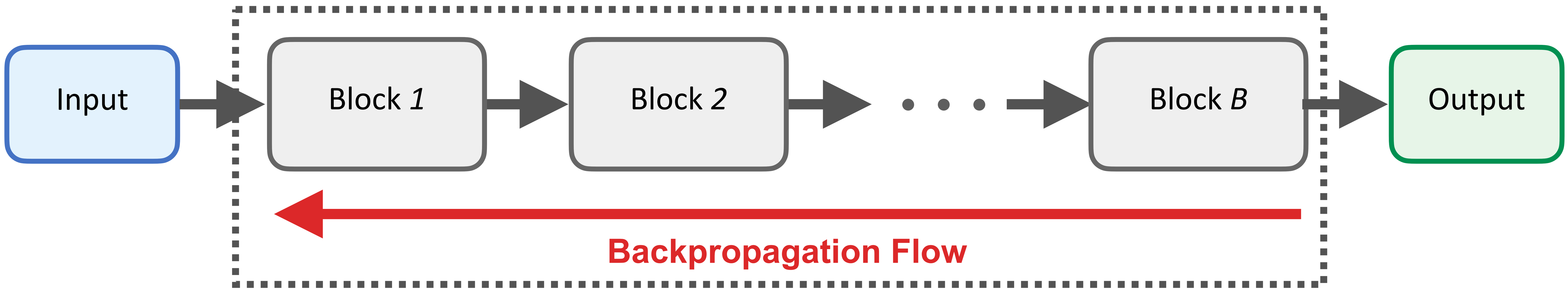
DiffusionBlocks: Blockwise Training for Generative Models via Score-Based Diffusion

Makoto Shing, Takuya Akiba

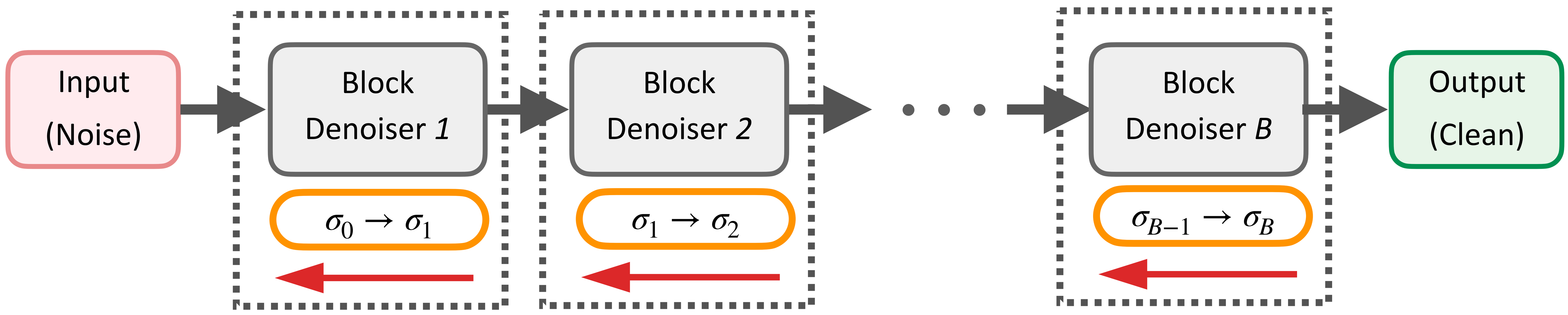
sakana.ai

End-to-end Backpropagation



Ours: DiffusionBlocks

Each block ***independently*** learns diffusion-based denoising for a specific range.



Background

🚨 ***Problem:***

- End-to-end backprop requires storing *ALL* activations
- Memory bottleneck limits large-scale training accessibility

🚀 ***Key Idea***

- **Interpret network blocks as diffusion denoising steps!**

Method

Core Framework

Interpret neural networks as reverse diffusion process:

- **Input:** $z_{\sigma_{\max}} \sim \mathcal{N}(0, \sigma_{\max}^2 I)$ (maximum noise)
- **Output:** $z_0 \sim p_{\text{data}}$ (clean data)
- **Block i:** denoising range $[\sigma_i, \sigma_{i+1}]$

Training objective for block i:

$$L(\theta_i) = \mathbb{E}\left[w(\sigma)\|D_{\theta_i}(\mathbf{z}_\sigma, \sigma, \mathbf{x}) - \mathbf{y}\|_2^2\right],$$

where $z_\sigma = z_0 + \sigma\epsilon, \epsilon \sim \mathcal{N}(0, I)$.

Equi-Probability Partitioning

- How we partition the noise levels among blocks is crucial.
- Ensure that each block handles a same difficulty:

$$\int_{\sigma_i}^{\sigma_{i+1}} p_\sigma(\sigma) d\sigma = \frac{1}{B}$$

- Noise boundaries:

$$\sigma_i = \exp\left(P_{\text{mean}} + P_{\text{std}} \cdot \Phi^{-1}(p_i)\right)$$

where p_i ensures equal cumulative probability and Φ^{-1} is the inverse CDF of the standard normal distribution.

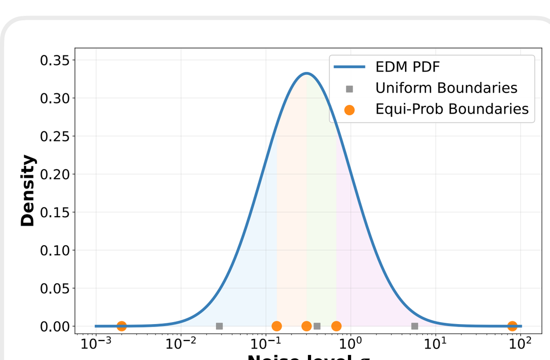


Fig 1. Colored regions represent individual blocks under our partitioning,

Memory Efficiency

- Each block trained with independent objectives, resulting in **no gradient communication between blocks!**
- Memory: O(L/B) vs. O(L) for end-to-end backprop

Experiments

Image Generation

Settings:

- **Dataset:** CIFAR-10 / ImageNet
- **Architecture:** DiT-S / DiT-L partitioned into 4 blocks
- **Evaluation:** FID scores (lower = better)

Method	CIFAR-10	ImageNet
End-to-end Backdrop	41.87	16.62
DiffusionBlocks	**41.39**	**15.55**

✅ **Superior quality** + **4× memory reduction** during training
✅ **3× faster** inference

Language Modeling

Settings:

- **Dataset:** LM-1B
- **Architecture:** Llama-style (12 layers) partitioned into 4 blocks
- **Evaluation:** MAUVE score (higher = better)

Method	MAUVE (↑)
End-to-end Backdrop	0.595
DiffusionBlocks	**0.779**

✅ **Superior quality** + **4× memory reduction** during training

Ablation Study

Table 1. Effect of block partitioning strategy

Method	FID (↓)
Uniform	68.06
Equi-Probability	**45.50**

Table 2. Effect of block count

Method	FID (↓)	Speed
B=2	38.58	2x
B=3	41.39	2x
B=4	41.39	**4x**
B=6	53.74	**6x**