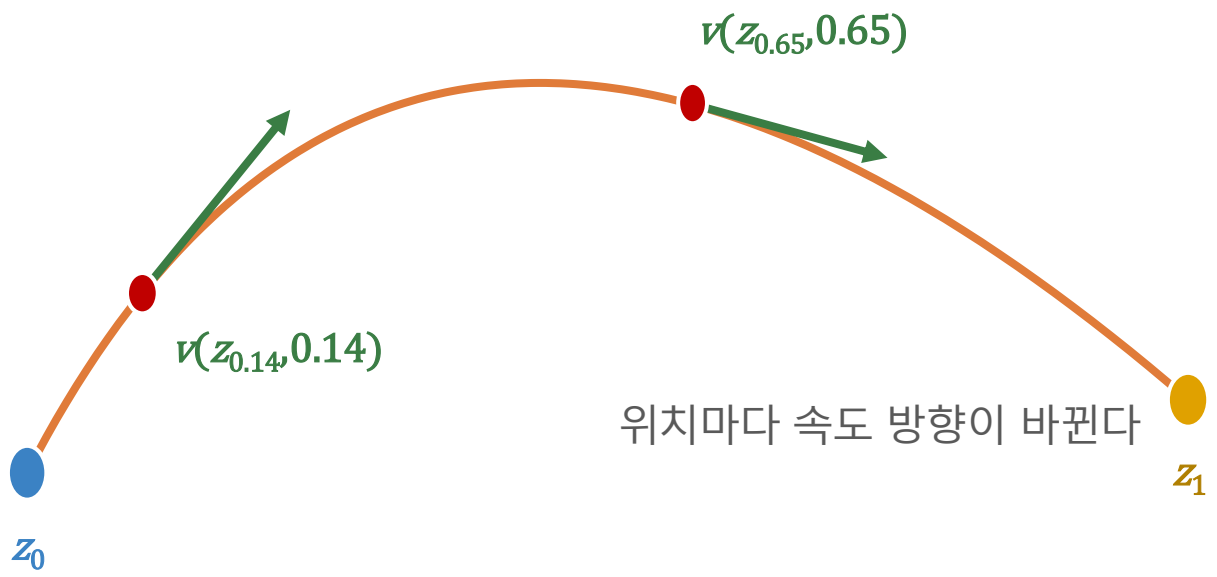


Introduction

Advanced Flow Matching

❖ **하지만 실제 marginal 경로는 휘어 있다 → ODE를 적분해야 한다**

- 직선 conditional path들의 평균인 marginal 경로는 일반적으로 **곡선** (정답 field 자체의 성질)
- 위치마다 속도 방향이 달라 ODE $dz_t/dt = v_\theta(z_t, t)$를 **적분**해야 한다: $z_1 = z_0 + \int_0^1 v_\theta \, d\tau$

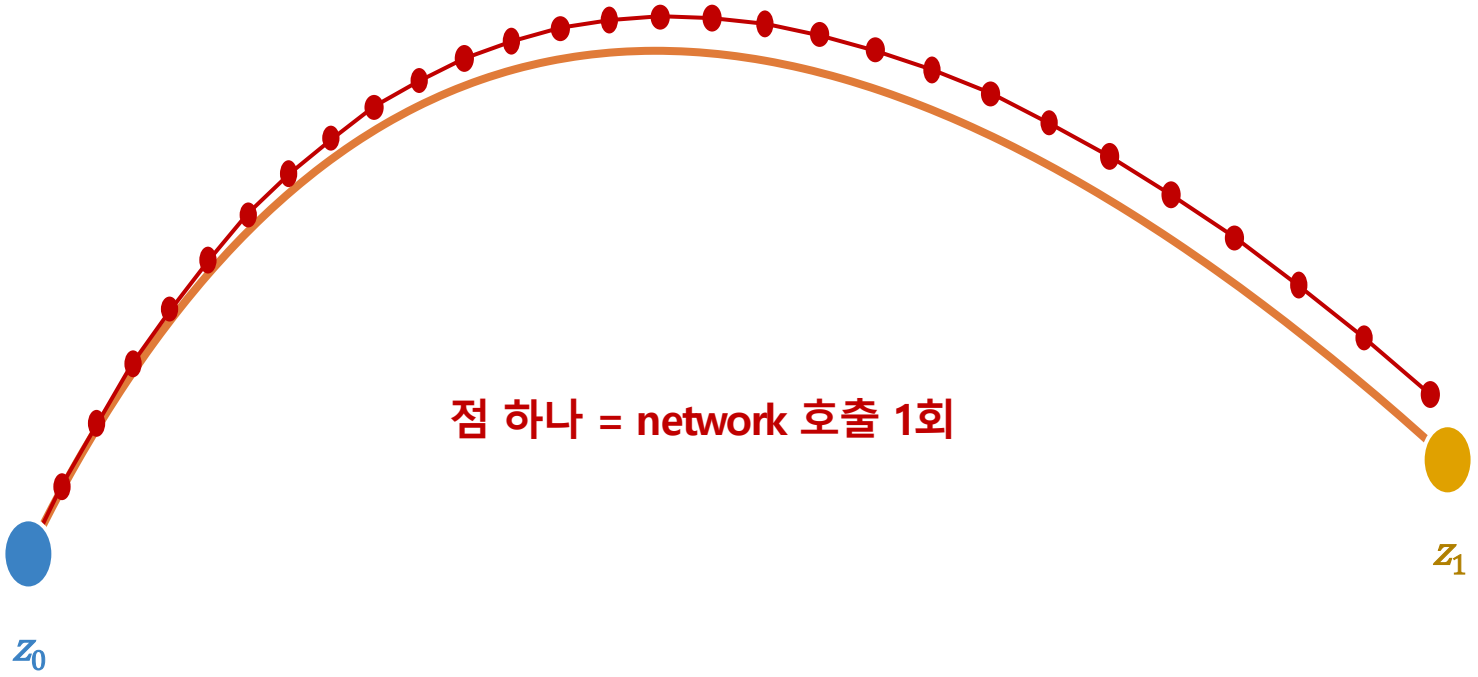


Introduction

Advanced Flow Matching

❖ Euler method: 잘게 나눠 여러 번 이동 → 정확하지만 느리다

- Euler update: $z_{ti+1} = z_{ti} + \Delta\, t\, v_\theta(z_{ti}, t_i)$
- 점 하나 = network 1회 → **step 수 = NFE** (예: SiT-XL/2는 250 step × 2 (CFG) = 500 NFE)
- 정확한 생성을 위해서는 **<u>많은 횟수의 velocity 호출이 필요</u>**

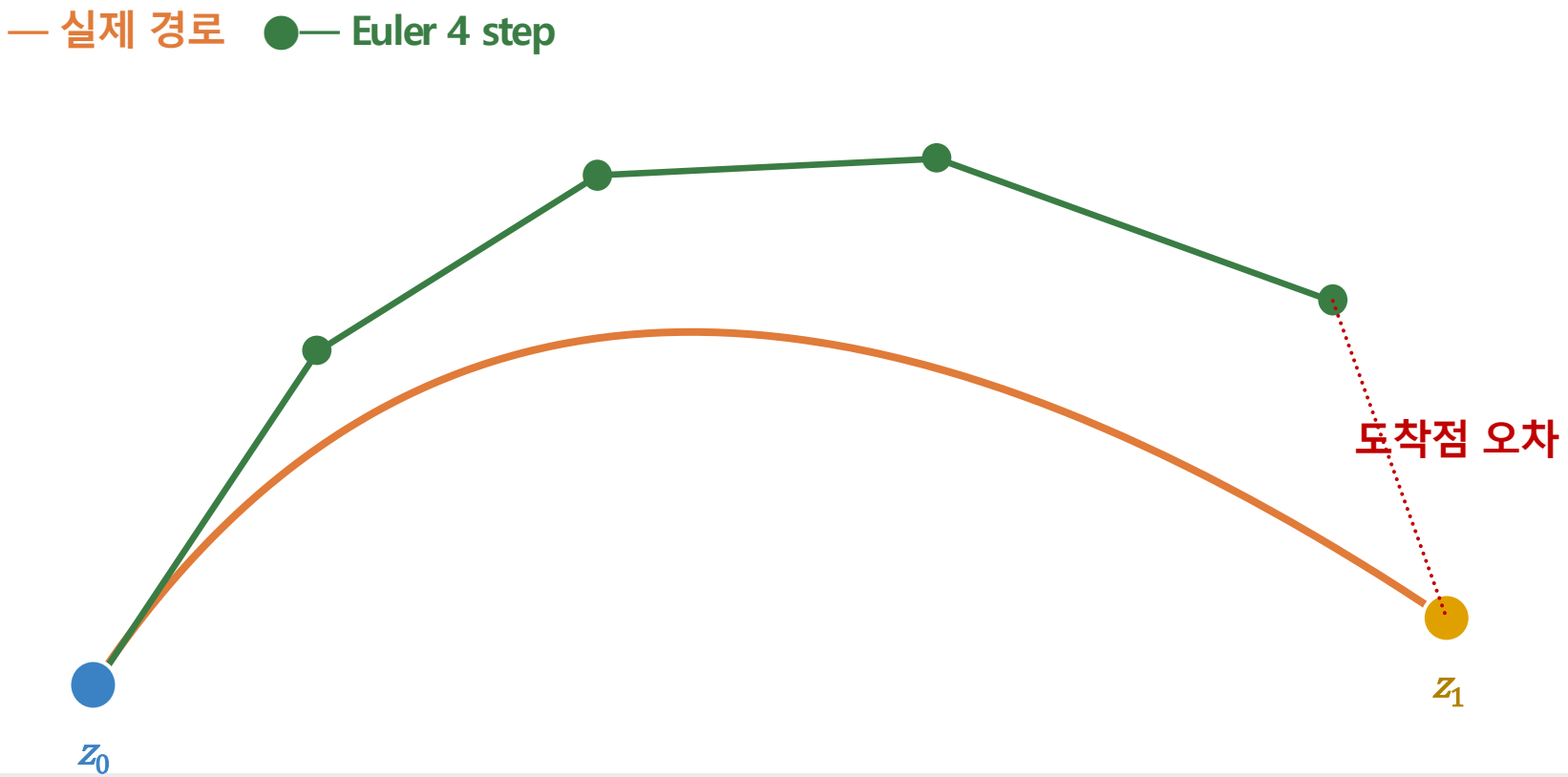


Introduction

Advanced Flow Matching

❖ **Step을 크게 해서 더 빠르게 생성하면 어떨까?**

- 큰 step: 출발 시점의 접선 방향으로 멀리 이동
- 곡률 때문에 step마다 경로를 벗어나 **오차 누적** → 데이터가 아닌 곳에 도착
- **적은 step → 빠르지만 품질 저하**

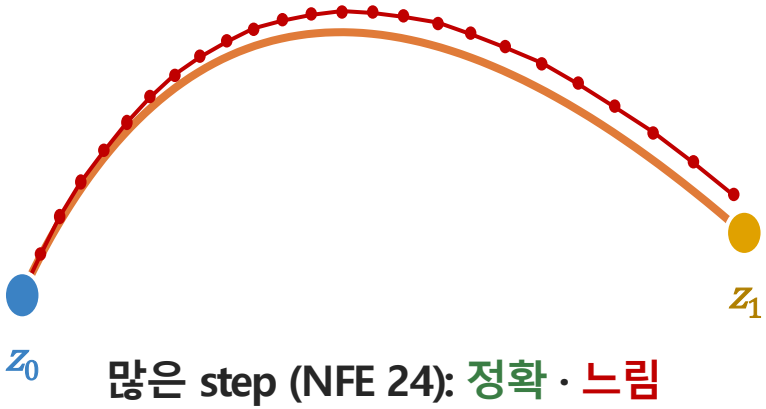


Introduction

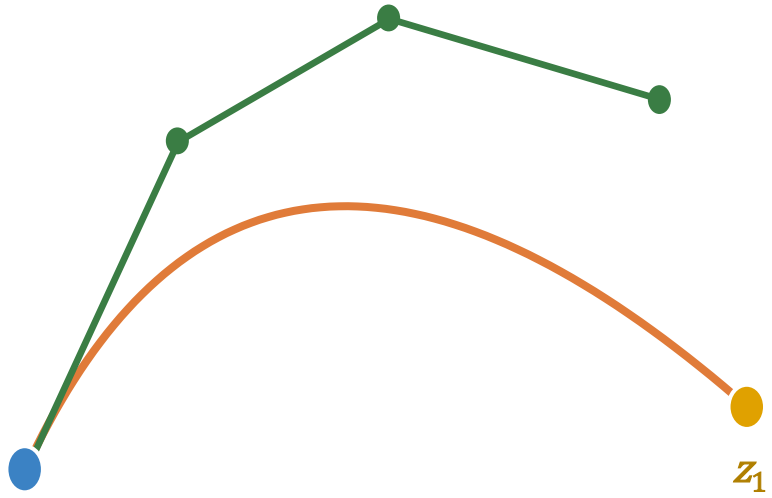
Advanced Flow Matching

❖ Flow Matching의 한계는 무엇일까?

- 많은 step: 정확하지만 느림 · 적은 step: 빠르지만 부정확
- 근본 원인: network는 **지금 이 순간의 속도**만 알려줌 → 멀리 가려면 여러 번 물어봐야 한다
- 고품질 데이터를 위해서는 많은 횟수의 network 호출이 불가피함



많은 step (NFE 24): 정확 · 느림



적은 step (NFE 3): 빠름 · 부정확

MeanFlow

Advanced Flow Matching

❖ Mean Flows for One-step Generative Modeling

- Neurips 2025 Oral, CMU & MIT
- 2026. 11. 27 기준 761회 인용

Mean Flows for One-step Generative Modeling

Zhengyang Geng[1]* **Mingyang Deng**[2] **Xingjian Bai**[2] **J. Zico Kolter**[1] **Kaiming He**[2]

[1]CMU [2]MIT

Abstract

We propose a principled and effective framework for one-step generative modeling. We introduce the notion of average velocity to characterize flow fields, in contrast to instantaneous velocity modeled by Flow Matching methods. A well-defined identity between average and instantaneous velocities is derived and used to guide neural network training. Our method, termed the *MeanFlow* model, is self-contained and requires no pre-training, distillation, or curriculum learning. MeanFlow demonstrates strong empirical performance: it achieves an FID of 3.43 with a single function evaluation (1-NFE) on ImageNet 256×256 trained from scratch, significantly outperforming previous state-of-the-art one-step diffusion/flow models. Our study substantially narrows the gap between one-step diffusion/flow models and their multi-step predecessors, and we hope it will motivate future research to revisit the foundations of these powerful models. Our code is available at https://github.com/gsunshine/meanflow.

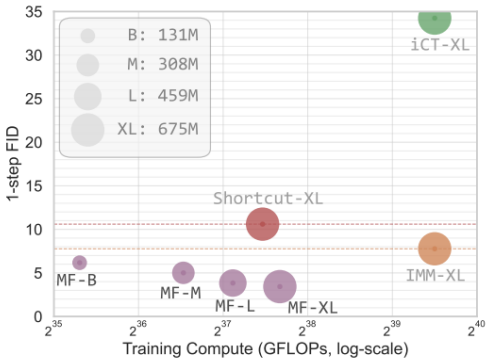




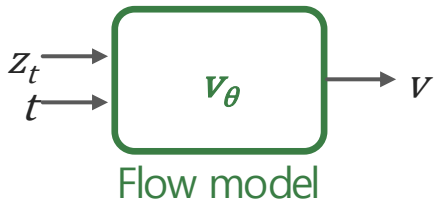
Figure 1: **One-step generation on ImageNet 256×256 from scratch**. Our *MeanFlow* (**MF**) model achieves significantly better generation quality than previous state-of-the-art one-step diffusion/flow methods. Here, iCT [43], Shortcut [12], and our MF are all **1-NFE** generation, while IMM's 1-step result [52] involves 2-NFE guidance. Detailed numbers are in Tab. 2. Images shown are generated by our 1-NFE model.

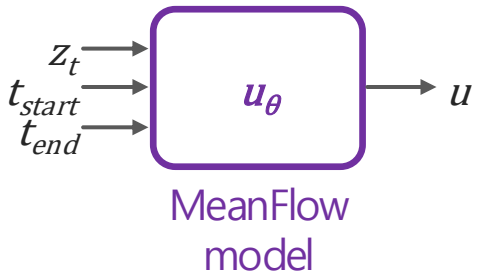
MeanFlow

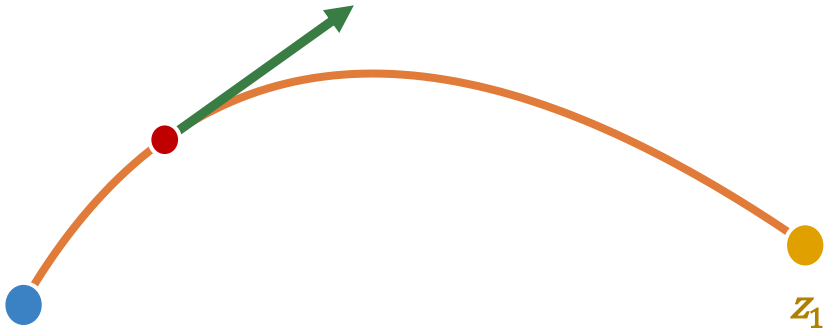
Advanced Flow Matching

❖ 순간 속도 대신 평균 속도를 예측하면?

- Flow model $v_\theta(z_t, t)$: 지금 이 순간의 방향 (접선)
- MeanFlow model $u_\theta(z_t, t_{start}, t_{end})$: 구간 전체의 **평균 방향** → 경로가 휘어도 한 번에 도착







Instantaneous velocity: '지금 어디로?'



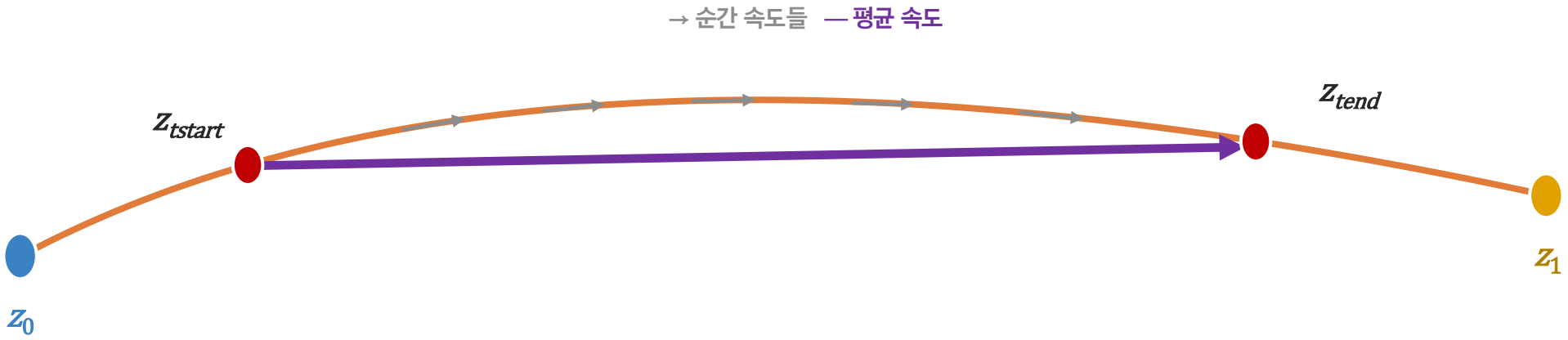
Average velocity: '결국 어디로?'

MeanFlow

Advanced Flow Matching

❖ 어떻게 정답을 정의할 수 있을까?

- 분자: 순간 속도의 적분 = **실제 변위** · 분모: **구간 길이**
- 평균 속도를 알면 적분 없이 **한 번에 이동** — 단, 정의 자체에 적분이 존재



$$u(z_t, t_{start}, t_{end}) = \frac{1}{t_{end} - t_{start}} \int_{t_{start}}^{t_{end}} v(z_\tau, \tau)\, d\tau$$

MeanFlow

Advanced Flow Matching

❖ **Average velocity를 알면: noise에서 data까지 한 번에 갈 수 있지 않을까?**

- Flow model은 **순간 속도** $v_\theta(z_t, t)$, MeanFlow model은 구간의 **평균 속도** $u_\theta(z_t, t_{end}, t_{start})$를 예측
- $t_{start} = 0$, $t_{end} = 1$을 넣으면 한 번의 계산으로 $z_0 \rightarrow z_1$ — **iterative sampling 불필요**



$$z_1 = z_0 + u_\theta(z_0, t_{end} = 1, t_{start} = 0)$$

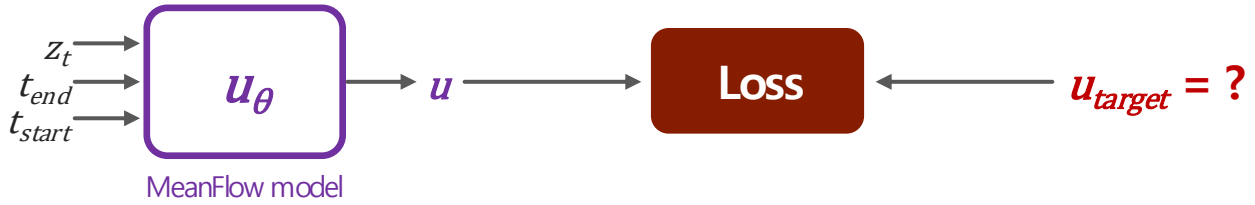
가장 이상적인 one-step generation — No iterative sampling

MeanFlow

Advanced Flow Matching

❖ 그렇다면 MeanFlow model은 무엇을 정답으로 학습해야 할까?

- Flow Matching처럼 network 출력 u를 정답 u_{target}에 회귀하면 된다
- 문제: 정의에 **적분**이 있어 u_{target}을 직접 계산하려면 학습 중 매번 적분이 필요



$$u(z_t, t_{end}, t_{start}) = \frac{1}{t_{end} - t_{start}} \int_{t_{start}}^{t_{end}} v(z_\tau, \tau)\, d\tau$$

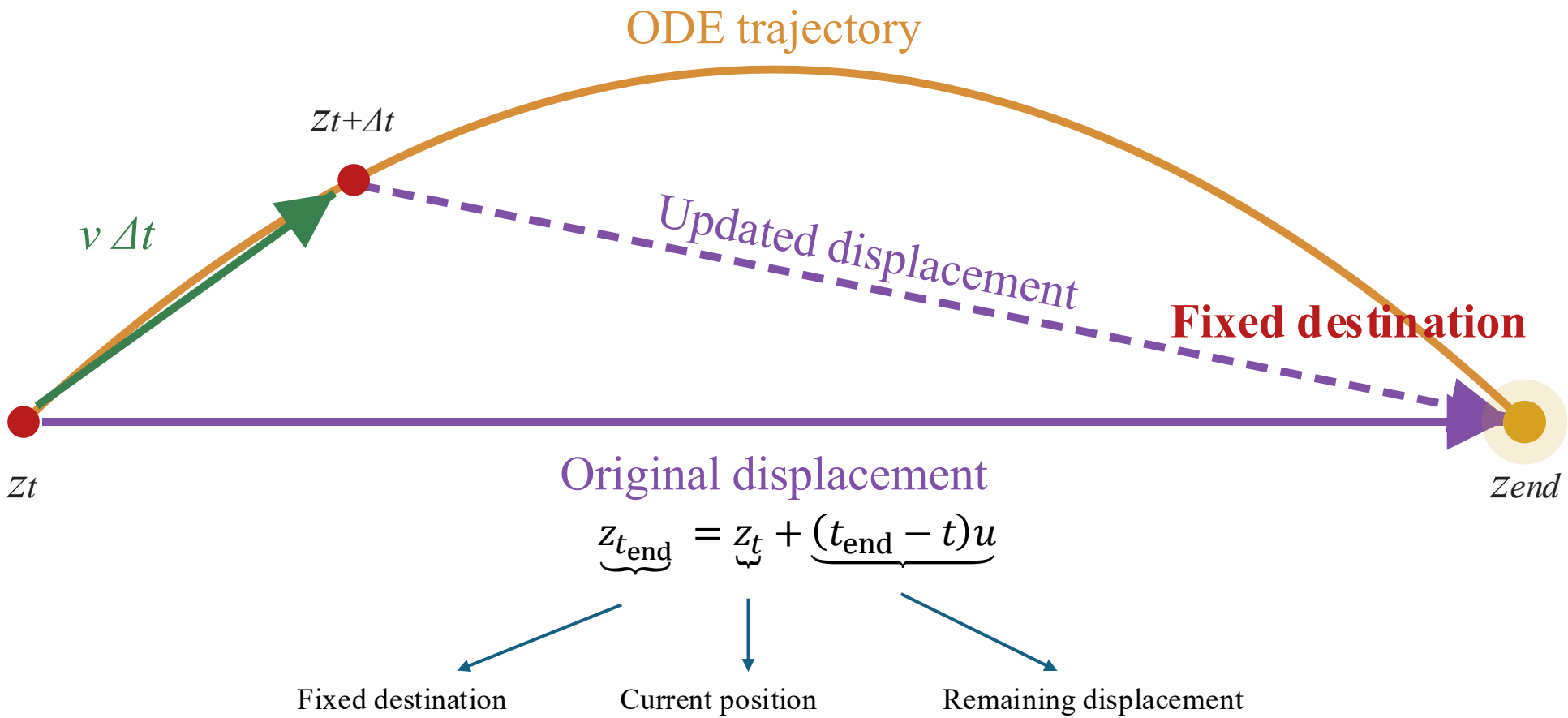
적분 없이 u_{target}을 만들 수 있을까? → loss function 설계

MeanFlow

Advanced Flow Matching

❖ 동일한 trajectory에서는 출발점이 변해도 목적지는 변하지 않는다

- 출발 시점이 변하면 현재 위치와 남은 평균 속도도 함께 변함
- 그러나 동일한 ODE trajectory 위에서는 목표 시점의 도착 위치가 항상 동일하다

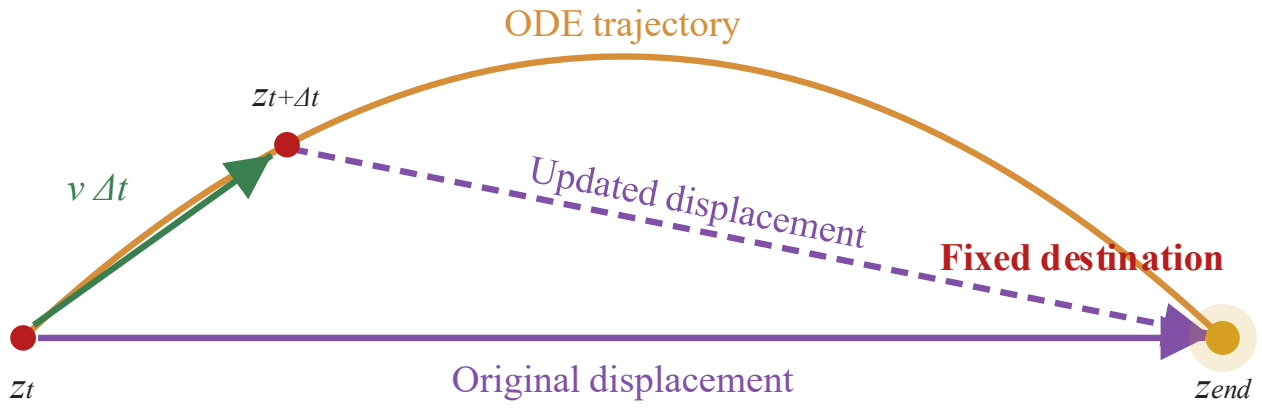


MeanFlow

Advanced Flow Matching

❖ **목적지의 불변성으로부터 무엇을 유도할 수 있을까?**

- 동일한 trajectory에서 출발 시점이 변하더라도 목표 시점의 도착 위치는 동일
- 목적지를 나타내는 식을 출발 시점에 대해 미분
- **MeanFlow Identity : 적분 대신 평균 속도의 변화율을 이용**



$$z_{t_{end}} = z_t + (t_{end} - t)u$$ → Differentiate along the trajectory & rearrangement $$u(z_t, t_{\text{end}}, t) = v(z_t, t) + (t_{\text{end}} - t)\frac{du}{dt}$$

평균 속도 순간 속도 평균 속도의 변화율

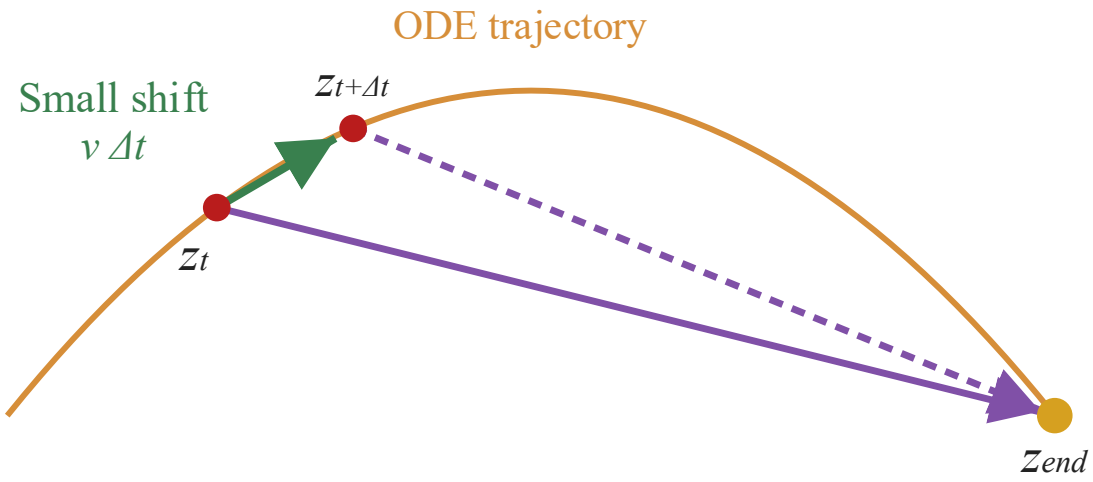
MeanFlow

Advanced Flow Matching

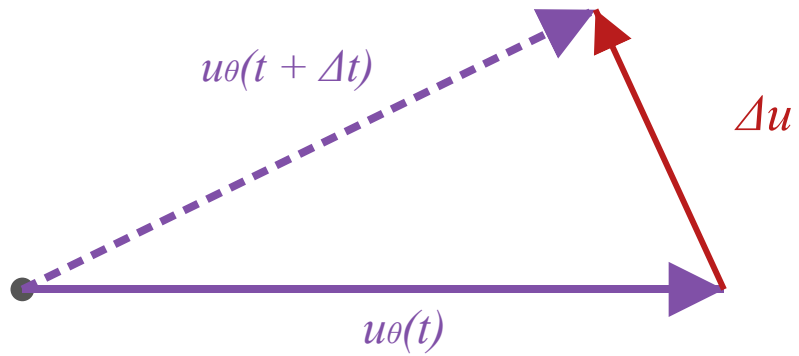
❖ **Trajectory를 따라 이동할 때 평균 속도는 어떻게 변할까?**

- 시간이 흐르면 **현재 위치와 시작 시점이 동시에 변화하지만 목표 시점은 고정**
- Jacobian Matrix를 이용해 이러한 **이동 방향에 대한 모델 출력의 변화율을 계산**

Move along the trajectory



Change in average velocity



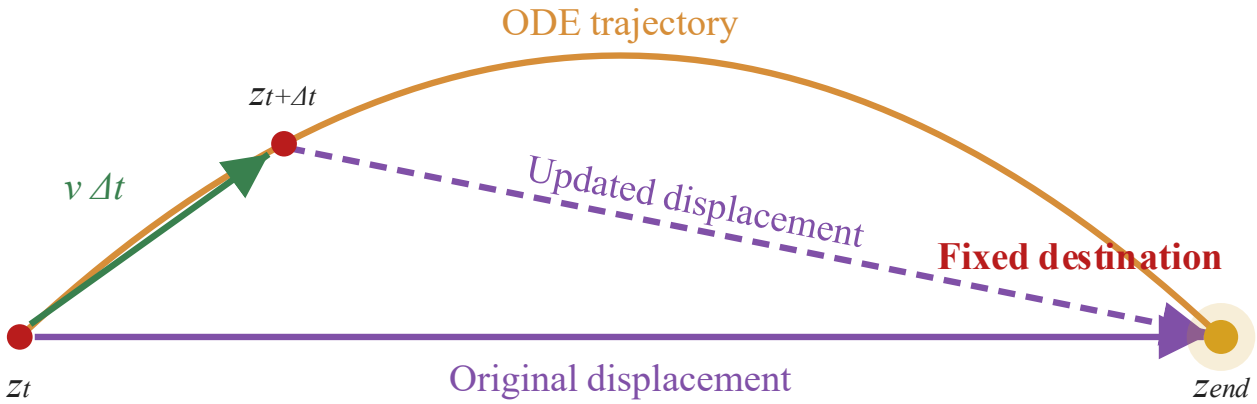
$$\frac{du_\theta}{dt} = \lim_{\Delta t \to 0} \frac{u_\theta(z_{t+\Delta t}, t_{\text{end}}, t + \Delta t) - u_\theta(z_t, t_{\text{end}}, t)}{\Delta t}$$

MeanFlow

Advanced Flow Matching

❖ MeanFlow Identity를 활용하여 평균 속도를 어떻게 학습할까?

- 데이터에서 직접 계산한 순간 속도와 현재 모델에서 계산한 항을 결합하여 학습 target 구성
- **<u>예측한 평균 속도와 구성된 target 간의 차이를 최소화하도록 학습</u>**



$$u_{target} = v_{cond} + (t_{end} - t)du_\theta/dt$$

$$\min_\theta \ \mathbb{E}\left[\left\|u_\theta - \mathrm{sg}(u_{\mathrm{target}})\right\|_2^2\right]$$

Stop-gradient (sg) : 역전파 과정에서 target을 상수 취급하여 jacobian matrix에 대한 고차 미분 계산 방지

MeanFlow

Advanced Flow Matching

❖ 정성적 이미지 결과는 잘 생성될까?

- 기존 one-step 방법은 pretrained model, distillation, discretization curriculum에 의존
- MeanFlow는 평균 속도라는 **target 자체를 바꾸는** 근본적 접근으로 from scratch 1-step 생성

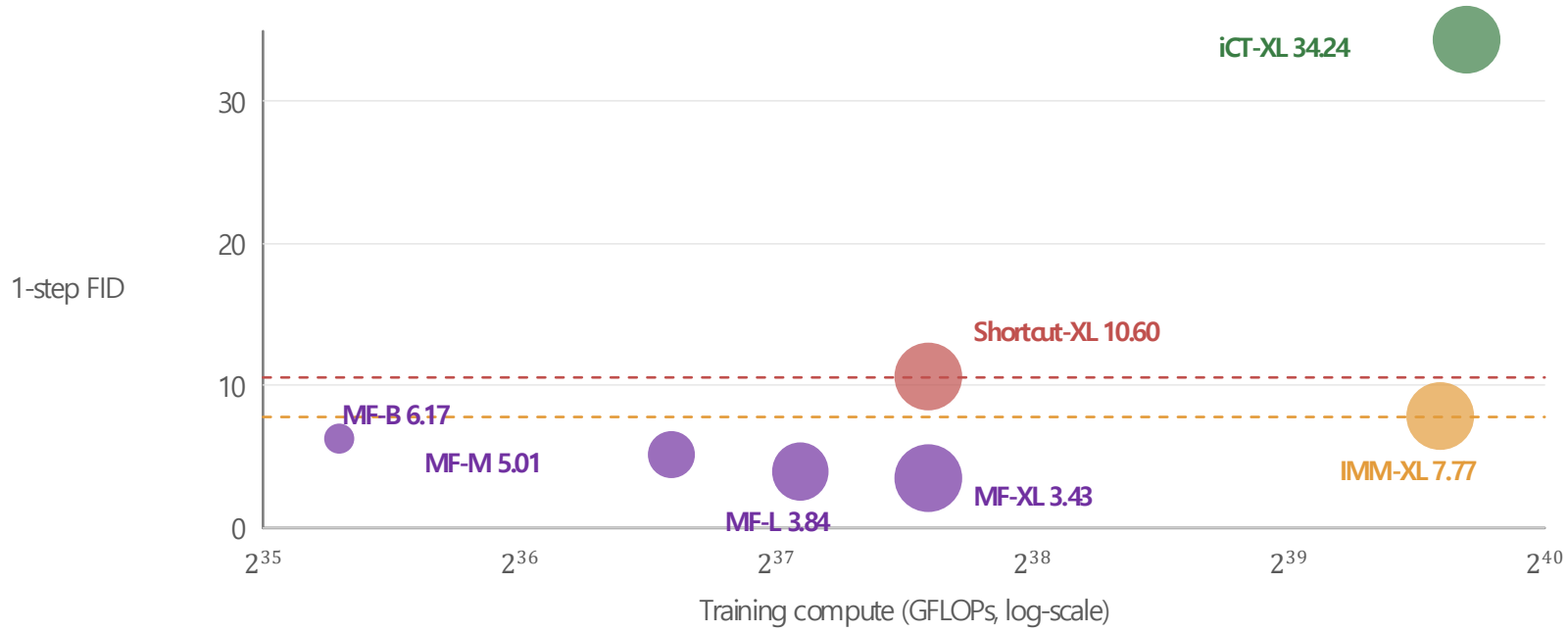


MeanFlow

Advanced Flow Matching

❖ **정량적인 지표에서도 기존 1-step generation 방식들보다 우수할까?**

- ImageNet 256×256, **1-step** 생성 FID (낮을수록 좋음) vs. 학습 compute
- **가장 적은 학습 compute로 가장 낮은 성능**을 보임



적은 학습 compute로 가장 낮은 1-step FID

AlphaFlow

Advanced Flow Matching

❖ AlphaFlow: Understanding and Improving MeanFlow Models

- ICLR 2026, Snap Inc & University of Michigan
- 2026. 11. 27 기준 63회 인용

ALPHAFLOW: UNDERSTANDING AND IMPROVING MEANFLOW MODELS

Huijie Zhang [1,2] * **Aliaksandr Siarohin**[1] **Willi Menapace**[1]
Michael Vasilkovsky[1] **Sergey Tulyakov**[1] **Qing Qu**[2] **Ivan Skorokhodov**[1]
[1]Snap Inc. [2]Department of EECS, University of Michigan

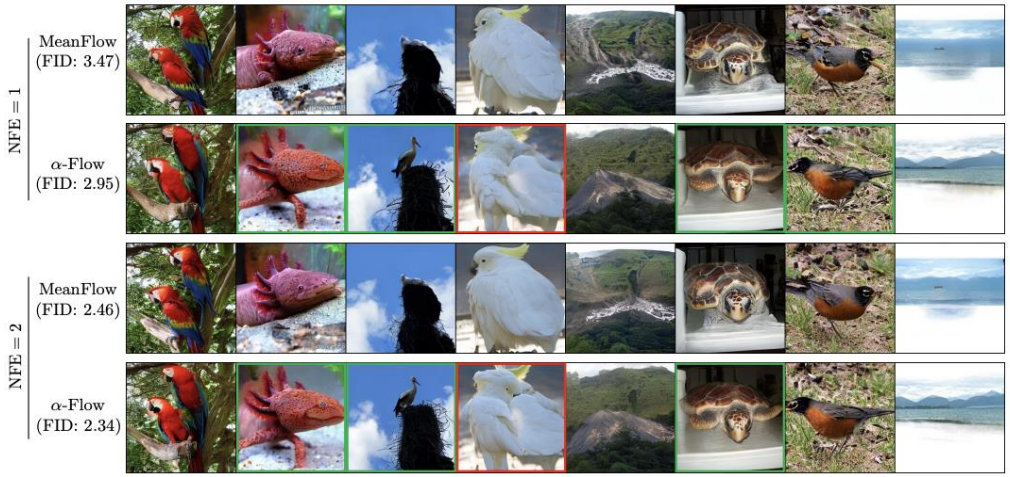


Figure 1: *Uncurated* samples (seeds 1-8) from the DiT-XL/2 model for MeanFlow Geng et al. (2025a) and α-Flow (our proposed method) produced with 1 (upper) and 2 (lower) sampling steps for ImageNet-1K 256^2.

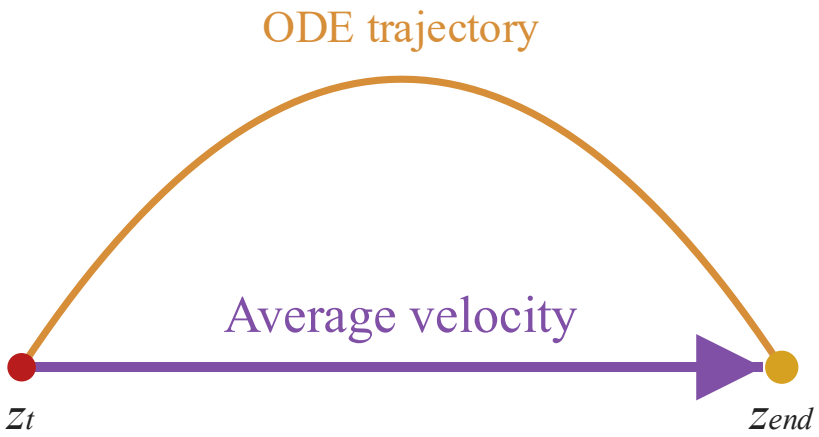
AlphaFlow

Advanced Flow Matching

❖ AlphaFlow는 MeanFlow의 어떤 부분에서 개선을 했을까?

- MeanFlow는 적은 추론 step으로 고품질 생성을 달성하기 위해 평균 속도를 학습
- 그러나 학습의 **75%는 시간 구간의 길이가 0인 순간 속도 학습에 할당**

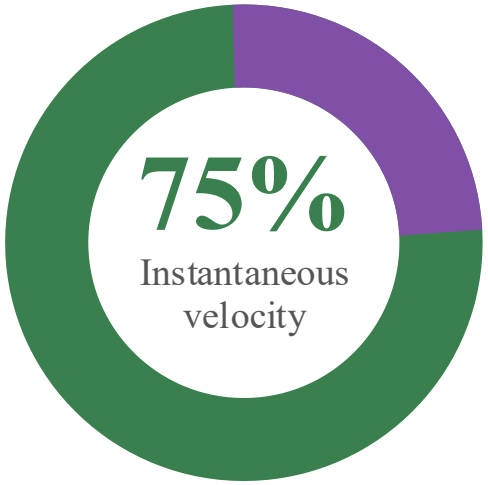
What we want to learn



Large trajectory jumps

Learning finite-time transport

How MeanFlow is trained



75% — Instantaneous velocity

25% — Average velocity

AlphaFlow

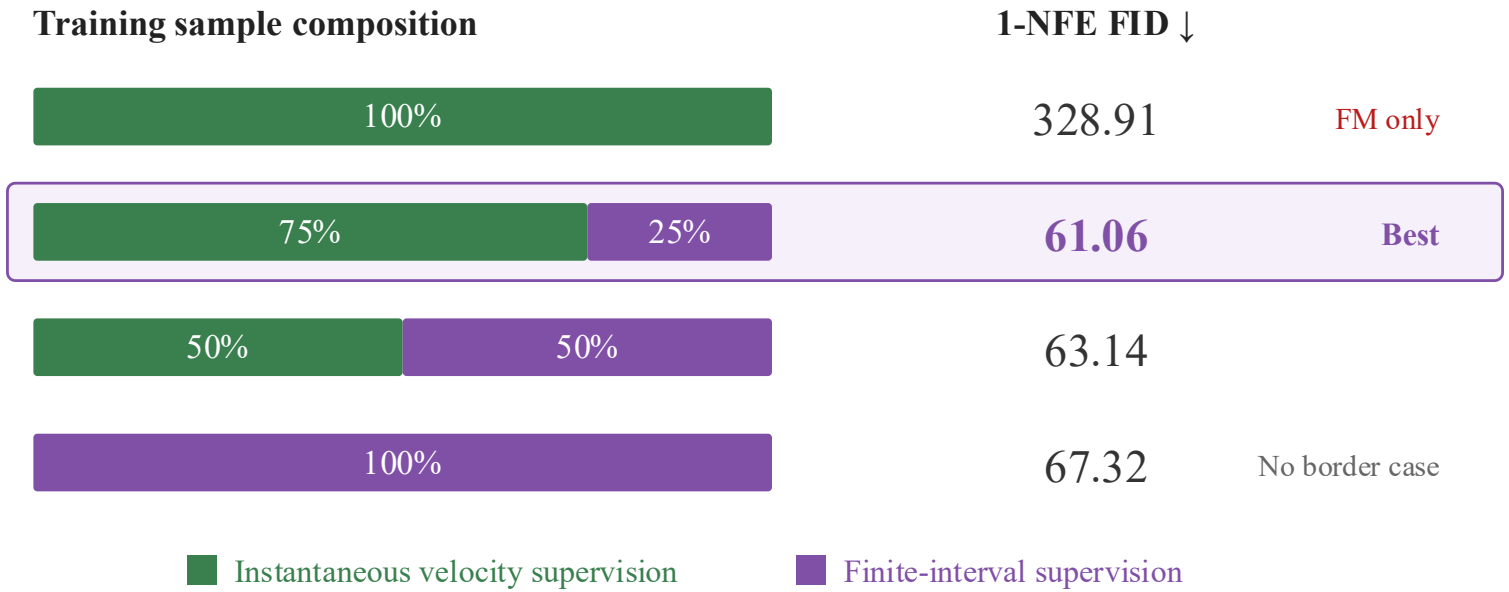
Advanced Flow Matching

❖ 평균 속도 학습에 모든 샘플을 사용하면 더 좋은 결과를 얻을 수 있을까?

- **실험 목적**: MeanFlow에서 순간 속도 학습이 실제로 필요한지 확인하기 위함
- 시간 구간의 길이가 0(순간 속도)인 학습 샘플의 비율을 조절하며 1-NFE 생성 성능을 비교

% of $r \neq t$	FID, 1-NFE
0% (= FM)	328.91
25%	**61.06**
50%	63.14
100%	67.32

(a) **Ratio of sampling** $r \neq t$. The 0% entry reduces to the standard Flow Matching baseline.



AlphaFlow

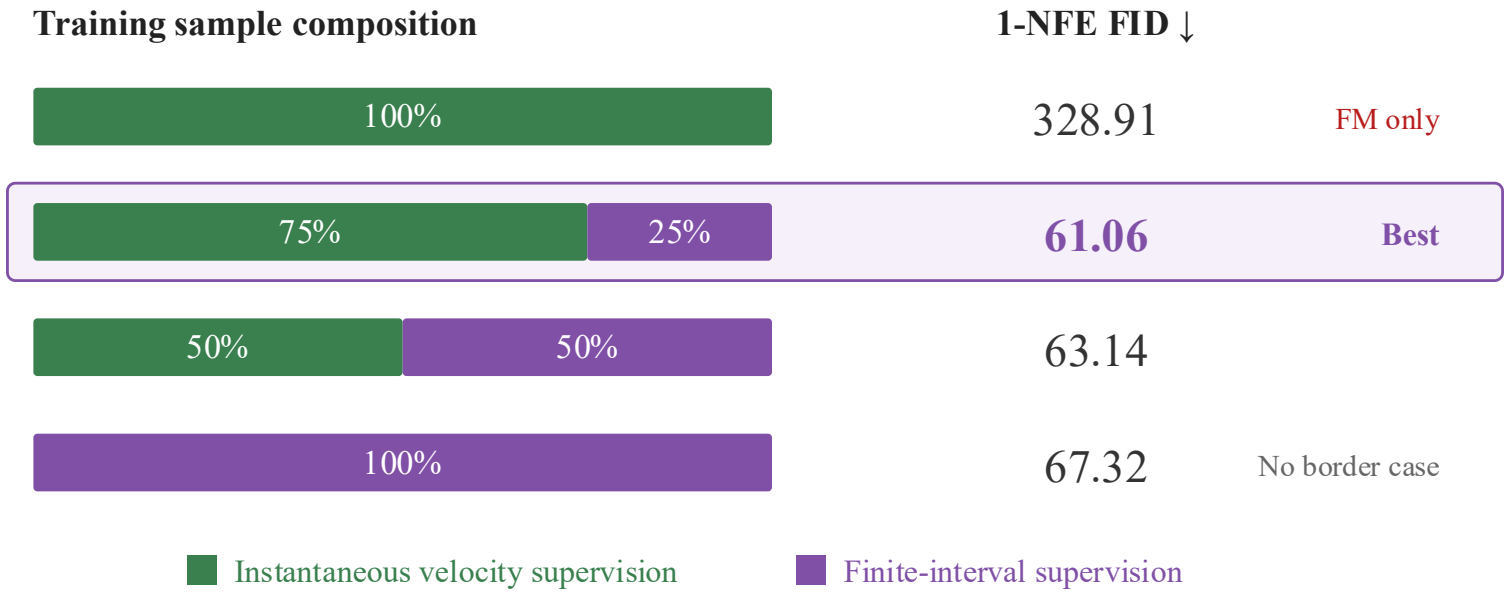
Advanced Flow Matching

❖ 평균 속도 학습에 모든 샘플을 사용하면 더 좋은 결과를 얻을 수 있을까?

- **실험 목적**: MeanFlow에서 순간 속도 학습이 실제로 필요한지 확인하기 위함
- 시간 구간의 길이가 0(순간 속도)인 학습 샘플의 비율을 조절하며 1-NFE 생성 성능을 비교

% of $r \neq t$	FID, 1-NFE
0% (= FM)	328.91
25%	**61.06**
50%	63.14
100%	67.32

(a) **Ratio of sampling** $r \neq t$. The 0% entry reduces to the standard Flow Matching baseline.



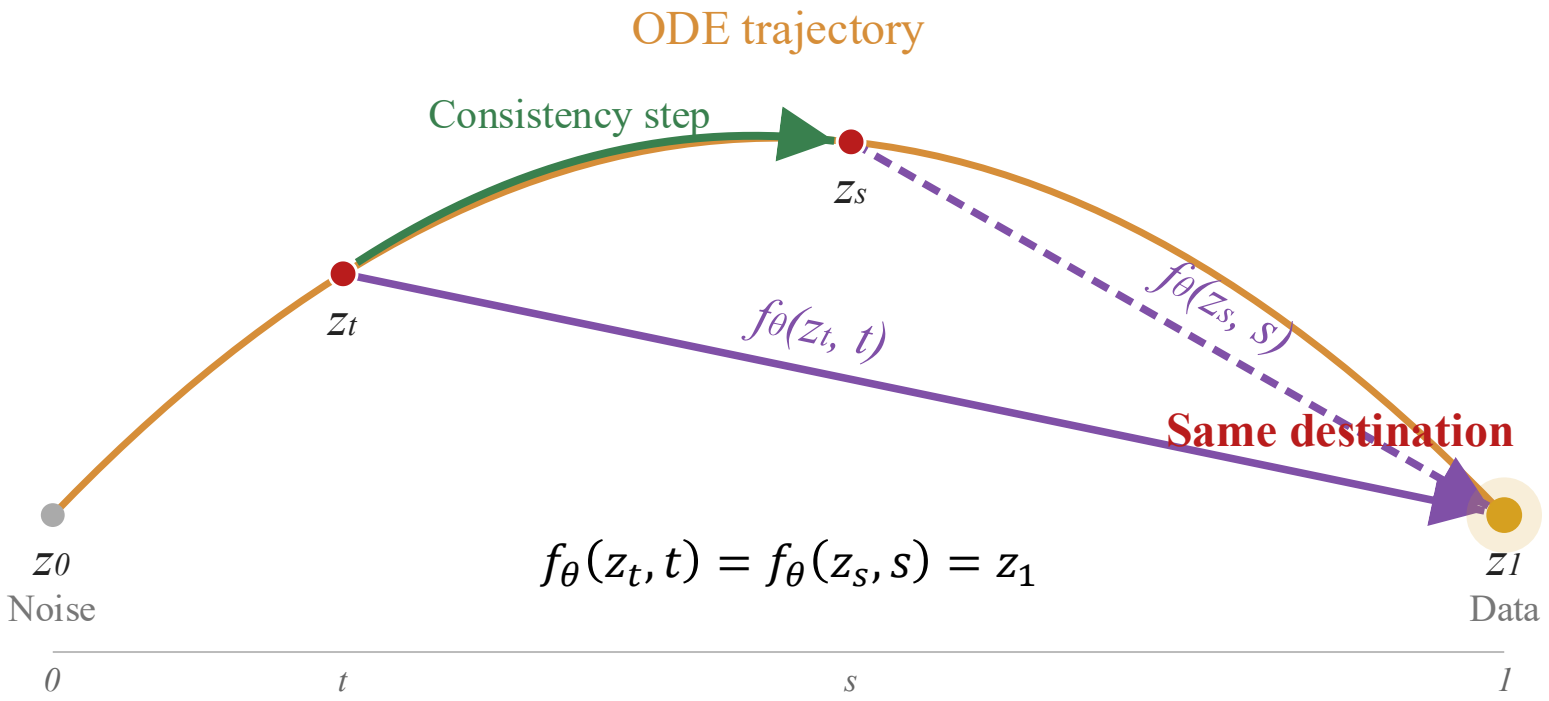
Main Question: 평균 속도를 학습하는 모델이 왜 대부분의 경우를 순간 속도를 학습하는데 쓰일까?

AlphaFlow

Advanced Flow Matching

❖ Consistency Models: 동일한 trajectory 위의 상태는 동일한 데이터로 매핑

- 서로 다른 시점의 noisy sample을 입력받더라도 동일한 trajectory 위에 존재하면 동일한 clean sample을 예측
- 이러한 예측 결과의 일관성(Self-consistency)을 학습
- **Consistency**: 출발 위치가 달라져도 **동일한 trajectory의 목적지는 같아야 함**



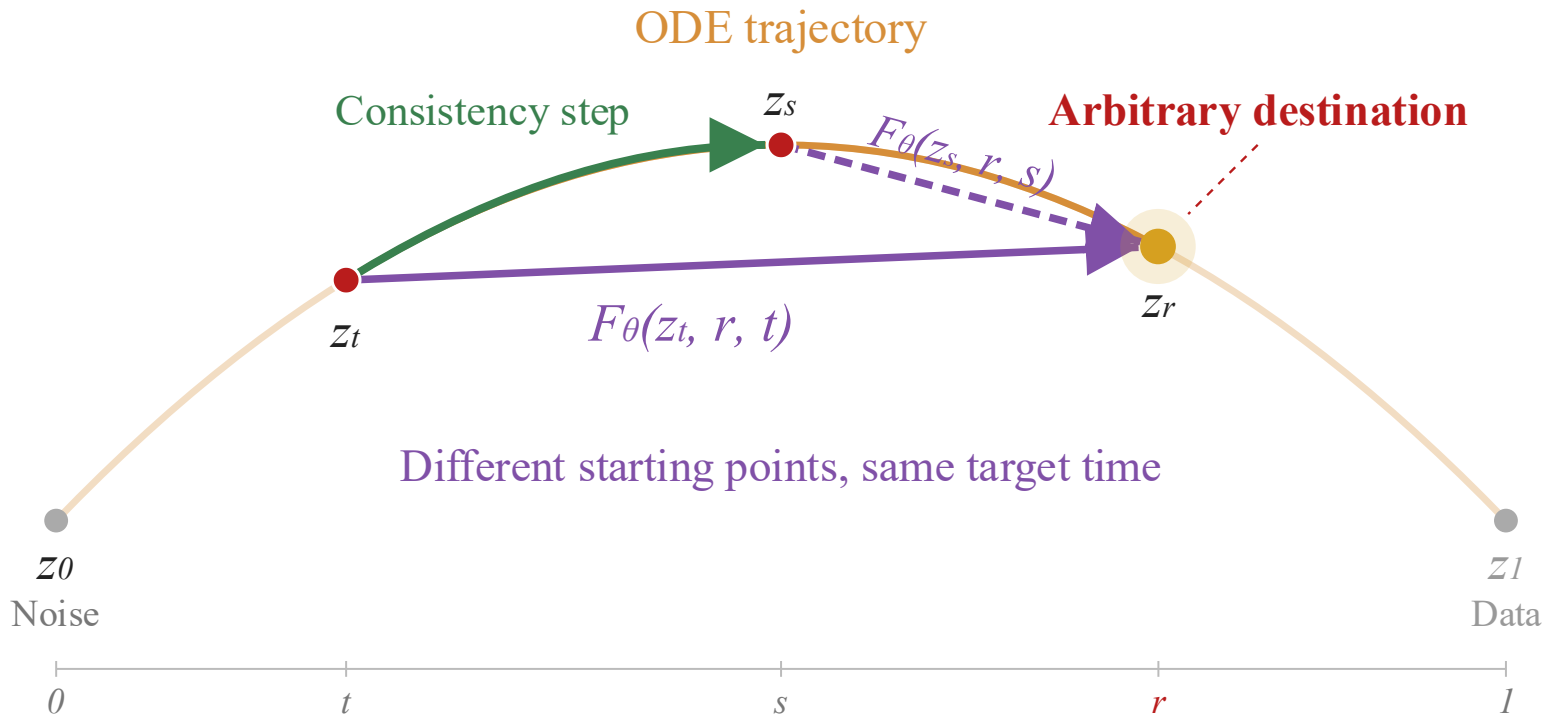
Song, Y., Dhariwal, P., Chen, M., & Sutskever, I. (2023). Consistency Models. *International Conference on Machine Learning (ICML).*

AlphaFlow

Advanced Flow Matching

❖ Consistency Trajectory Model(CTM): 임의의 시간 구간으로 Consistency 확장

- 기존 Consistency Model은 최종 clean sample에 대한 일관성을 학습
- CTM은 임의의 목표 시점을 설정하고, 해당 시점에 대한 예측 결과의 일관성을 학습
- **Trajectory Consistency**: 최종 데이터뿐만 아닌, **<u>임의의 목표 시점에 대해서도 예측의 일관성을 학습</u>**



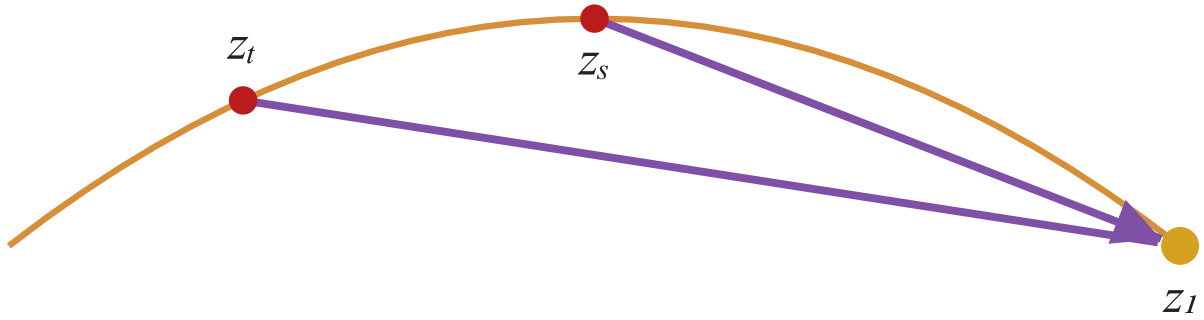
Kim, D., Lai, C. H., Liao, W., Murata, N., Takida, Y., Uesaka, T., ... & Ermon, S. (2024, May). Consistency trajectory models: Learning probability flow ode trajectory of diffusion. In *International Conference on Learning Representations* (Vol. 2024, pp. 44493-44525).

AlphaFlow

Advanced Flow Matching

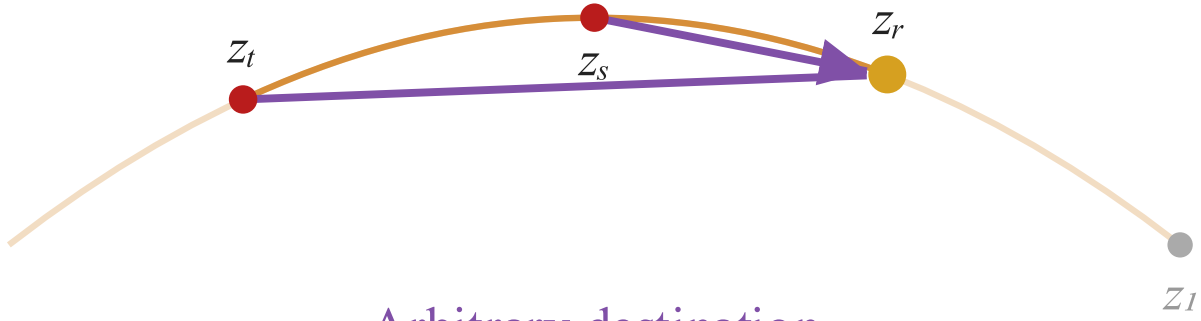
❖ **Consistency Model과 Consistency Trajectory Model의 차이는 무엇일까?**

Consistency Model



Fixed final destination

Consistency Trajectory Model



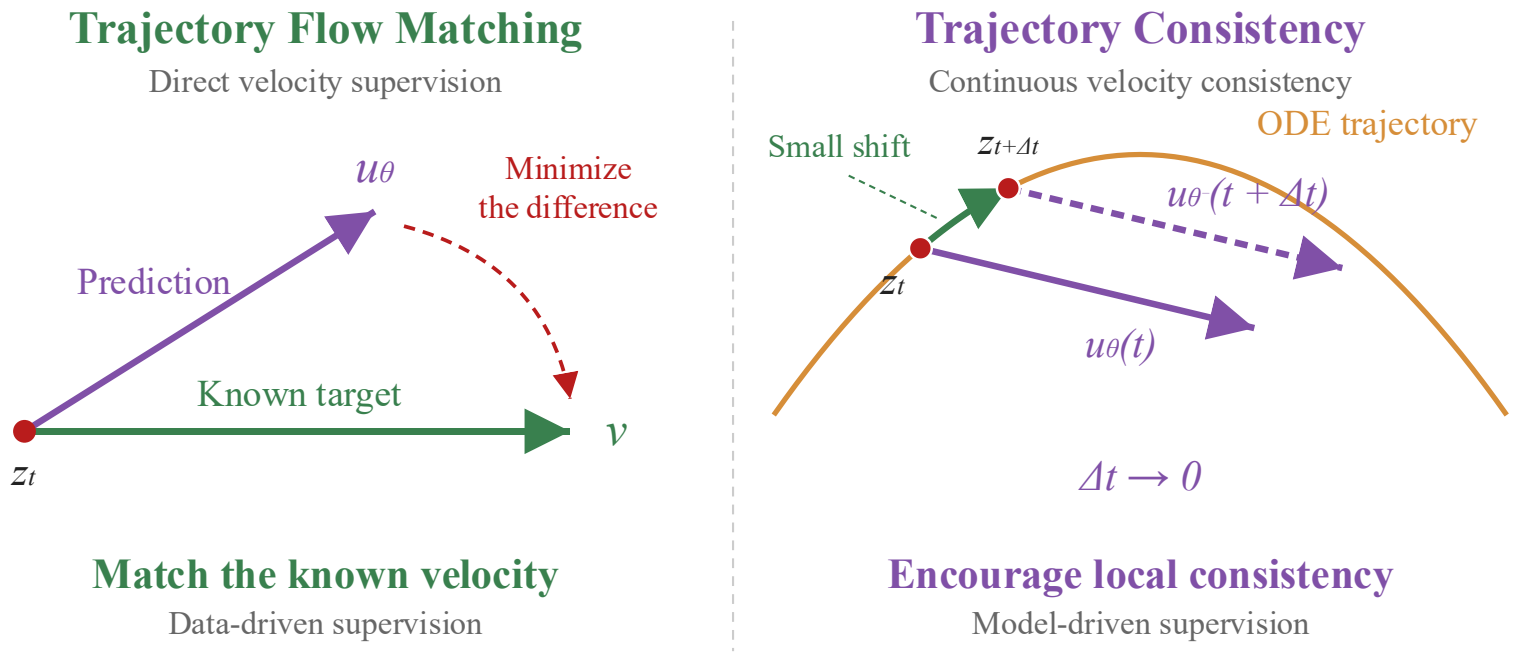
Arbitrary destination

AlphaFlow

Advanced Flow Matching

❖ MeanFlow는 실제로 무엇을 학습하고 있었는가?

- MeanFlow의 Loss를 분해하면 서로 다른 두 가지 학습 목적이 존재
- 순간 속도에 대한 직접적인 Supervision과 평균 속도에 대한 연속시간 consistency를 동시에 학습
- MeanFlow는 **<u>Flow Matching과 Consistency Training의 학습 목적을 동시에 최적화</u>**



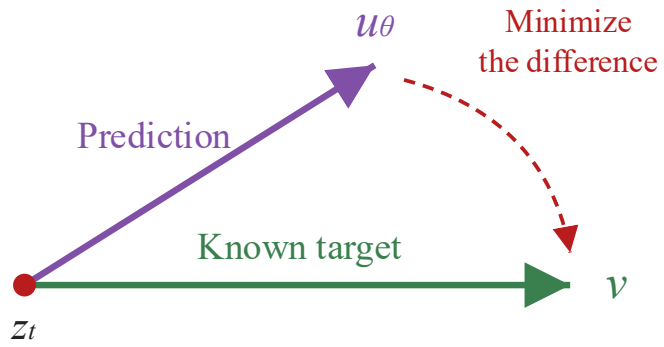
AlphaFlow

Advanced Flow Matching

❖ **MeanFlow는 실제로 무엇을 학습하고 있었는가?**

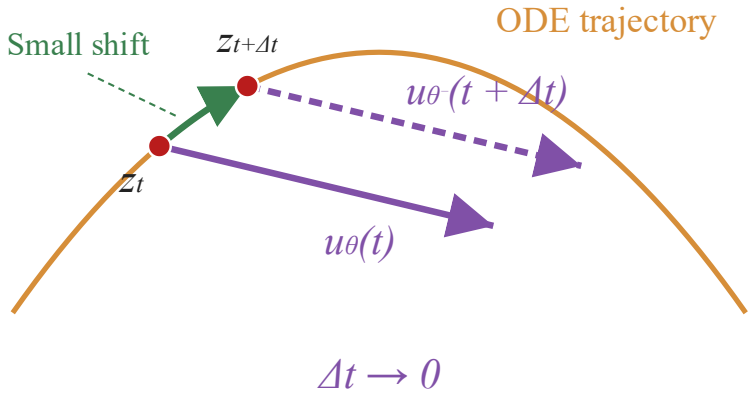
Trajectory Flow Matching

Direct velocity supervision



Trajectory Consistency

Continuous velocity consistency



$\Delta t \to 0$

Match the known velocity

Data-driven supervision

Encourage local consistency

Model-driven supervision

$$\mathcal{L}_{\text{MF}} = \mathbb{E}\left[\left\| u_\theta - v - (r-t)\frac{du_{\theta^-}}{dt}\right\|_2^2\right]$$

$$\mathcal{L}_{\text{MF}} = \mathcal{L}_{\text{TFM}} + \mathcal{L}_{\text{TC}_c} + C$$

$$\mathbb{E}[\|u_\theta - v\|_2^2]$$

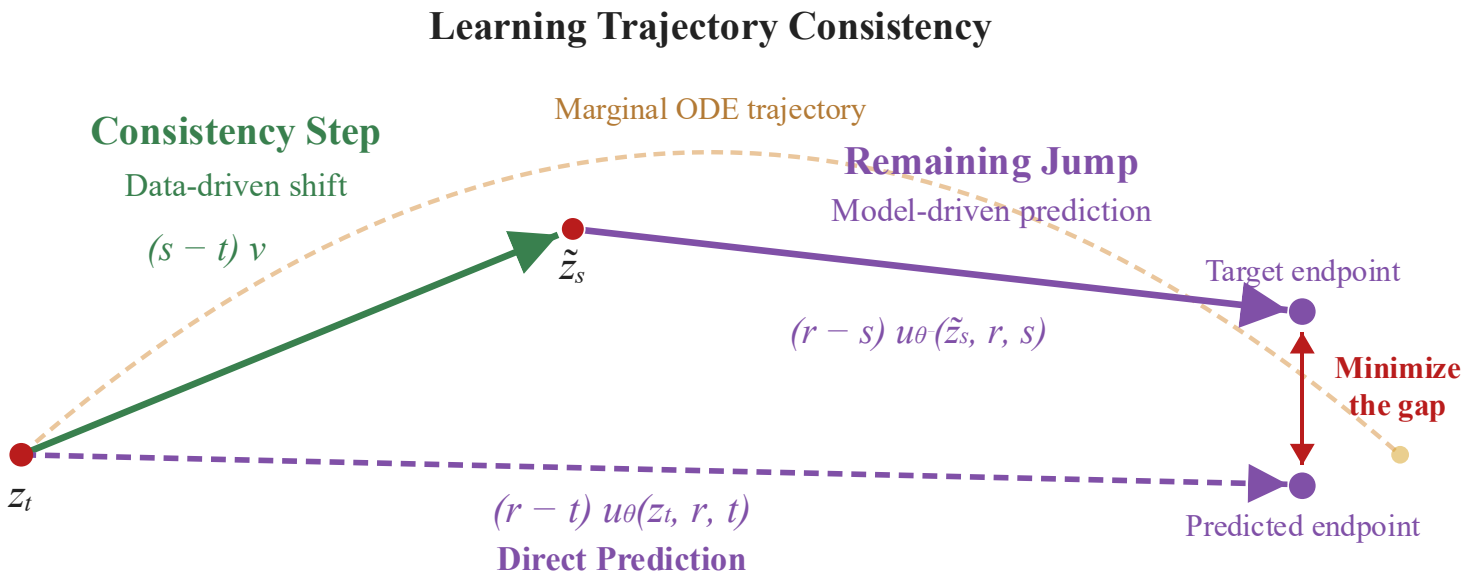
$$\mathbb{E}\left[(r-t)u_\theta^\top \frac{du_{\theta^-}}{dt}\right]$$

AlphaFlow

Advanced Flow Matching

❖ 두 부분 구간의 이동을 활용하여 전체 구간의 평균 속도를 학습

- 직접 이동한 결과와 consistency step을 거쳐 이동한 결과가 일치하도록 학습
- **조건부 순간 속도와 모델이 예측한 평균 속도를 결합하여 학습 target 구성**



$$\mathcal{L}_\alpha(\theta) = \mathbb{E}\left[\frac{1}{\alpha}\left\|u_\theta(z_t, r, t) - \mathrm{sg}\left(u_{\text{target}}\right)\right\|_2^2\right]$$

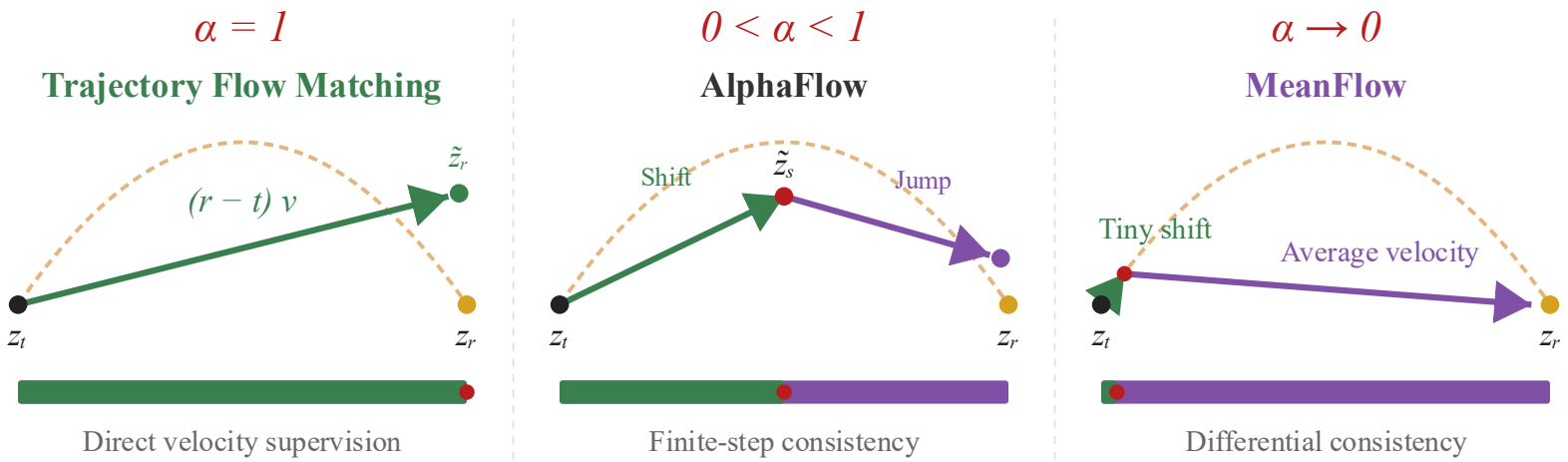
AlphaFlow

Advanced Flow Matching

❖ α를 조절하며 학습 목적을 점진적으로 변환

- 학습 초기에는 데이터로부터 직접 계산할 수 있는 velocity를 중심으로 학습
- **이후 α를 점진적으로 감소시키며 MeanFlow의 학습 목적으로 전환**

$$u_{\text{target}} = \alpha v + (1-\alpha)u_{\theta^-}(\tilde{z}_s, r, s)$$

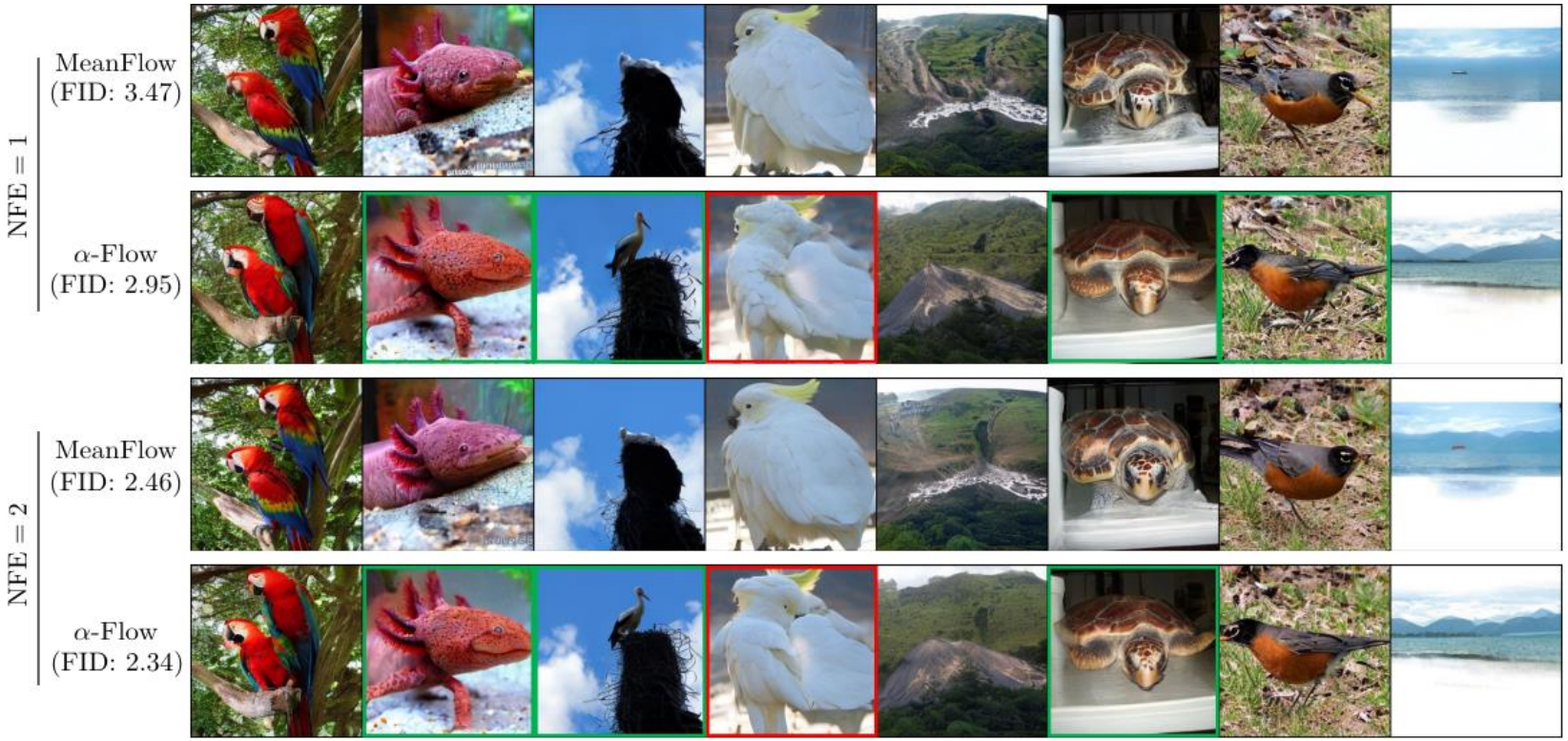


AlphaFlow

Advanced Flow Matching

❖ AlphaFlow는 MeanFlow보다 이미지 생성을 잘할까?

- 추가적인 추론 step 없이도 이미지의 구조와 세부 표현을 개선

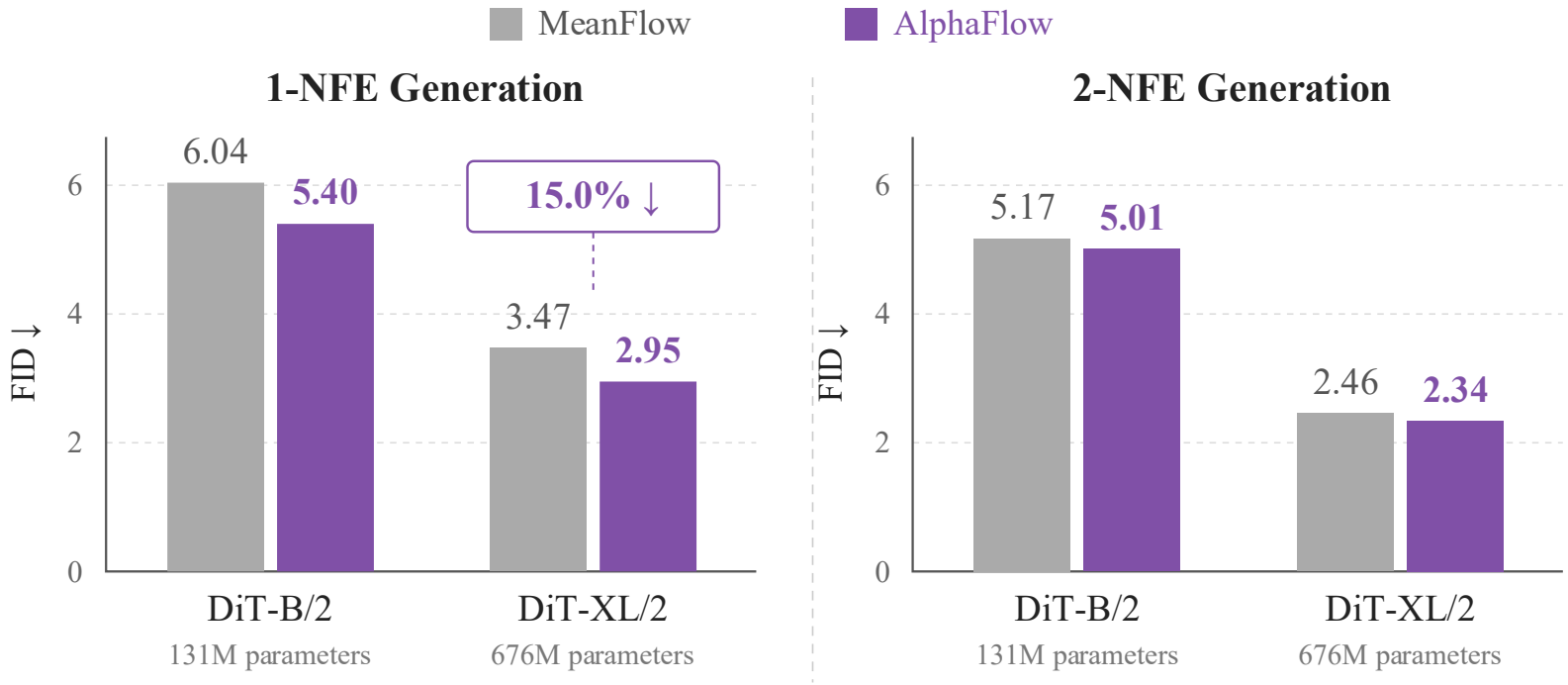


AlphaFlow

Advanced Flow Matching

❖ **학습 전략의 변화만으로 생성 품질을 향상시킬 수 있을까?**

- 동일한 DiT architecture와 학습 epoch를 사용해 MeanFlow와 AlphaFlow의 생성 성능 비교
- ImageNet 256 x 256에서 1-NFE 및 2-NFE 생성 결과를 FID로 평가

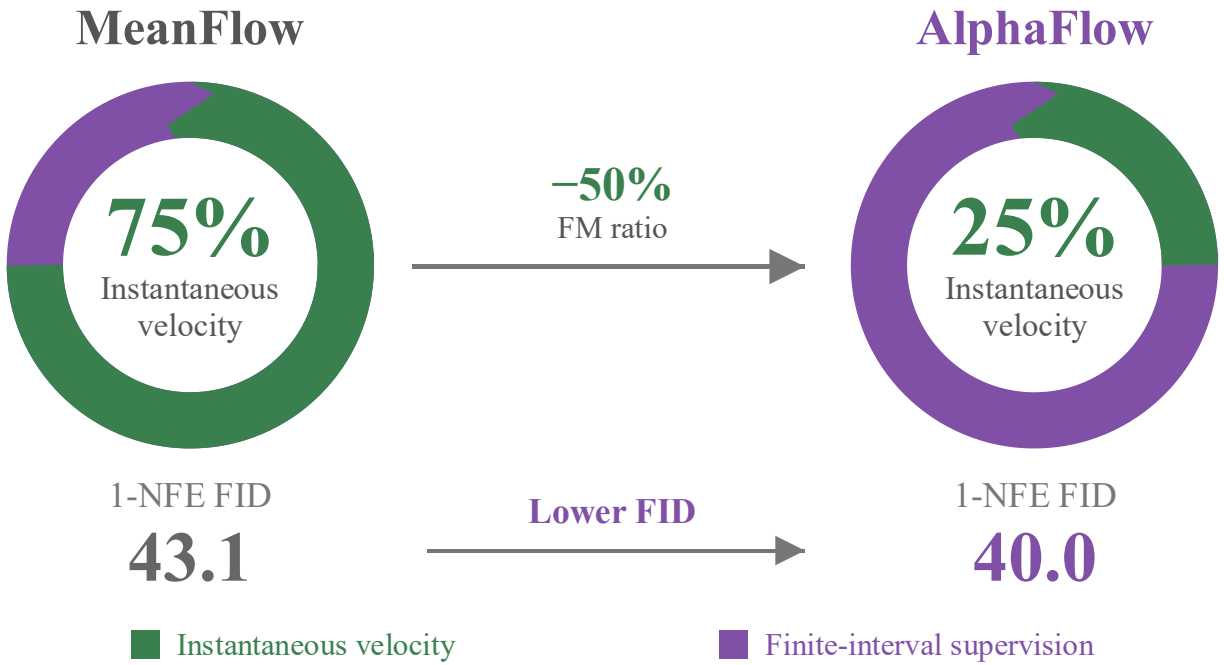


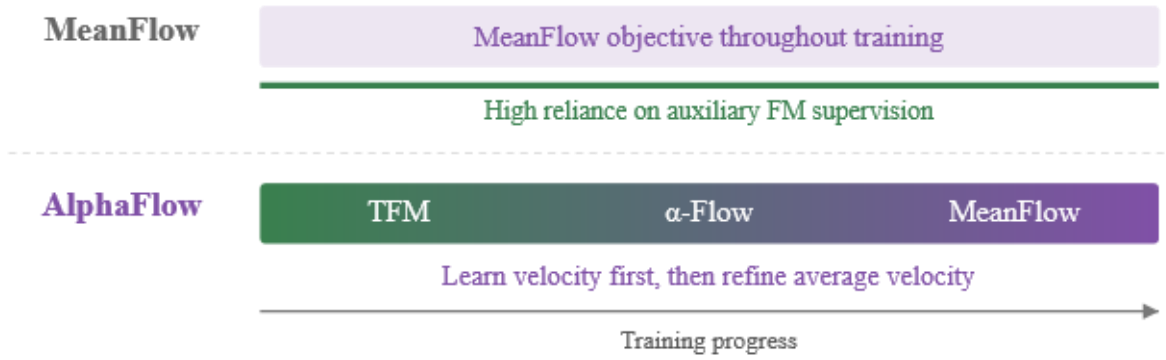
AlphaFlow

Advanced Flow Matching

❖ AlphaFlow는 순간 속도 학습에 대한 의존성을 줄일 수 있을까?

- 기존 MeanFlow는 높은 비율의 순간 속도 학습을 필요로 함
- AlphaFlow의 학습 전략이 이러한 보조 학습의 필요성을 줄일 수 있는지 확인

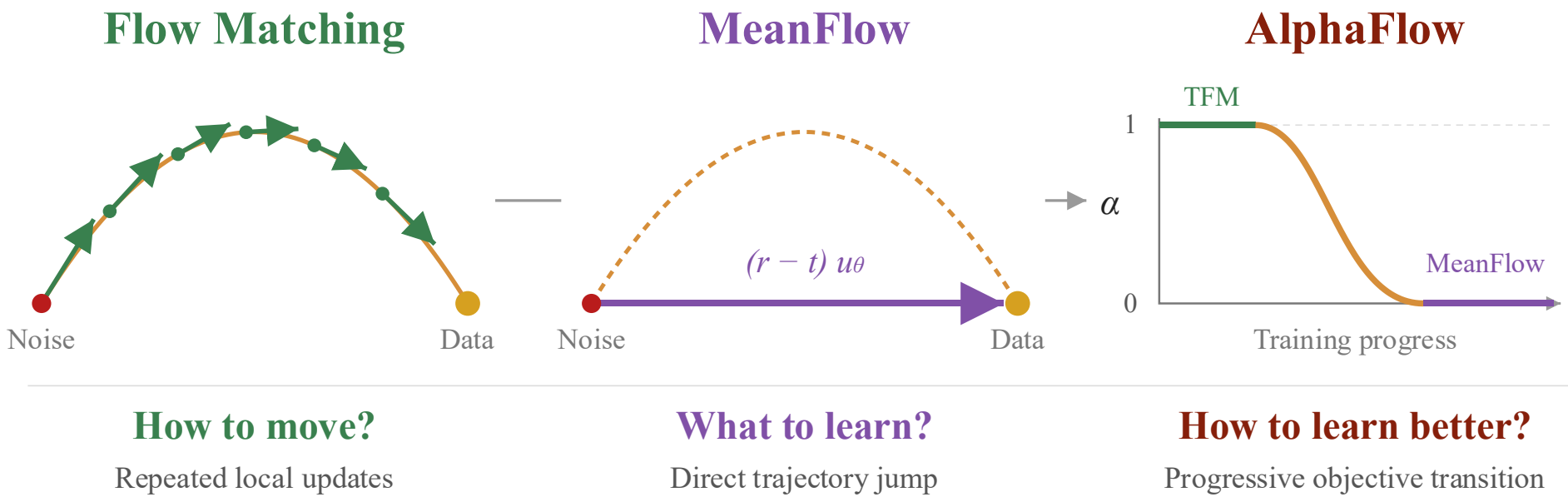




Advanced Flow Matching

❖ From Learning Velocity to Learning Transport

- MeanFlow: 순간 속도 대신 평균 속도를 학습하여 반복적인 ODE 적분의 필요성을 줄임
- AlphaFlow: 학습 목적을 점진적으로 전환하여 평균 속도의 학습을 개선



고맙습니다