Model-based Diffusion

for Trajectory Optimization

Chaoyi Pan*, Zeji Yi*, Guanya Shi+, Guannan Qu+ 2024.6.24

Background: Diffusion Model in Planning

Background: Diffusion Model in Planning

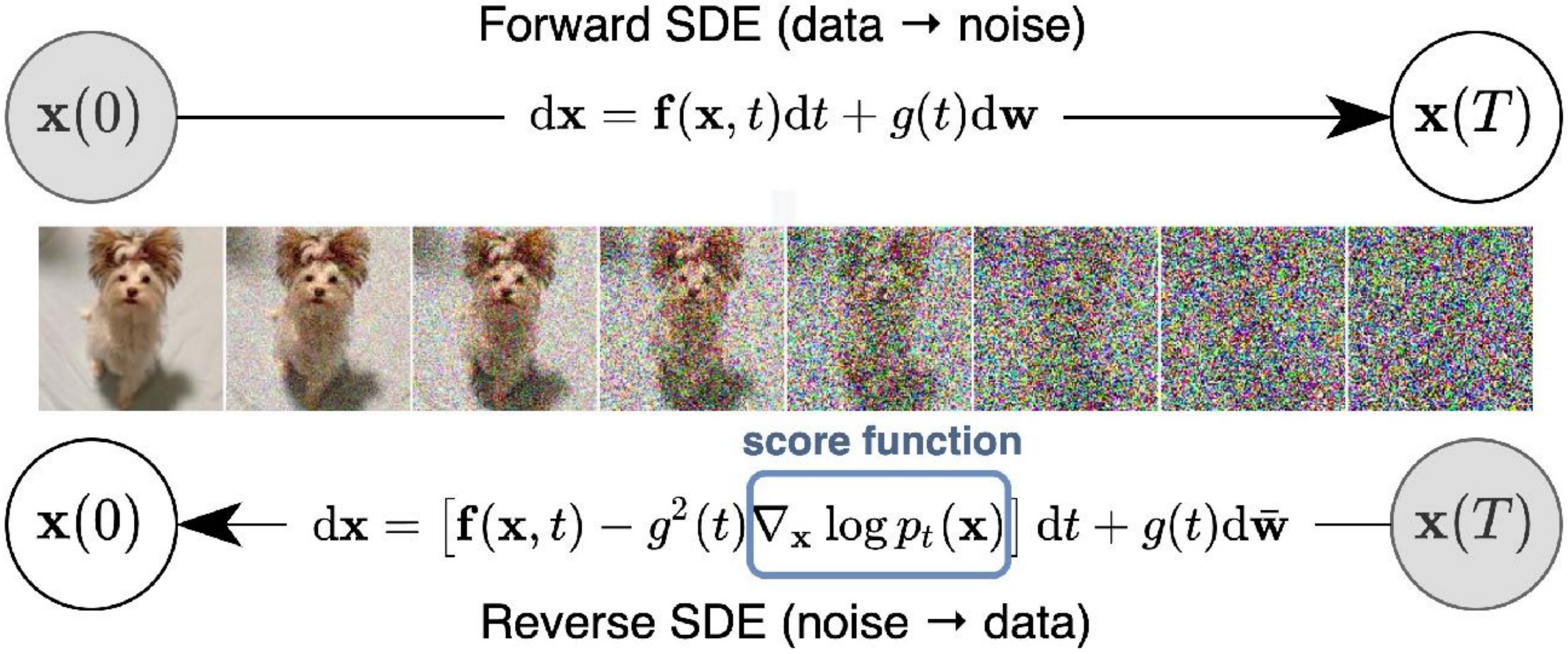
Diffusion model as a powerful sampler

- A diffusion model is a generative model capable of generating **samples from a given distribution**

Background: Diffusion Model in Planning

Diffusion model as a powerful sampler

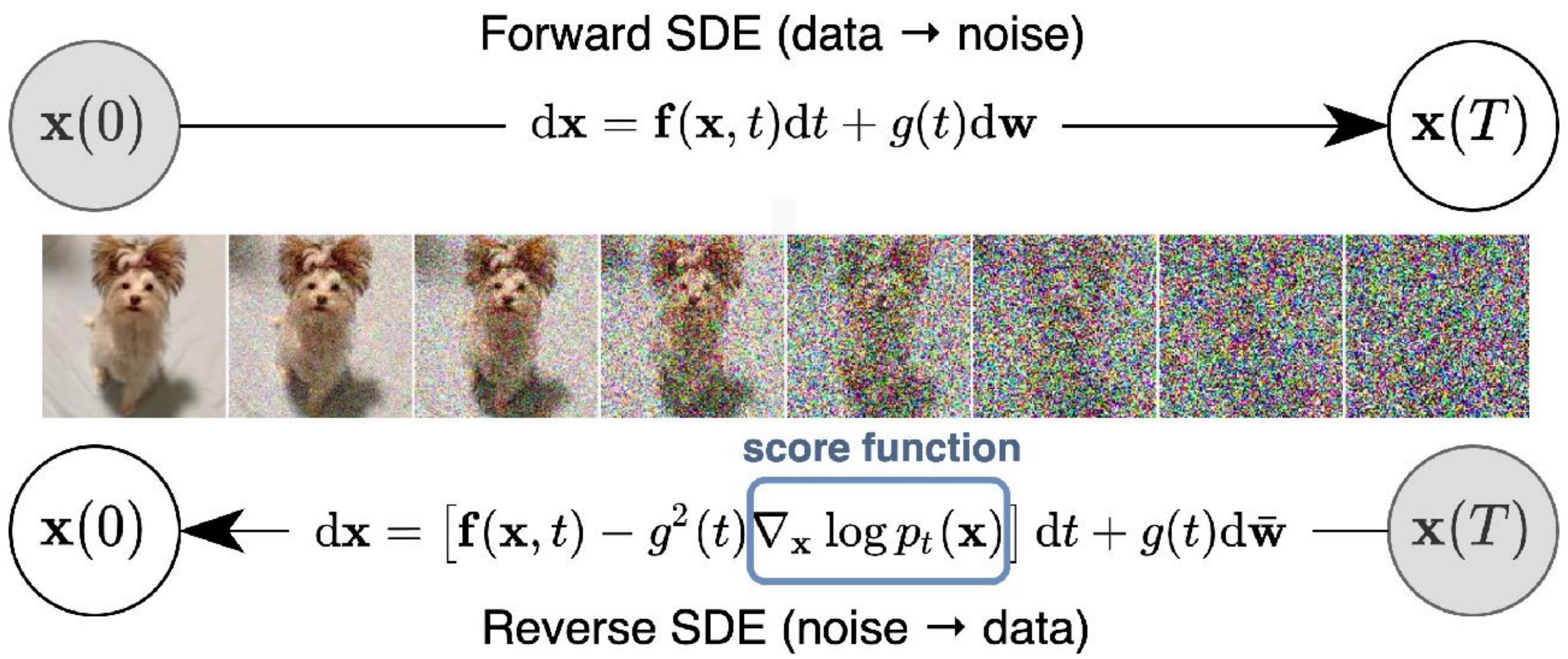
- Forward: diffusion achieve that by corrupt the distribution with noise to make it **smooth and easier to sample from**



Background: Diffusion Model in Planning

Diffusion model as a powerful sampler

- Forward: diffusion achieve that by corrupt the distribution with noise to make it **smooth and easier to sample from**



$$p_{i|i-1}(\,\cdot \mid Y^{(i-1)}) \sim \mathcal{N}(\sqrt{\alpha_i}\,Y^{(i-1)}, \sqrt{1-\alpha_i}\,I)$$

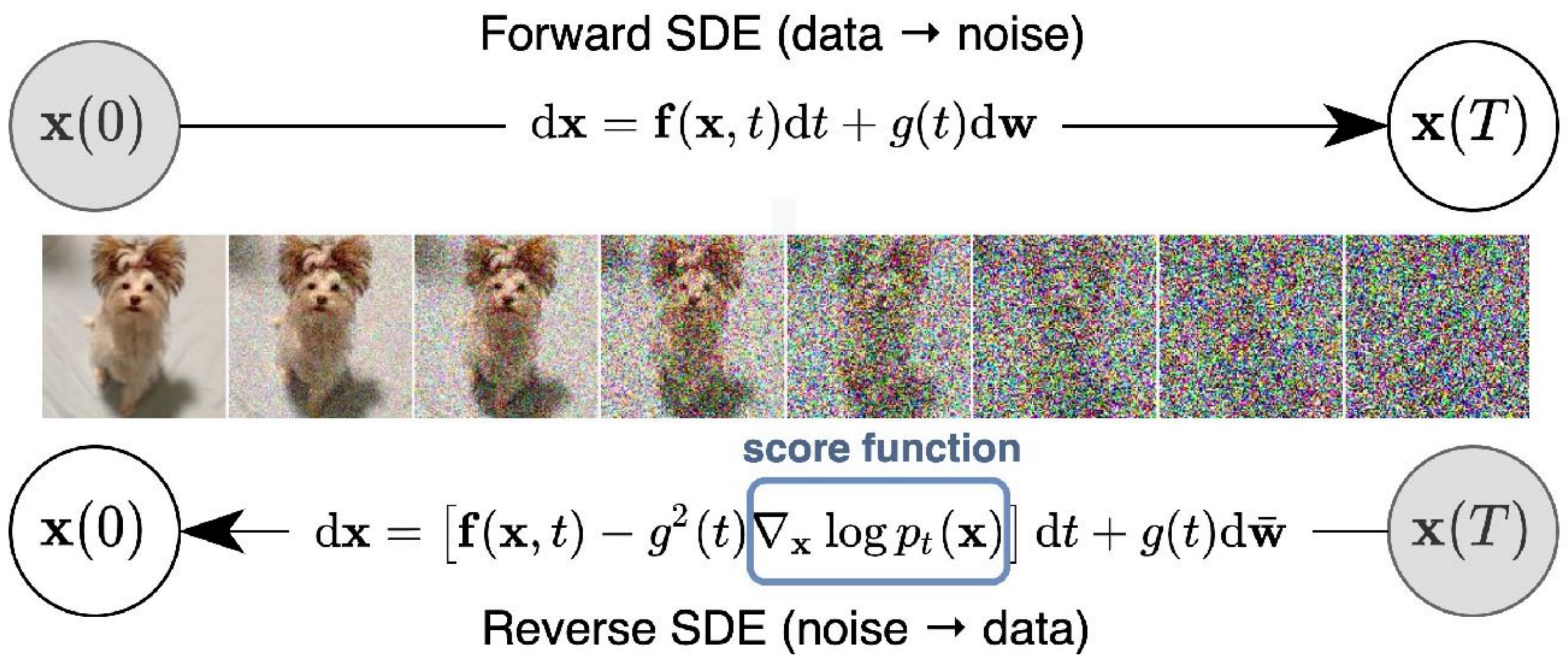
$$p_{i|0}(\,\cdot \mid Y^{(0)}) \sim \mathcal{N}(\sqrt{\bar{\alpha}_i}\,Y^{(0)}, \sqrt{1-\bar{\alpha}_i}\,I), \quad \bar{\alpha}_i = \prod_{k=1}^{i} \alpha_k\,.$$

$$Y_k^{(i-1)} = \frac{1}{\sqrt{\alpha_i}}\left(Y_k^{(i)} + \frac{1-\alpha_i}{2}\nabla_{Y^{(i)}}\log p_i(Y_k^{(i)})\right) + \sqrt{1-\alpha_i}\,\mathbf{z}_i$$

Background: Diffusion Model in Planning

Diffusion model as a powerful sampler

- Reverse: diffusion model can recover the original distribution iteratively, where the **score function** is the key component



$$p_{i|i-1}(\,\cdot\mid Y^{(i-1)}) \sim \mathcal{N}(\sqrt{\alpha_i}Y^{(i-1)}, \sqrt{1-\alpha_i}I)$$

$$p_{i|0}(\,\cdot\mid Y^{(0)}) \sim \mathcal{N}(\sqrt{\bar{\alpha}_i}Y^{(0)}, \sqrt{1-\bar{\alpha}_i}I), \quad \bar{\alpha}_i = \prod_{k=1}^{i} \alpha_k\,.$$

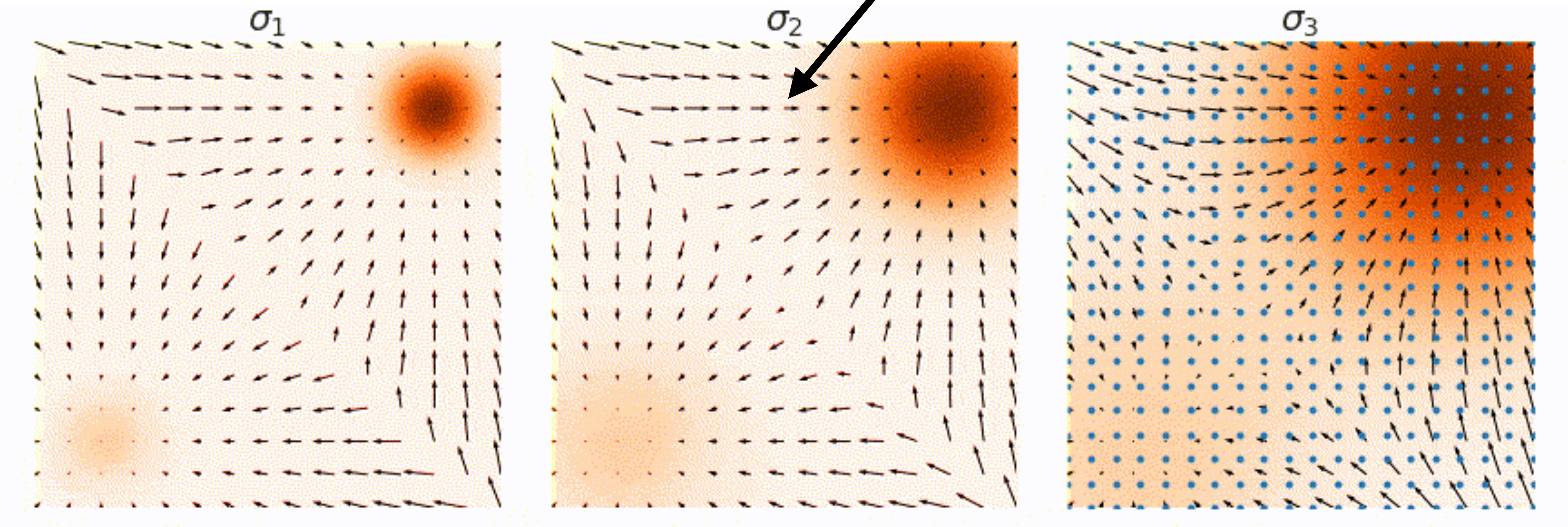
$$Y_k^{(i-1)} = \frac{1}{\sqrt{\alpha_i}}\left(Y_k^{(i)} + \frac{1-\alpha_i}{2}\nabla_{Y^{(i)}} \log p_i(Y_k^{(i)})\right) + \sqrt{1-\alpha_i}\mathbf{z}_i$$

Background: Diffusion Model in Planning

Diffusion model as a powerful sampler

- Reverse: diffusion model can recover the original distribution iteratively, where the **score function** is the key component

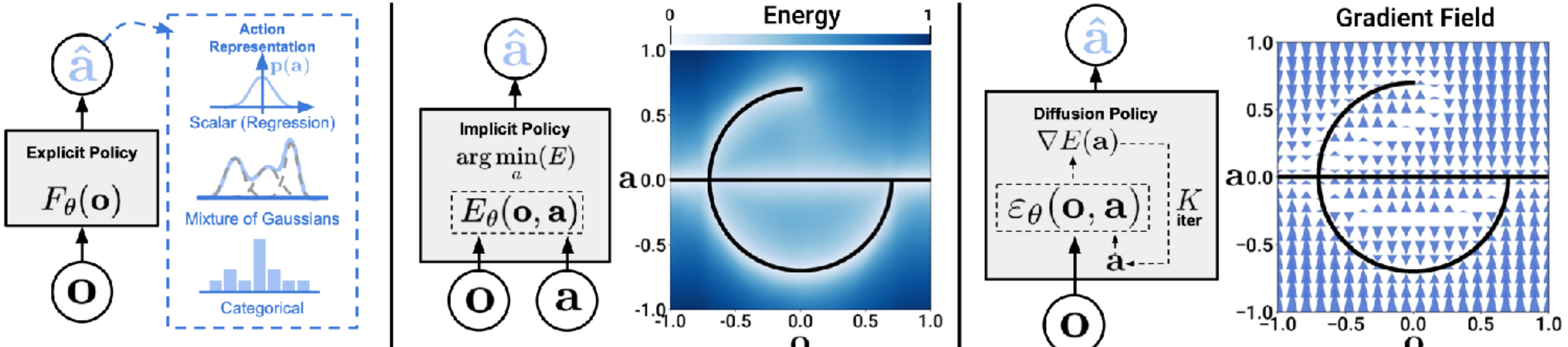
$$Y_k^{(i-1)} = \frac{1}{\sqrt{\alpha_i}}\left(Y_k^{(i)} + \frac{1-\alpha_i}{2}\nabla_{Y^{(i)}}\log p_i(Y_k^{(i)})\right) + \sqrt{1-\alpha_i}\mathbf{z}_i$$



Background: Diffusion Model in Planning

Diffusion model as a powerful sampler

- Advantages
  - **Multimodal**: It effectively handles multimodal distributions, a challenge often encountered when directly predicting distributions.



(a) Explicit Policy (b) Implicit Policy (c) Diffusion Policy

Background: Diffusion Model in Planning

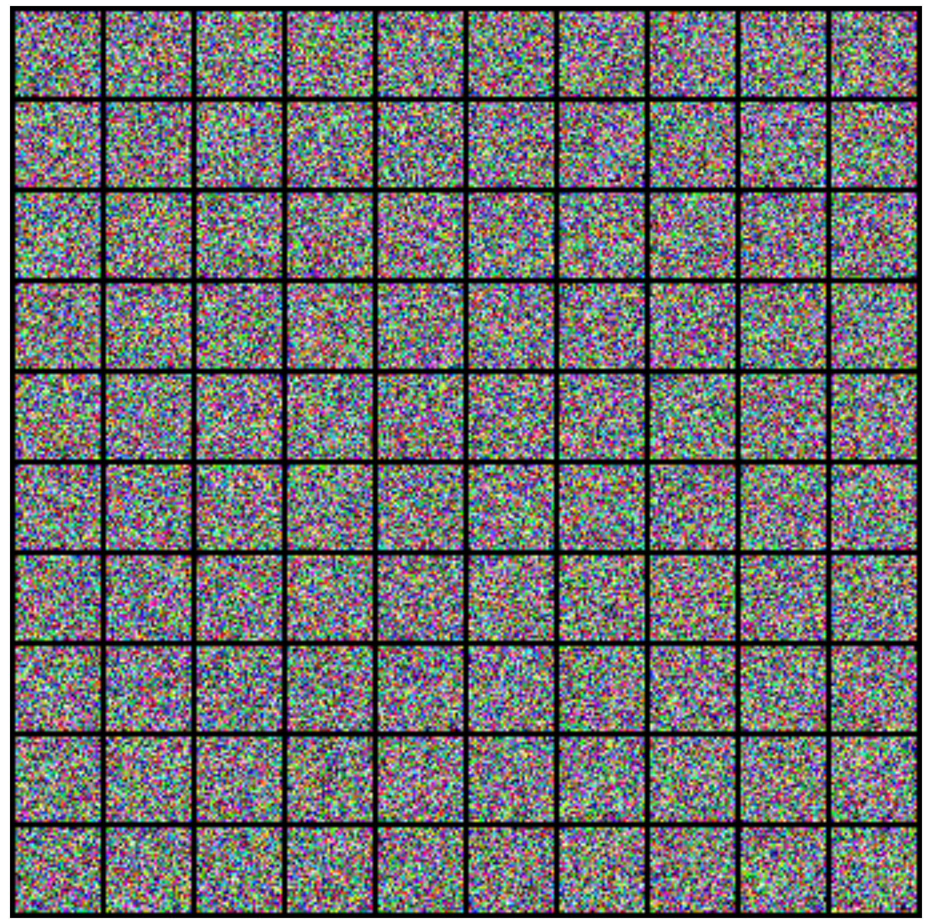
Diffusion model as a powerful sampler

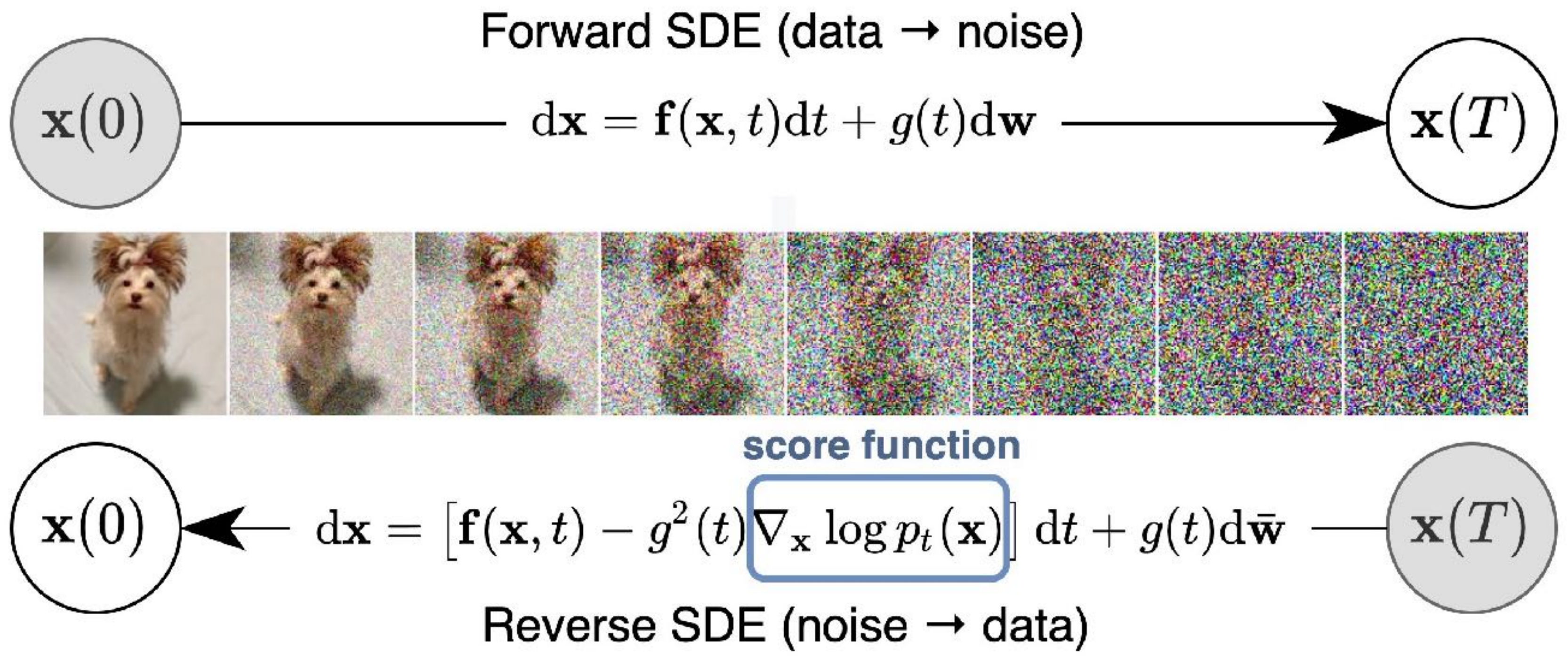
- Advantages
  - **Multimodal**: It effectively handles multimodal distributions, a challenge often encountered when directly predicting distributions.
  - **Scalable**: This approach scales well with high-dimensional distribution matching problems, making it versatile for various applications.
  - **Stable**: Grounded in solid math and a standard multi-stage diffusion training procedure, the model ensures stability during training.
  - **Non-autoregressive**: Its capability to predict entire trajectory sequences in a non-autoregressive manne.

Background: Diffusion Model in Planning

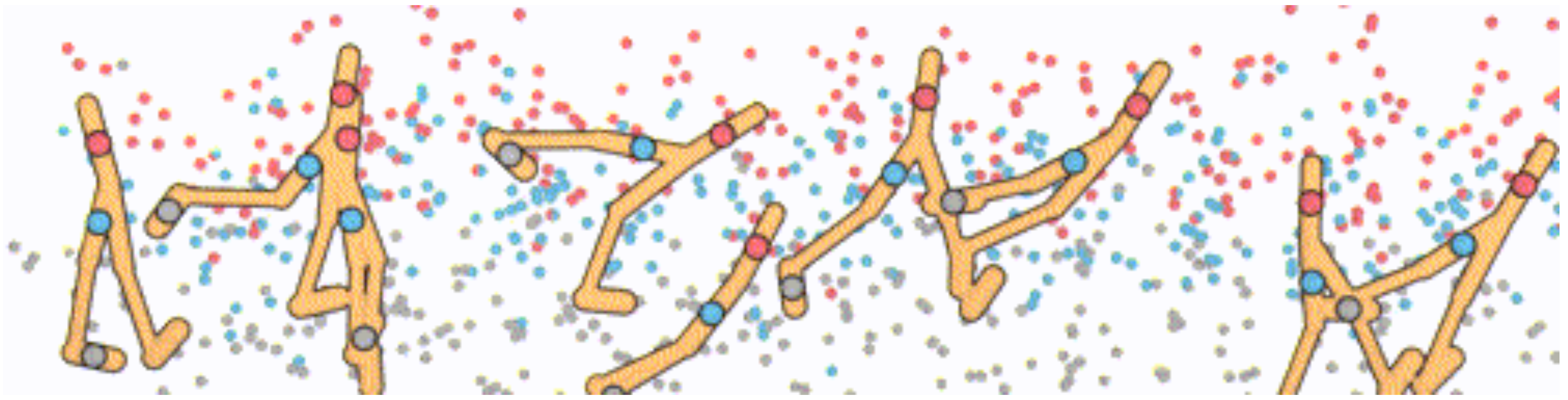
Diffusion solving trajectory optimization as a sampling problem

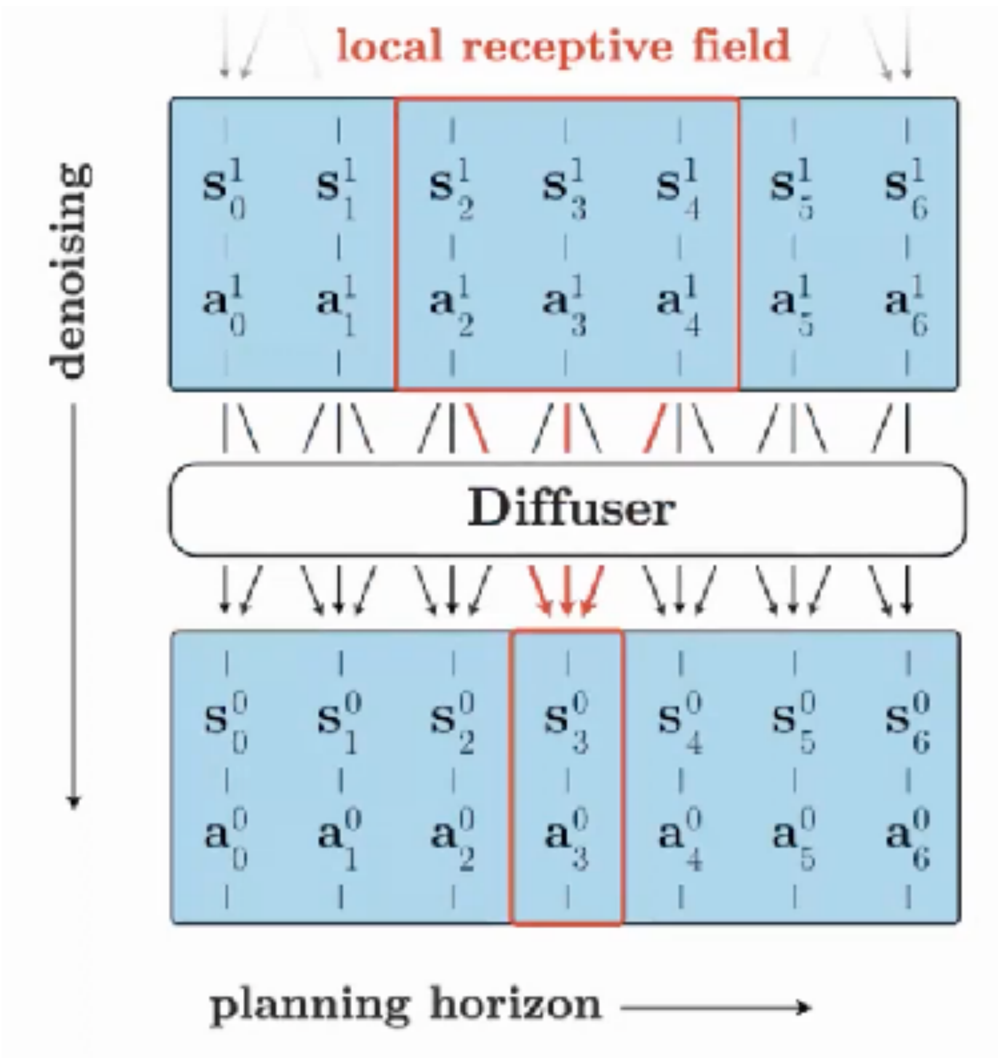
- Image Generation





- Planning

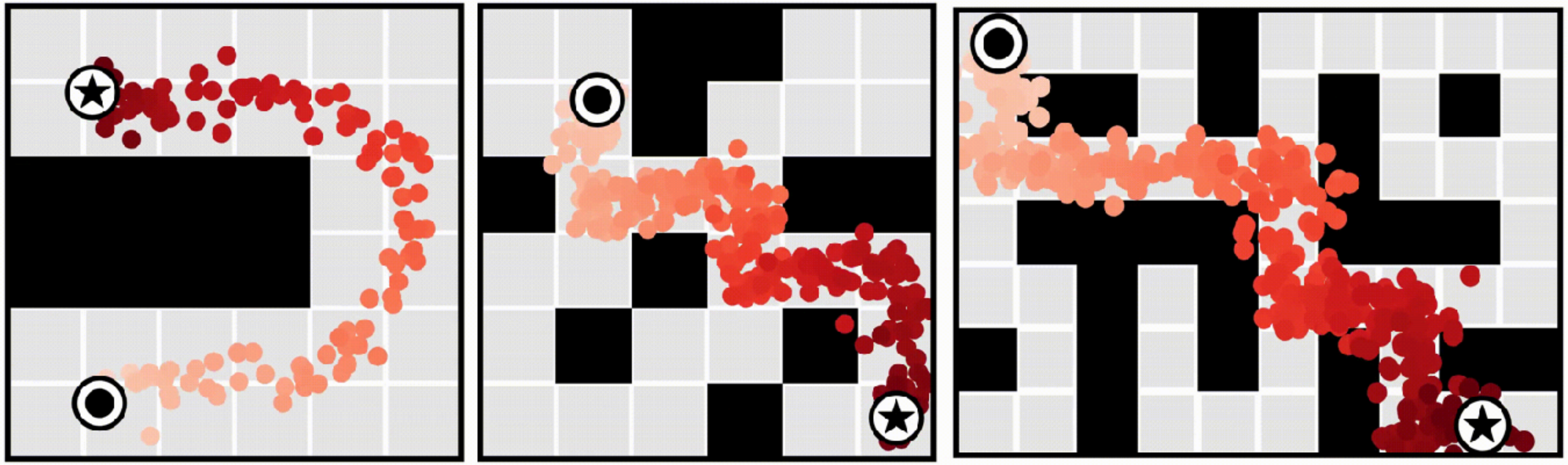




Background: Diffusion Model in Planning

Diffusion solving trajectory optimization as a sampling problem

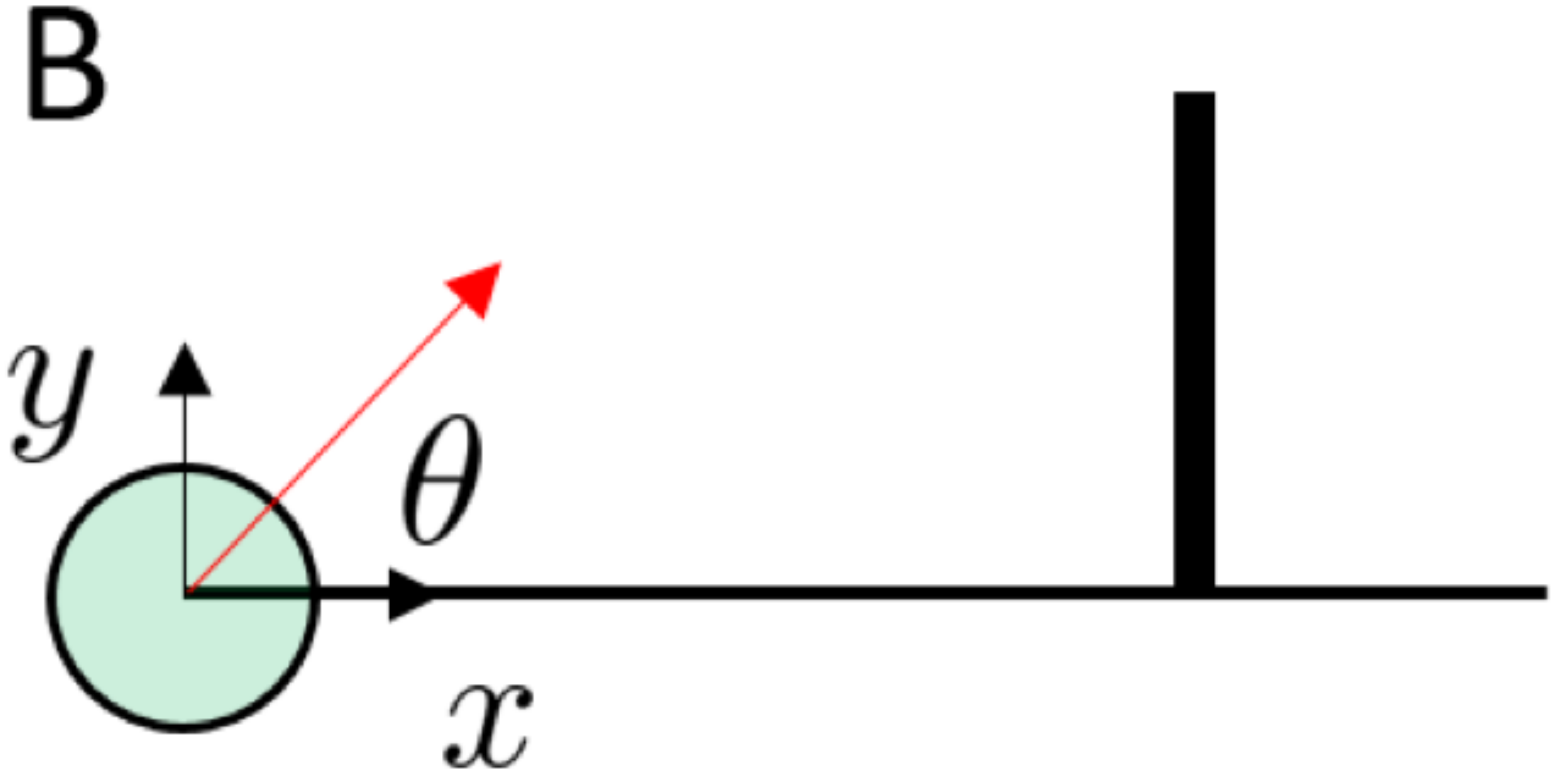
- Powerful trajectory generator but highly depends on the training data.



Background: Diffusion Model in Planning

Example: Throwing Ball Over a Wall with Diffusion

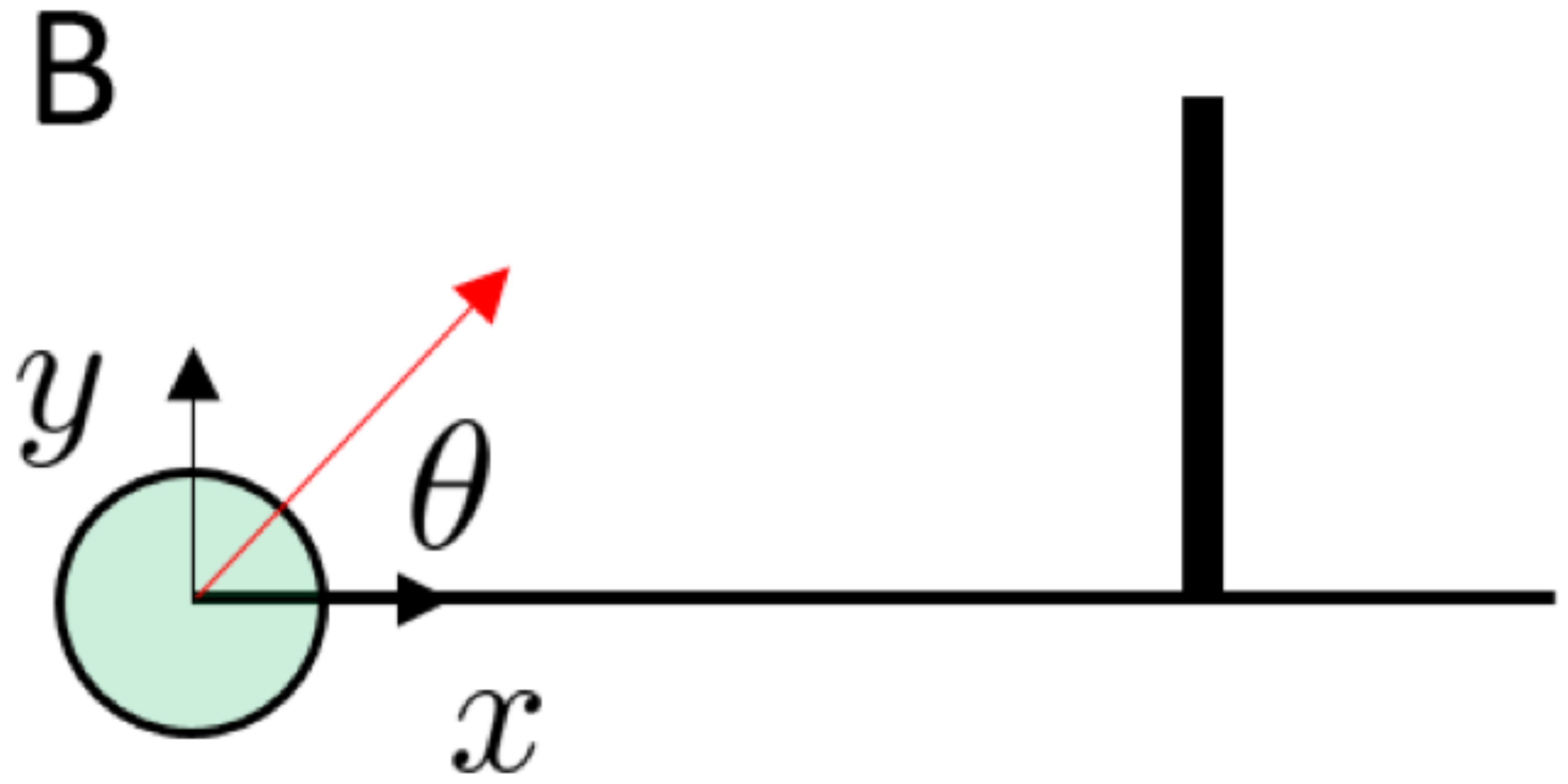
- Task: a throwing ball over a wall to maximize the distance

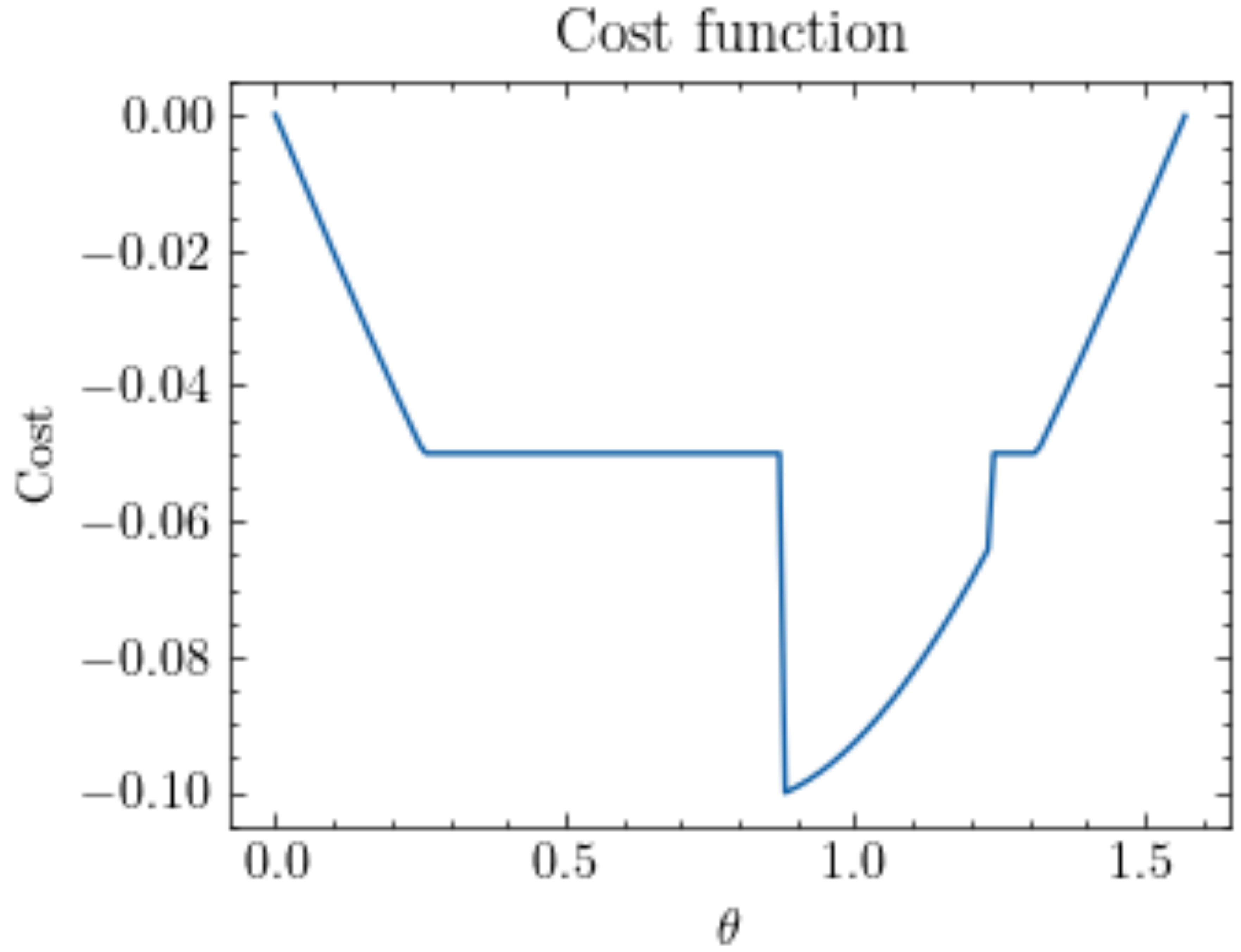


Background: Diffusion Model in Planning

Example: Throwing Ball Over a Wall with Diffusion

- Task: a throwing ball over a wall to maximize the distance



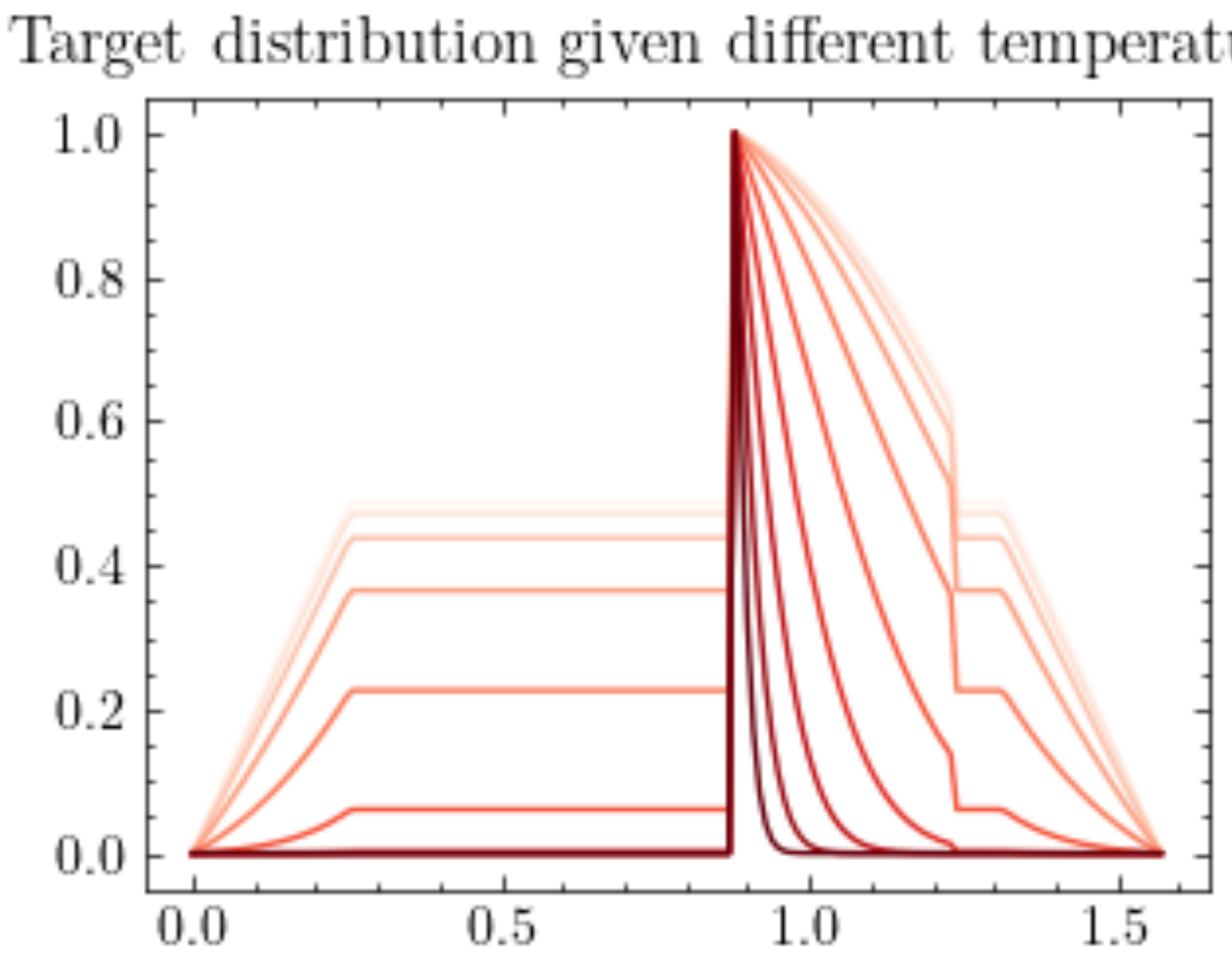


$$\min_{u} J(f(x_{\text{init}}, u)) = l(f(x_{\text{init}}, u), u)$$

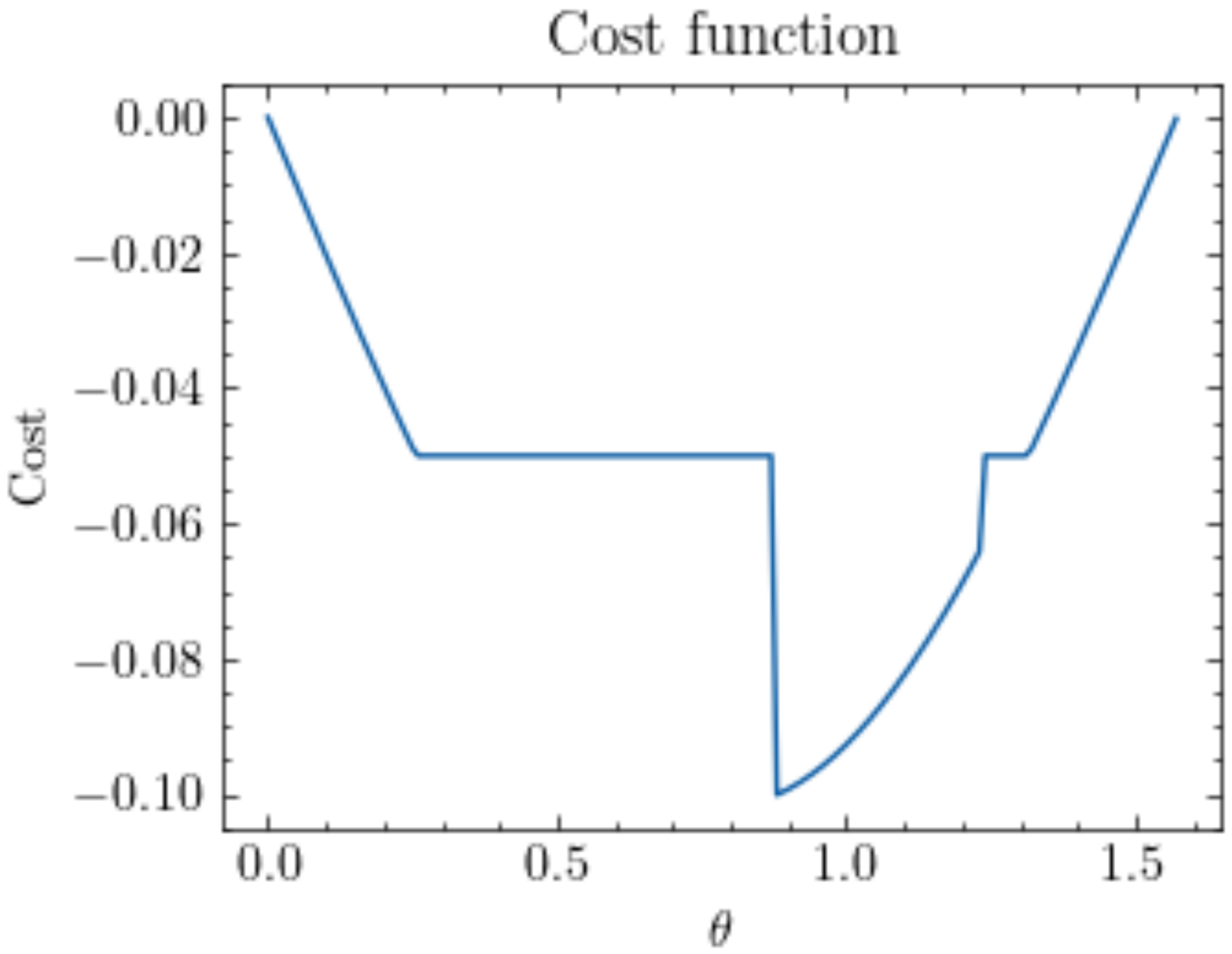
Background: Diffusion Model in Planning

Example: Throwing Ball Over a Wall with Diffusion

- Diffusion model bypasses the difficulty by transforming the optimization problem into a sampling problem.



$$p_0(Y) \propto \exp(-\frac{J(Y)}{\lambda})$$

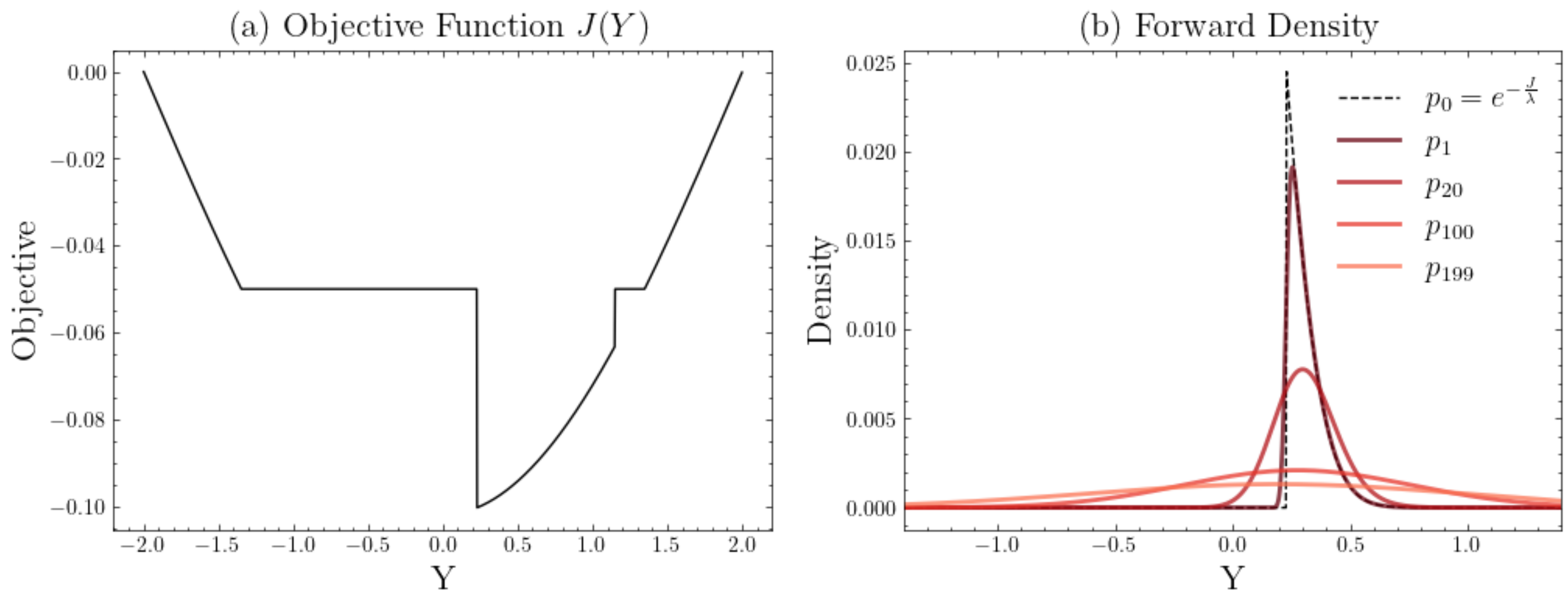


$$\min_u J(f(x_{\text{init}}, u)) = l(f(x_{\text{init}}, u), u)$$

Background: Diffusion Model in Planning

Example: Throwing Ball Over a Wall with Diffusion

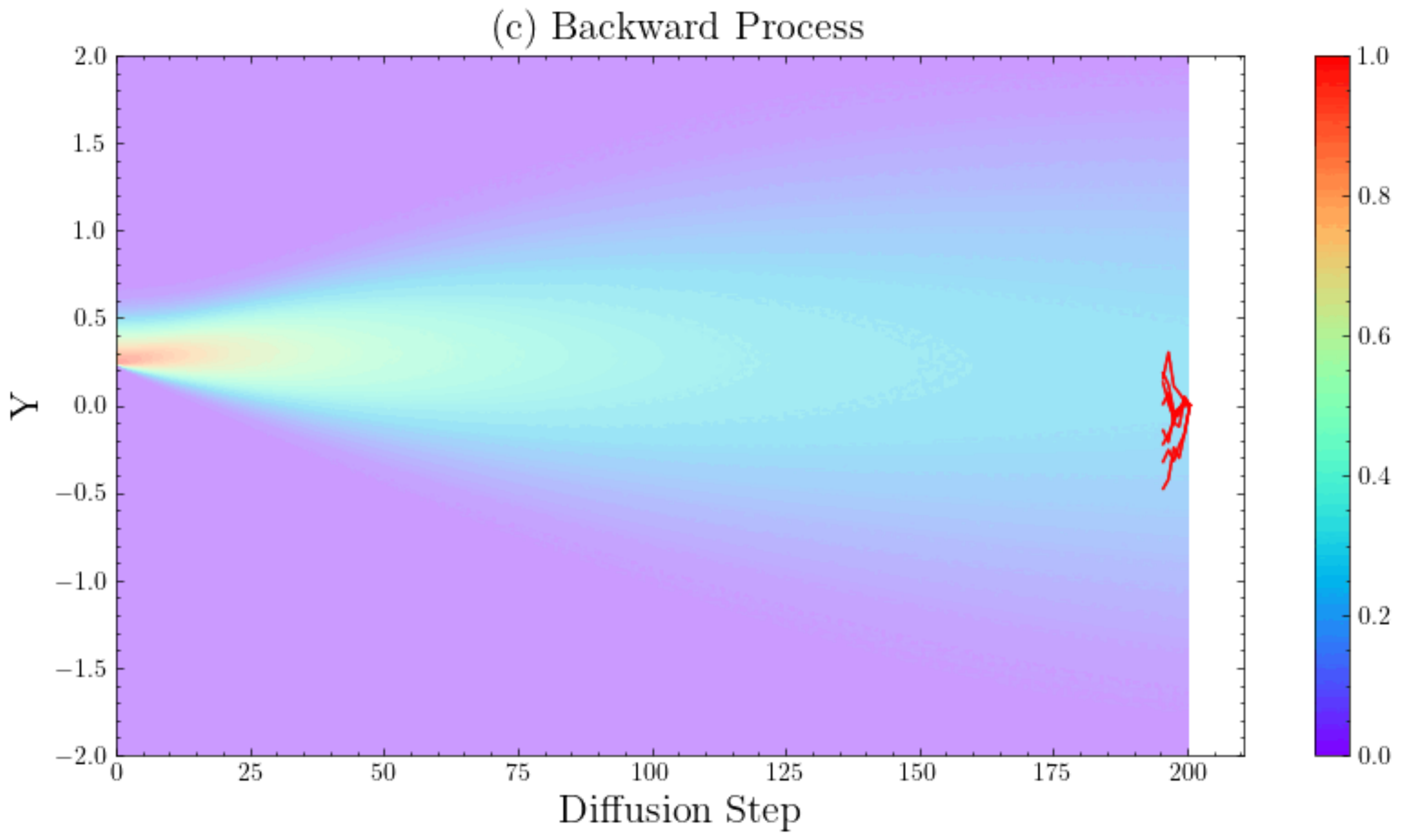
- Forward: smooth the desired distribution and make problem convex



Background: Diffusion Model in Planning

Example: Throwing Ball Over a Wall with Diffusion

- backward: start from the corrupted distribution and recover the original distribution iteratively with reverse SDE

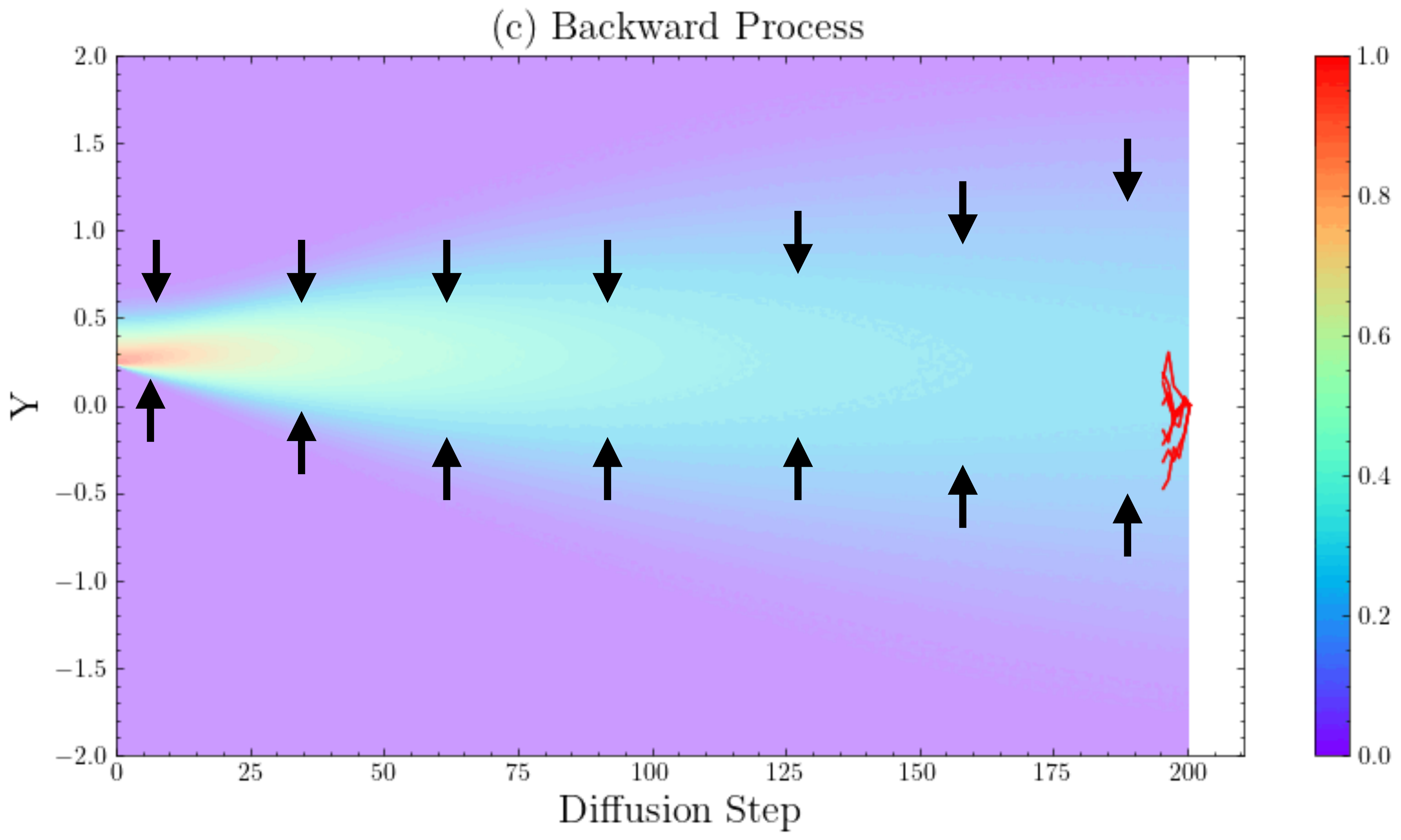


Background: Diffusion Model in Planning

Example: Throwing Ball Over a Wall with Diffusion

- backward: start from the corrupted distribution and recover the original distribution iteratively with reverse SDE

$$\nabla_{Y^{(i)}} \log p_i(Y_k^{(i)})$$



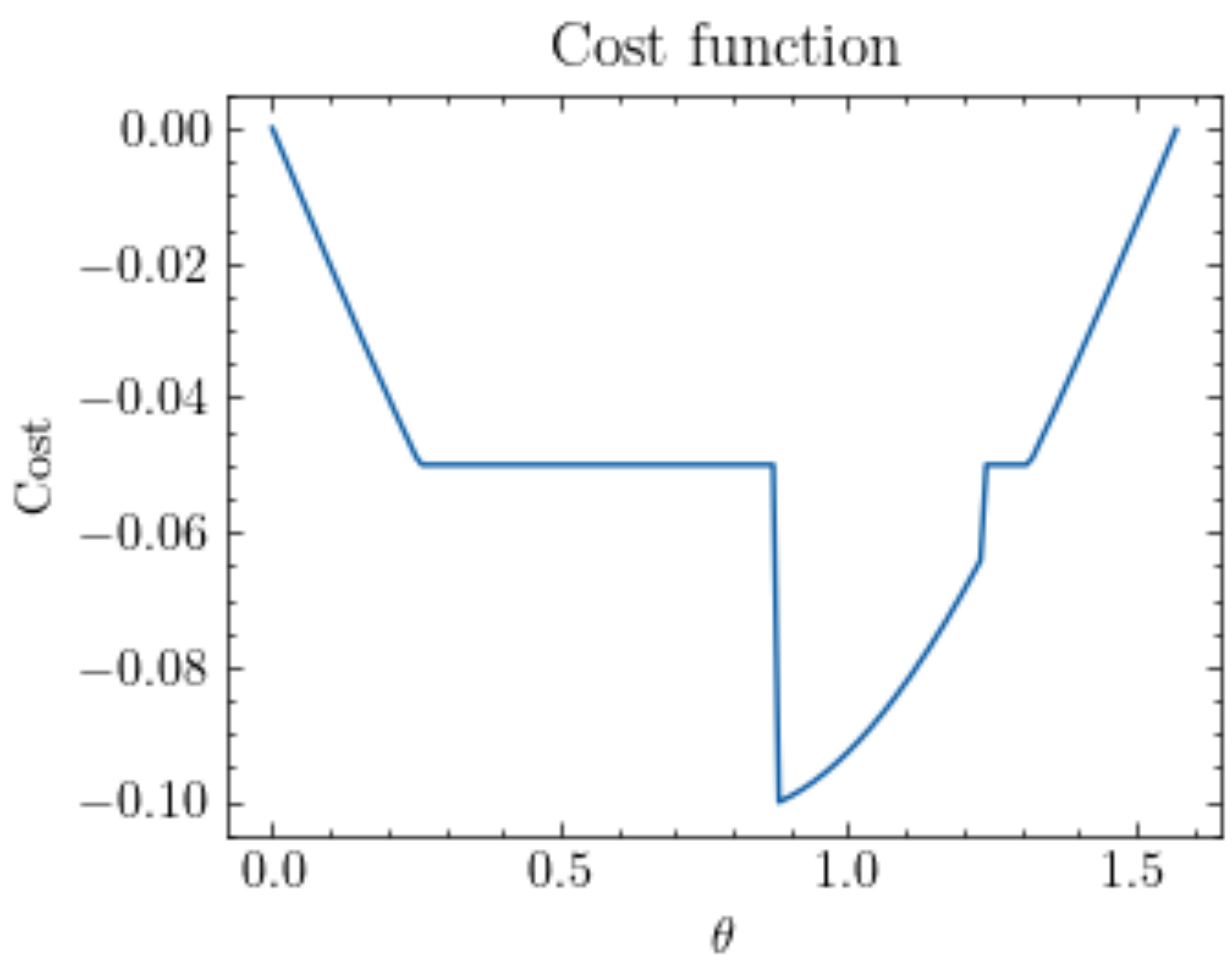
Introduction: Why is Model-based Diffusion?

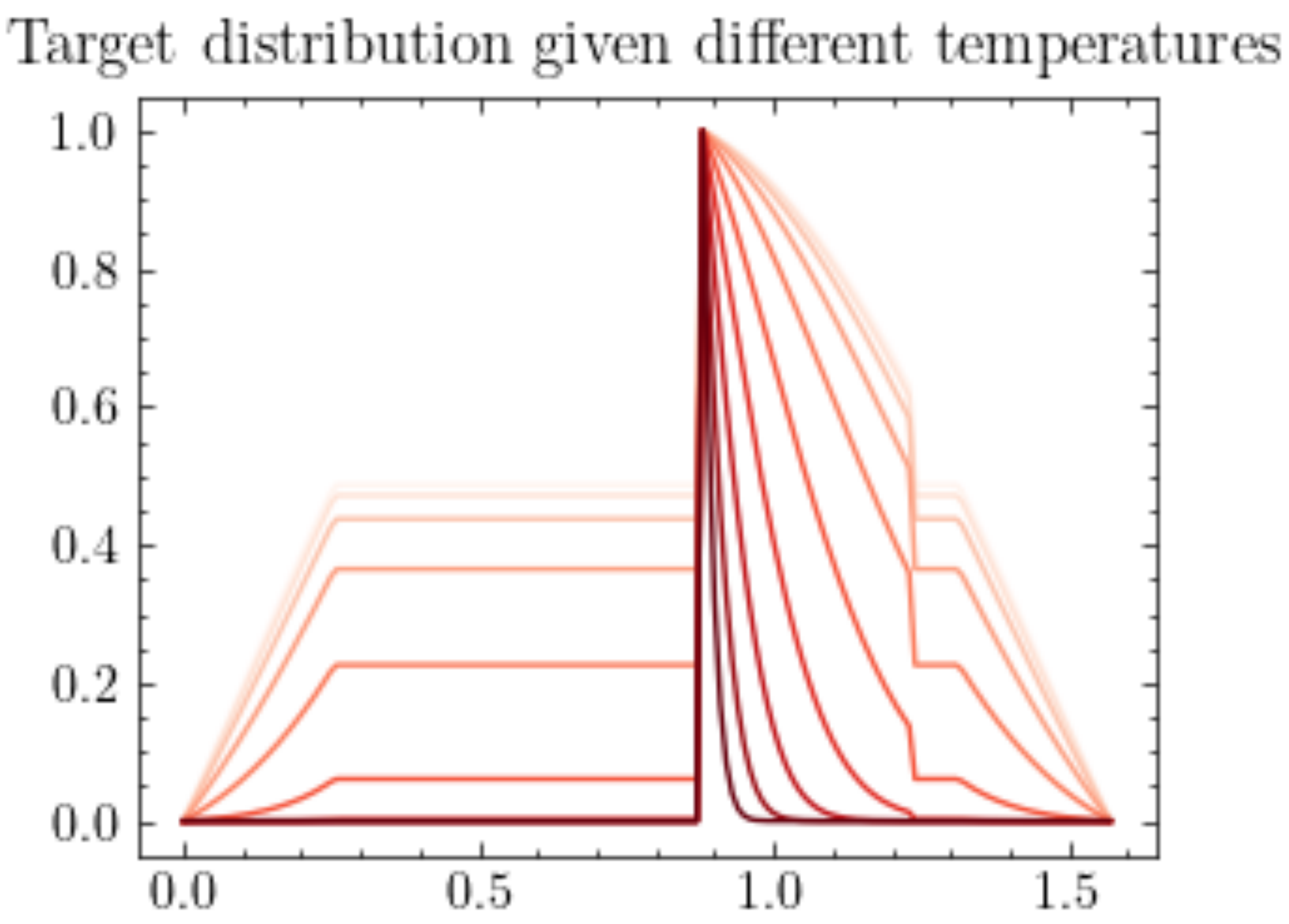
Model-based Diffusion use <u>model information</u> to generate score instead of data

Introduction: Why is Model-based Diffusion?

Model-based Diffusion Make Score Function Conditional on Model

- Model information: dynamics and cost can be evaluated at any point.





Introduction: Why is Model-based Diffusion?

Model-based Diffusion Make Score Function Conditional on Model

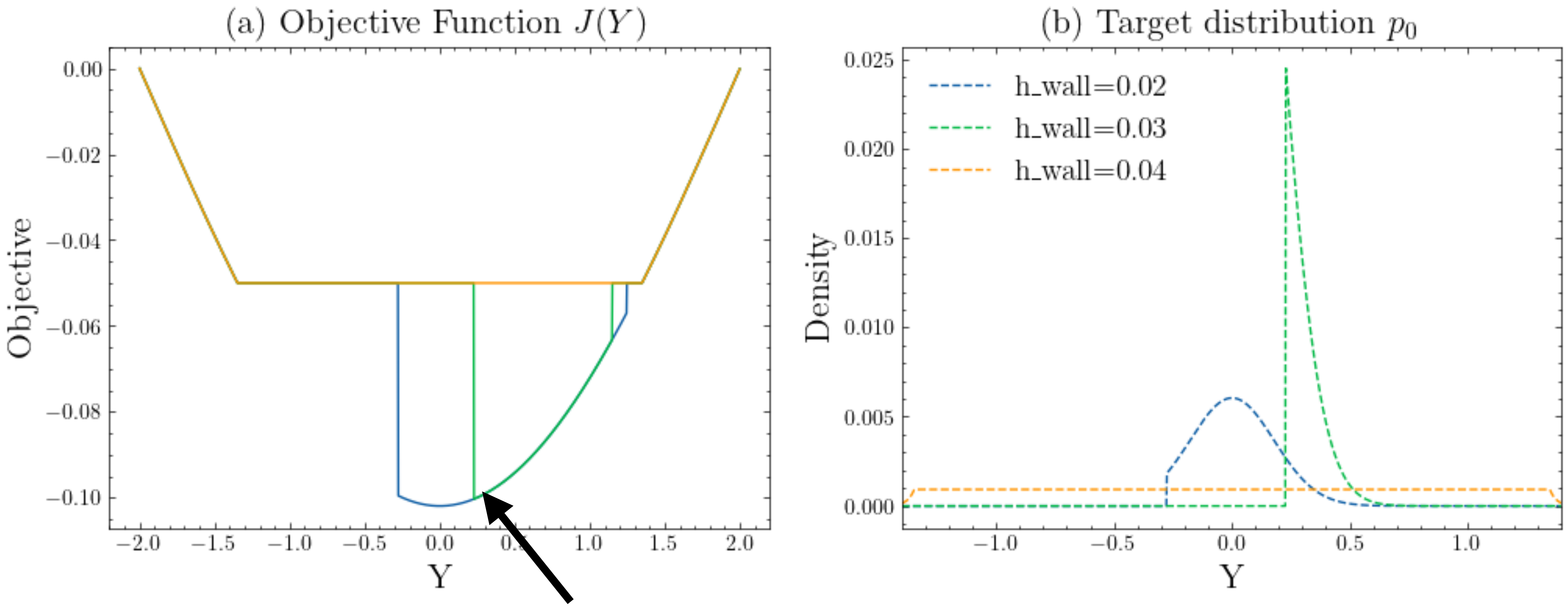
- MBD contribution: Data-free score computation + Faster backward process

Aspect	Model-Based Diffusion (MBD)	Model-Free Diffusion (MFD)
Target distribution	Known, but hard to sample	Unknown, but have data from it
Objective	Sample high-likelihood solution	Generate diverse samples

Introduction: Why is Model-based Diffusion?

Limitation of Model-Free Diffusion: not generalizable

- Model-free diffusion: cannot react to new distribution (in TO, could be dynamics/ cost changes)



Method: What is Model-based Diffusion?

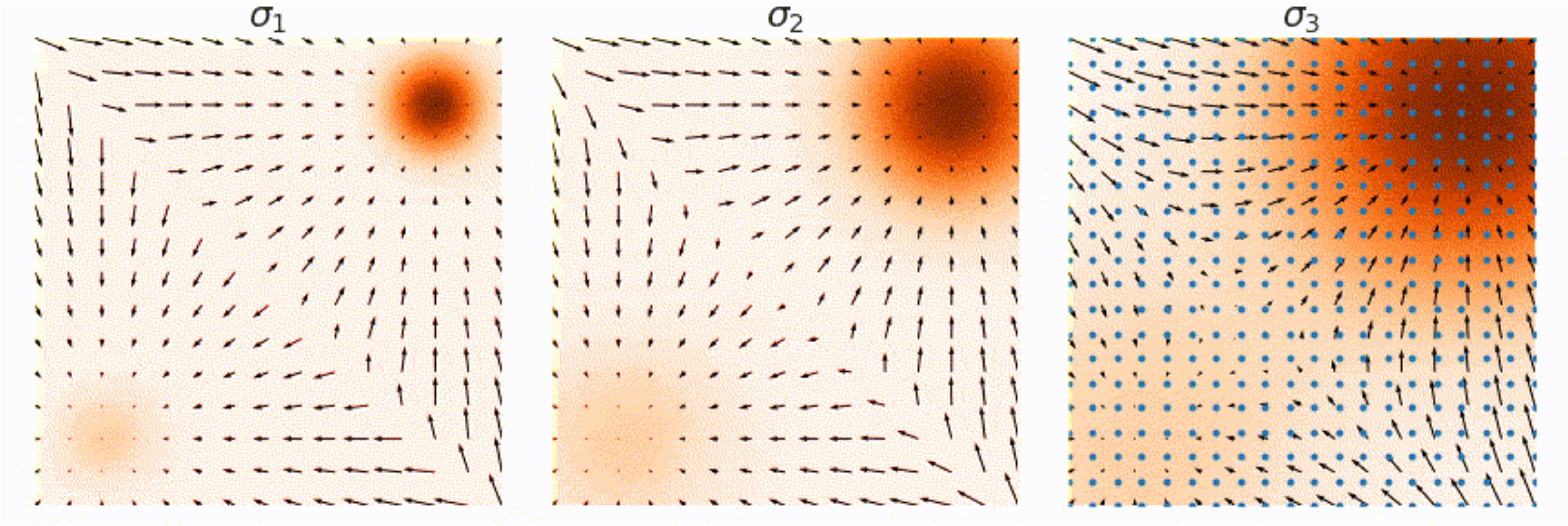
Method: What is Model-based Diffusion?

Score Function Computation with Model

- Model-based score computation: $\nabla_{Y^{(i)}} \log p_i(Y_k^{(i)})$

$$p_i = \int p_{i|0} p_0 dY^{(0)}$$

$$p_{i|0}(Y^{(i)} \mid Y^{(0)}) \propto \exp(-\frac{1}{2} \frac{\left(Y^{(i)} - \sqrt{\bar{\alpha}_i} Y^{(0)}\right)^\top \left(Y^{(i)} - \sqrt{\bar{\alpha}_i} Y^{(0)}\right)}{1 - \bar{\alpha}_i})$$



Method: What is Model-based Diffusion?

Score Function Computation with Model

- Model-based score computation: $\nabla_{Y^{(i)}} \log p_i(Y_k^{(i)})$ $\quad\quad p_i = \int p_{i|0} p_0 dY^{(0)}$

$$\nabla_{Y^{(i)}} \log p_i(Y^{(i)}) = \frac{\nabla_{Y^{(i)}} \int p_{i|0}(Y^{(i)} \mid Y^{(0)}) p_0(Y^{(0)})\, dY^{(0)}}{\int p_{i|0}(Y^{(i)} \mid Y^{(0)}) p_0(Y^{(0)})\, dY^{(0)}}$$

Gradient into int

$$= \frac{\int \nabla_{Y^{(i)}} p_{i|0}(Y^{(i)} \mid Y^{(0)}) p_0(Y^{(0)})\, dY^{(0)}}{\int p_{i|0}(Y^{(i)} \mid Y^{(0)}) p_0(Y^{(0)})\, dY^{(0)}}$$

Gaussian

$$= \frac{\int - \frac{Y^{(i)} - \sqrt{\bar{\alpha}_i} Y^{(0)}}{1 - \bar{\alpha}_i} p_{i|0}(Y^{(i)} \mid Y^{(0)}) p_0(Y^{(0)})\, dY^{(0)}}{\int p_{i|0}(Y^{(i)} \mid Y^{(0)}) p_0(Y^{(0)})\, dY^{(0)}}$$

$$= -\frac{Y^{(i)}}{1 - \bar{\alpha}_i} + \frac{\sqrt{\bar{\alpha}_i}}{1 - \bar{\alpha}_i} \frac{\int Y^{(0)} p_{i|0}(Y^{(i)} \mid Y^{(0)}) p_0(Y^{(0)})\, dY^{(0)}}{\int p_{i|0}(Y^{(i)} \mid Y^{(0)}) p_0(Y^{(0)})\, dY^{(0)}}$$

Method: What is Model-based Diffusion?

Score Function Computation with Model

- Model-based score computation with Monte Carlo Approximation: $\nabla_{Y^{(i)}} \log p_i(Y_k^{(i)})$

$$
\begin{aligned}
\nabla_{Y^{(i)}} \log p_i(Y^{(i)}) &= -\frac{Y^{(i)}}{1-\bar{\alpha}_i} + \frac{\sqrt{\bar{\alpha}_i}}{1-\bar{\alpha}_i} \frac{\int Y^{(0)} \phi_i(Y^{(0)}) p_0(Y^{(0)})\, dY^{(0)}}{\int \phi_i(Y^{(0)}) p_0(Y^{(0)})\, dY^{(0)}} \\
&\approx -\frac{Y^{(i)}}{1-\bar{\alpha}_i} + \frac{\sqrt{\bar{\alpha}_i}}{1-\bar{\alpha}_i} \underbrace{\frac{\sum_{Y^{(0)} \in \mathcal{Y}^{(i)}} Y^{(0)} p_0(Y^{(0)})}{\sum_{Y^{(0)} \in \mathcal{Y}^{(i)}} p_0(Y^{(0)})}}_{\text{Monte Carlo Approximation}} \\
&:= -\frac{Y^{(i)}}{1-\bar{\alpha}_i} + \frac{\sqrt{\bar{\alpha}_i}}{1-\bar{\alpha}_i} \bar{Y}^{(0)}(\mathcal{Y}^{(i)})
\end{aligned}
$$

Method: What is Model-based Diffusion?

Speed Up Backward Process in MBD

- Backward process: Monte Carlo Score Ascent given the objective of sampling from **high density region**

Reverse SDE

$$Y^{(i-1)} = \frac{1}{\sqrt{\alpha_i}}\left(Y^i + \frac{1-\alpha_i}{2}\nabla_{Y^{(i)}}\log p_i(Y^{(i)})\right) + \sqrt{1-\alpha_i}\mathbf{z}_i$$

MCSA (ours)

$$Y^{(i-1)} = \frac{1}{\sqrt{\alpha_i}}\left(Y^{(i)} + (1-\bar{\alpha}_i)\nabla_{Y^{(i)}}\log p_i(Y^{(i)})\right)$$

Method: What is Model-based Diffusion?

Speed Up Backward Process in MBD

- Backward process: larger step size according to the smoothness of the distribution

Reverse SDE

$$Y^{(i-1)} = \frac{1}{\sqrt{\alpha_i}}\left(Y^i + \frac{1-\alpha_i}{2}\nabla_{Y^{(i)}}\log p_i(Y^{(i)})\right) + \sqrt{1-\alpha_i}\mathbf{z}_i$$

MCSA (ours)

$$Y^{(i-1)} = \frac{1}{\sqrt{\alpha_i}}\left(Y^{(i)} + (1-\bar{\alpha}_i)\nabla_{Y^{(i)}}\log p_i(Y^{(i)})\right)$$

Method: What is Model-based Diffusion?

Speed Up Backward Process in MBD

- Backward process: no noise term in the update

Reverse SDE

$$Y^{(i-1)} = \frac{1}{\sqrt{\alpha_i}} \left(Y^i + \frac{1-\alpha_i}{2} \nabla_{Y^{(i)}} \log p_i(Y^{(i)}) \right) + \sqrt{1-\alpha_i} \mathbf{z}_i$$

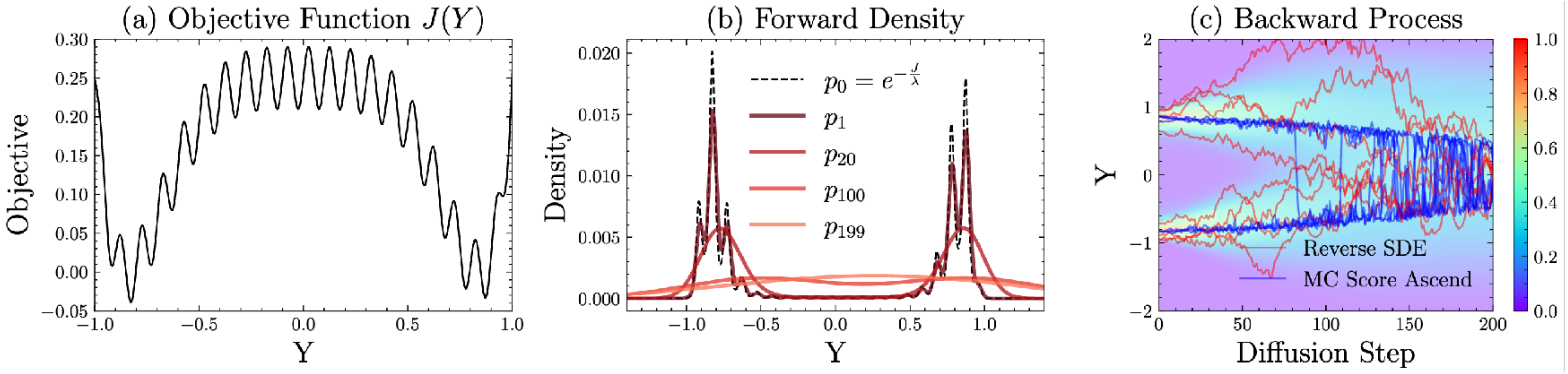
MCSA (ours)

$$Y^{(i-1)} = \frac{1}{\sqrt{\alpha_i}} \left(Y^{(i)} + (1-\bar{\alpha}_i) \nabla_{Y^{(i)}} \log p_i(Y^{(i)}) \right)$$

Method: What is Model-based Diffusion?

Speed Up Backward Process in MBD

- Backward process: converge faster while still capturing the multimodality



Method: What is Model-based Diffusion?

Model-free v.s. Model-based

Algorithm 1 Model-based Diffusion for Generic Optimization

Input: $Y^{(N)} \sim \mathcal{N}(\mathbf{0}, I)$
for $i = N$ to 1 **do**
  Sample $\mathcal{Y}^{(i)} \sim \mathcal{N}\left(\frac{Y^{(i)}}{\sqrt{\bar{\alpha}_{i-1}}}, \left(\frac{1}{\bar{\alpha}_{i-1}} - 1\right) I\right)$
  Calculate Eq. (9b) $\bar{Y}^{(0)}(\mathcal{Y}^{(i)}) = \frac{\sum_{Y^{(0)} \in \mathcal{Y}^{(i)}} Y^{(0)} p_0(Y^{(0)})}{\sum_{Y^{(0)} \in \mathcal{Y}^{(i)}} p_0(Y^{(0)})}$
  Estimate the score Eq. (9a): $\nabla_{Y^{(i)}} \log p_i(Y^{(i)}) \approx -\frac{Y^{(i)}}{1-\bar{\alpha}_i} + \frac{\sqrt{\bar{\alpha}_i}}{1-\bar{\alpha}_i} \bar{Y}^{(0)}(\mathcal{Y}^{(i)})$
  Monte Carlo score ascent Eq. (6): $Y^{(i-1)} \leftarrow \frac{1}{\sqrt{\alpha_i}} \left(Y^{(i)} + (1 - \bar{\alpha}_i) \nabla_{Y^{(i)}} \log p_i(Y^{(i)})\right)$
end for

Method: What is Model-based Diffusion?

Model-free v.s. Model-based

Aspect	Model-Based Diffusion (MBD)	Model-Free Diffusion (MFD)
Target distribution	Known, but hard to sample	Unknown, but have data from it
Objective	Sample high-likelihood solution	Generate diverse samples
Score Approximation	From model + data (optional)	From data
Backward Process	Monte Carlo Score Ascent	Reverse SDE

Method: What is Model-based Diffusion?

Solve Full Trajectory Optimization Problem with MBD

- Extend to full TrajOpt problem: considering constraints

$$
\begin{aligned}
\min_{x_{1:T}, u_{1:T}} \quad & J(x_{1:T}; u_{1:T}) = l_T(x_T) + \sum_{t=0}^{T-1} l_t(x_t, u_t) \\
\text{s.t.} \quad & x_0 = x_{\text{init}} \\
& x_{t+1} = f_t(x_t, u_t), \quad \forall t = 0,1,\ldots,T-1, \\
& g_t(x_t, u_t) \leq 0, \quad \forall t = 0,1,\ldots,T-1.
\end{aligned}
$$

Method: What is Model-based Diffusion?

Solve Full Trajectory Optimization Problem with MBD

- Extend to full TrajOpt problem: considering constraints

$$\min_{x_{1:T},u_{1:T}} J(x_{1:T}; u_{1:T}) = l_T(x_T) + \sum_{t=0}^{T-1} l_t(x_t, u_t)$$

$$\text{s.t.} \quad x_0 = x_{\text{init}}$$

$$x_{t+1} = f_t(x_t, u_t), \quad \forall t = 0,1,\ldots,T-1,$$

$$g_t(x_t, u_t) \le 0, \quad \forall t = 0,1,\ldots,T-1.$$

$$p_d(Y) \propto \prod_{t=1}^{T} \mathbf{1}(x_t = f_{t-1}(x_{t-1}, u_{t-1}))$$

$$p_J(Y) \propto \exp(-\frac{J(Y)}{\lambda})$$

$$p_g(Y) \propto \prod_{t=1}^{T} \mathbf{1}(g_t(x_t, u_t) \le 0)$$

$$p_0(Y) \propto p_d(Y) p_J(Y) p_g(Y)$$

Method: What is Model-based Diffusion?

Solve Full Trajectory Optimization Problem with MBD

- Sampling only from feasible trajectory space

$$
\begin{aligned}
\nabla_{Y_i} \log p_i(Y_i) &= -\frac{Y_i}{1-\bar{\alpha}_i} + \frac{\sqrt{\bar{\alpha}_i}}{1-\bar{\alpha}_i} \frac{\int Y_0 \phi_i(Y_0) p_d(Y_0) p_g(Y_0) p_J(Y_0)\, dY_0}{\int \phi_i(Y_0) p_d(Y_0) p_g(Y_0) p_J(Y_0)\, dY_0} \\
&\approx -\frac{Y_i}{1-\bar{\alpha}_i} + \frac{\sqrt{\bar{\alpha}_i}}{1-\bar{\alpha}_i} \frac{\sum_{Y_0 \in \mathcal{Y}_d^{(i)}} Y_0 p_J(Y_0) p_g(Y_0)}{\sum_{Y_0 \in \mathcal{Y}_d^{(i)}} p_J(Y_0) p_g(Y_0)} \\
&= -\frac{Y_i}{1-\bar{\alpha}_i} + \frac{\sqrt{\bar{\alpha}_i}}{1-\bar{\alpha}_i} \bar{Y}^{(0)},
\end{aligned}
$$

$$
\text{where} \quad \bar{Y}^{(0)} = \frac{\sum_{Y_0 \in \mathcal{Y}_d^{(i)}} Y_0 w(Y_0)}{\sum_{Y_0 \in \mathcal{Y}_d^{(i)}} w(Y_0)}, \quad w(Y_0) = p_J(Y_0) p_g(Y_0)
$$

Method: What is Model-based Diffusion?

Solve Full Trajectory Optimization Problem with MBD

Algorithm 2 Model-based Diffusion for Trajectory Optimization

Input: $Y^{(N)} \sim \mathcal{N}(\mathbf{0}, I)$
for $i = N$ to 1 **do**
  Sample $\mathcal{Y}^{(i)} \sim \mathcal{N}(\frac{Y^{(i)}}{\sqrt{\bar{\alpha}_{i-1}}}, (\frac{1}{\bar{\alpha}_{i-1}} - 1)I)$
  Get dynamically feasible samples: $\mathcal{Y}_d^{(i)} \leftarrow \text{rollout}(\mathcal{Y}^{(i)})$
  Calculate $\bar{Y}^{(0)}$ with Eq. (10d) (model only) or Eq. (13) (model + demonstration)
  Estimate the score Eq. (10c): $\nabla_{Y^{(i)}} \log p_i(Y^{(i)}) \approx -\frac{Y^{(i)}}{1-\bar{\alpha}_i} + \frac{\sqrt{\bar{\alpha}_i}}{1-\bar{\alpha}_i}\bar{Y}^{(0)}$
  Monte Carlo score ascent Eq. (6): $Y^{(i-1)} \leftarrow \frac{1}{\sqrt{\alpha_i}}\left(Y^{(i)} + (1-\bar{\alpha}_i)\nabla_{Y^{(i)}} \log p_i(Y^{(i)})\right)$
end for

Method: What is Model-based Diffusion?

Augment MBD with Diverse Data

- Data in MBD: regularize solution or guide to high-likelihood solution

$$p_0'(Y^{(0)}) \propto (1-\eta)p_d(Y^{(0)})p_J(Y^{(0)})p_g(Y^{(0)}) + \eta p_{\text{demo}}(Y^{(0)})p_J(Y_{\text{demo}})p_g(Y_{\text{demo}})$$

Method: What is Model-based Diffusion?

Augment MBD with Diverse Data

- Data in MBD: regularize solution or guide to high-likelihood solution

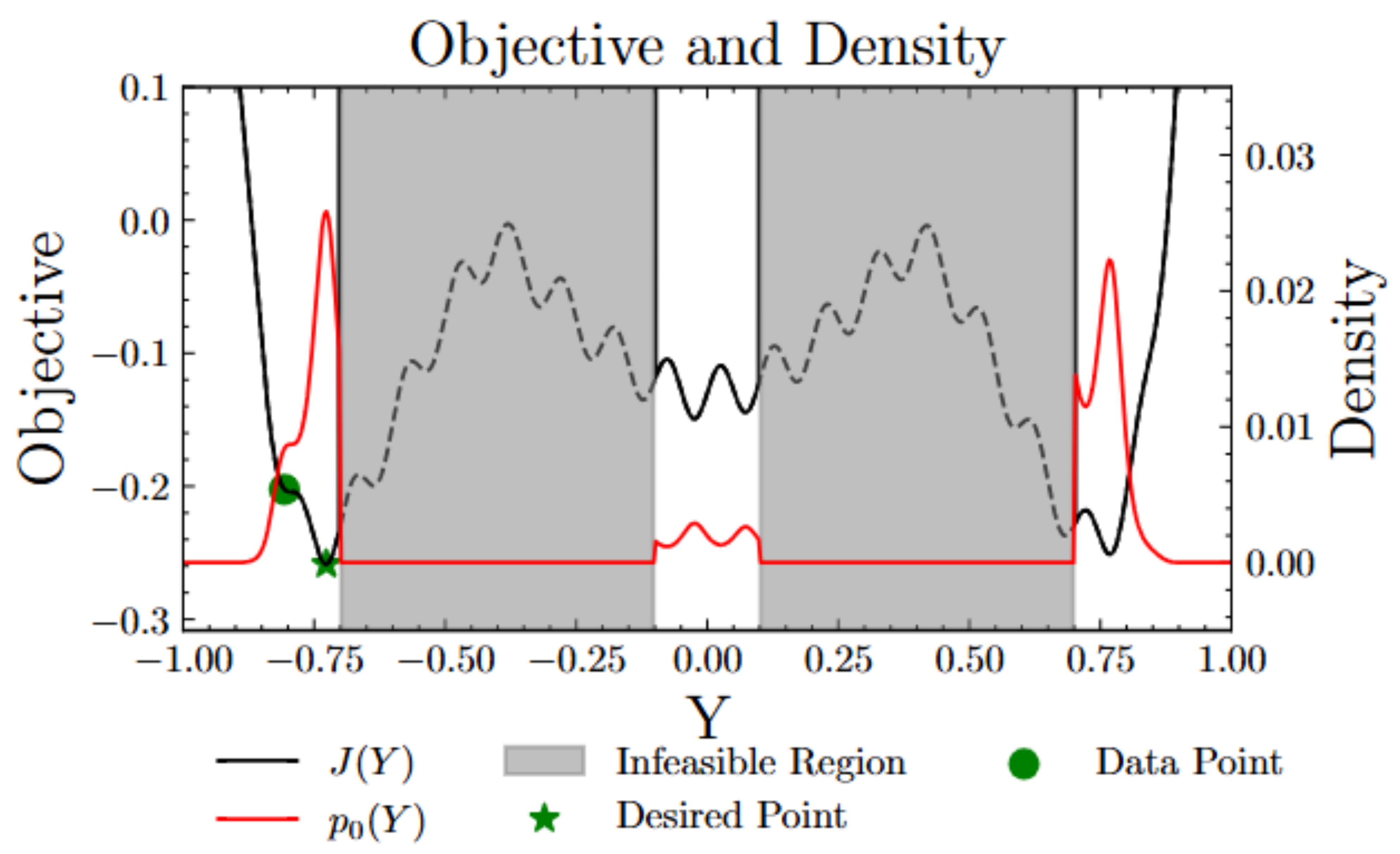
$$p_0'(Y^{(0)}) \propto (1-\eta)p_d(Y^{(0)})p_J(Y^{(0)})p_g(Y^{(0)}) + \eta p_{\text{demo}}(Y^{(0)})p_J(Y_{\text{demo}})p_g(Y_{\text{demo}})$$

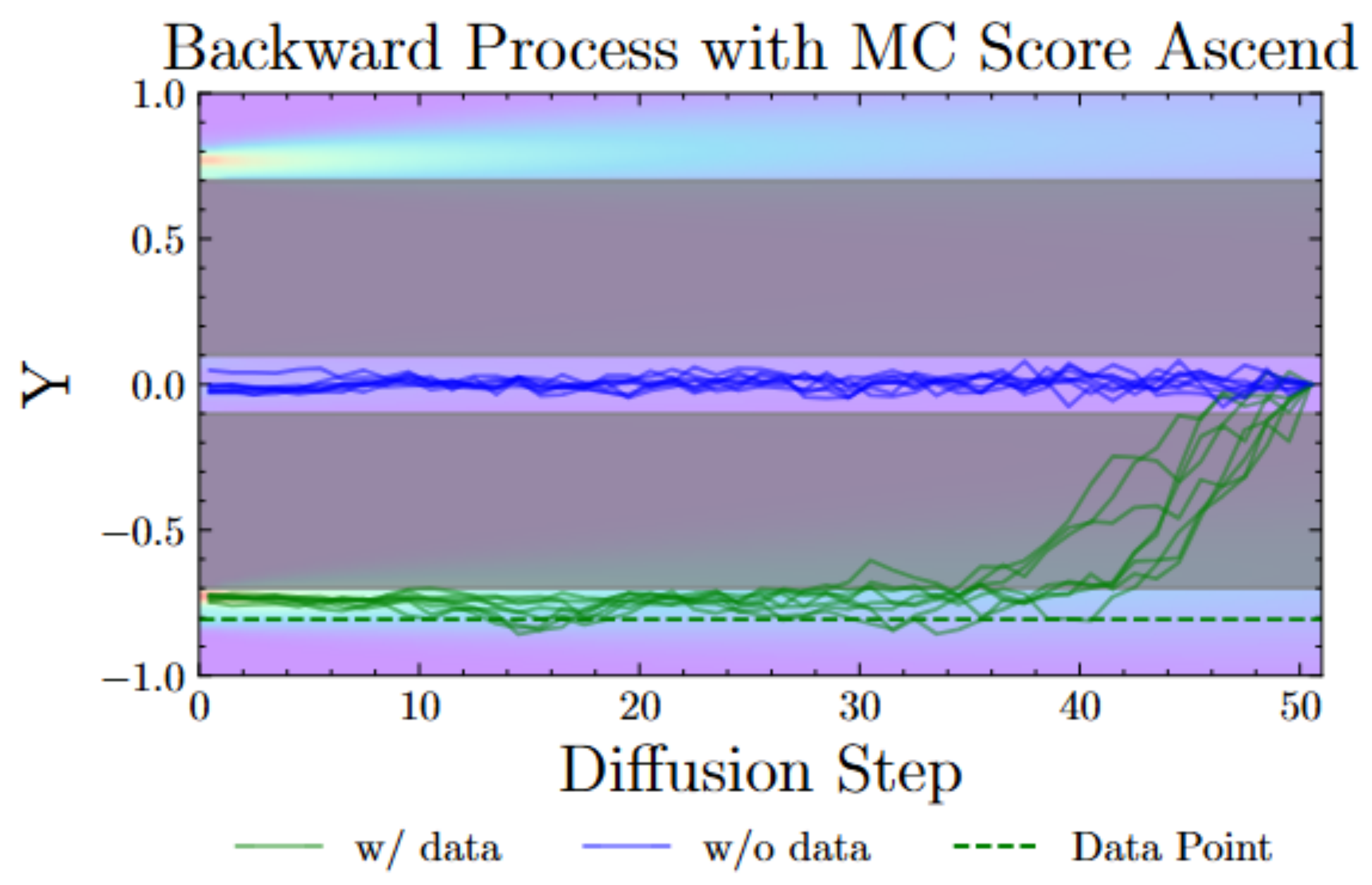
$$\eta = \begin{cases} 1 & p_d(Y^0)p_J(Y^0)p_g(Y^0) < p_{\text{demo}}(Y^0)p_J(Y_{\text{demo}})p_g(Y_{\text{demo}}) \\ 0 & p_d(Y^0)p_J(Y^0)p_g(Y^0) \geq p_{\text{demo}}(Y^0)p_J(Y_{\text{demo}})p_g(Y_{\text{demo}}) \end{cases}.$$

Method: What is Model-based Diffusion?

Augment MBD with Diverse Data

- Data in MBD: regularize solution or guide to high-likelihood solution



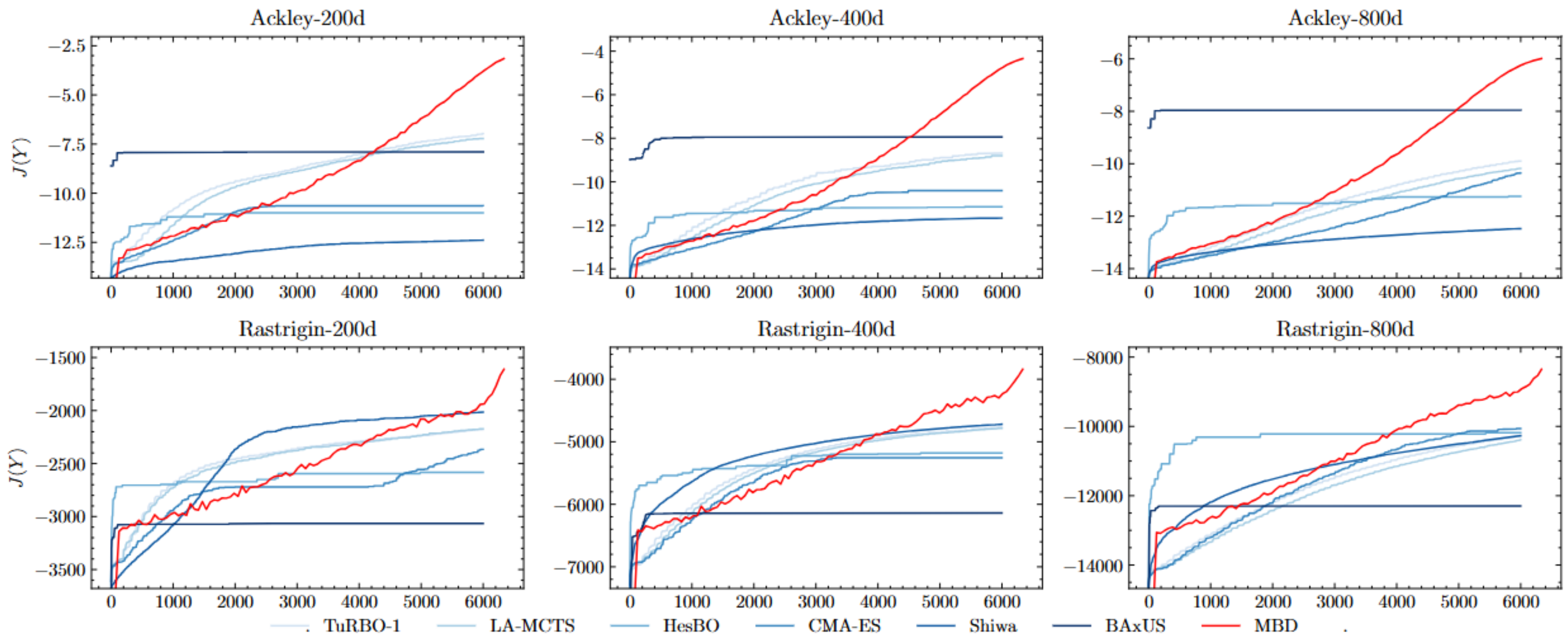


Results: How does MBD Perform?

Results: How does MBD Perform?

MBD for Zeroth Order Optimization

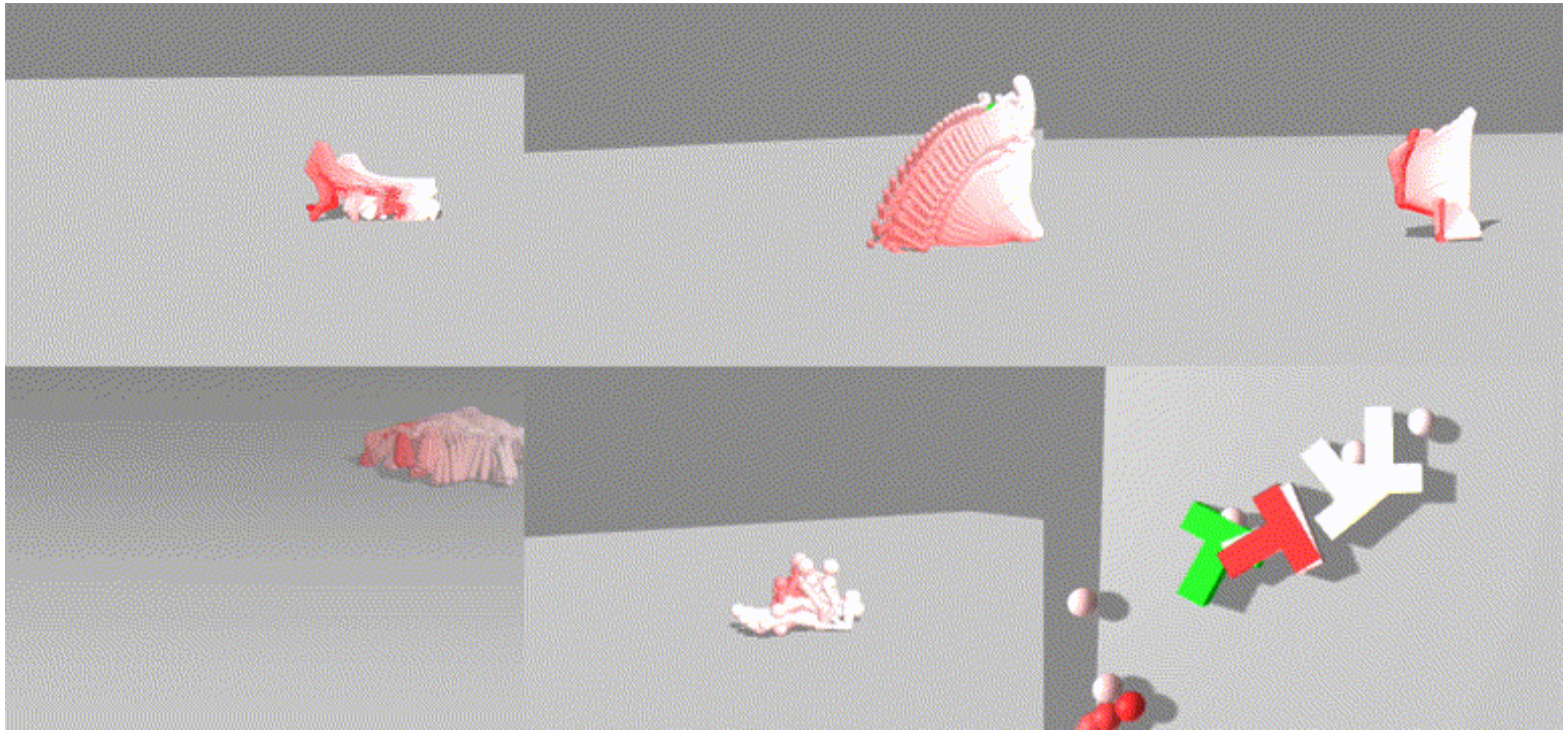
- Outperforms other Gaussian Process-based Bayesian Optimization methods by 23%.



Results: How does MBD Perform?

MBD for Trajectory Optimization

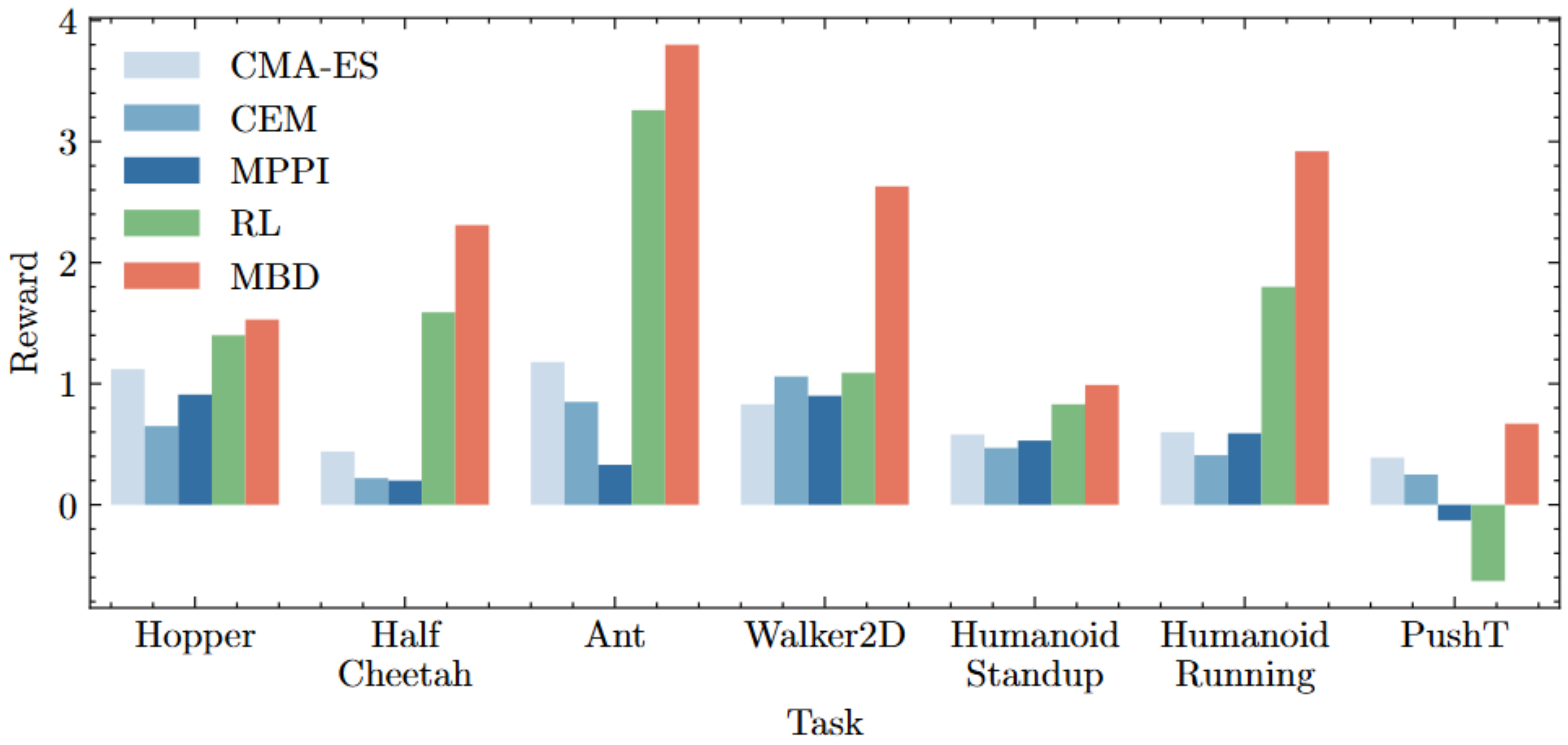
- Tasks: contact-rich locomotion and manipulation tasks



Results: How does MBD Perform?

MBD for Trajectory Optimization

- MBD outperforms PPO by 59%*

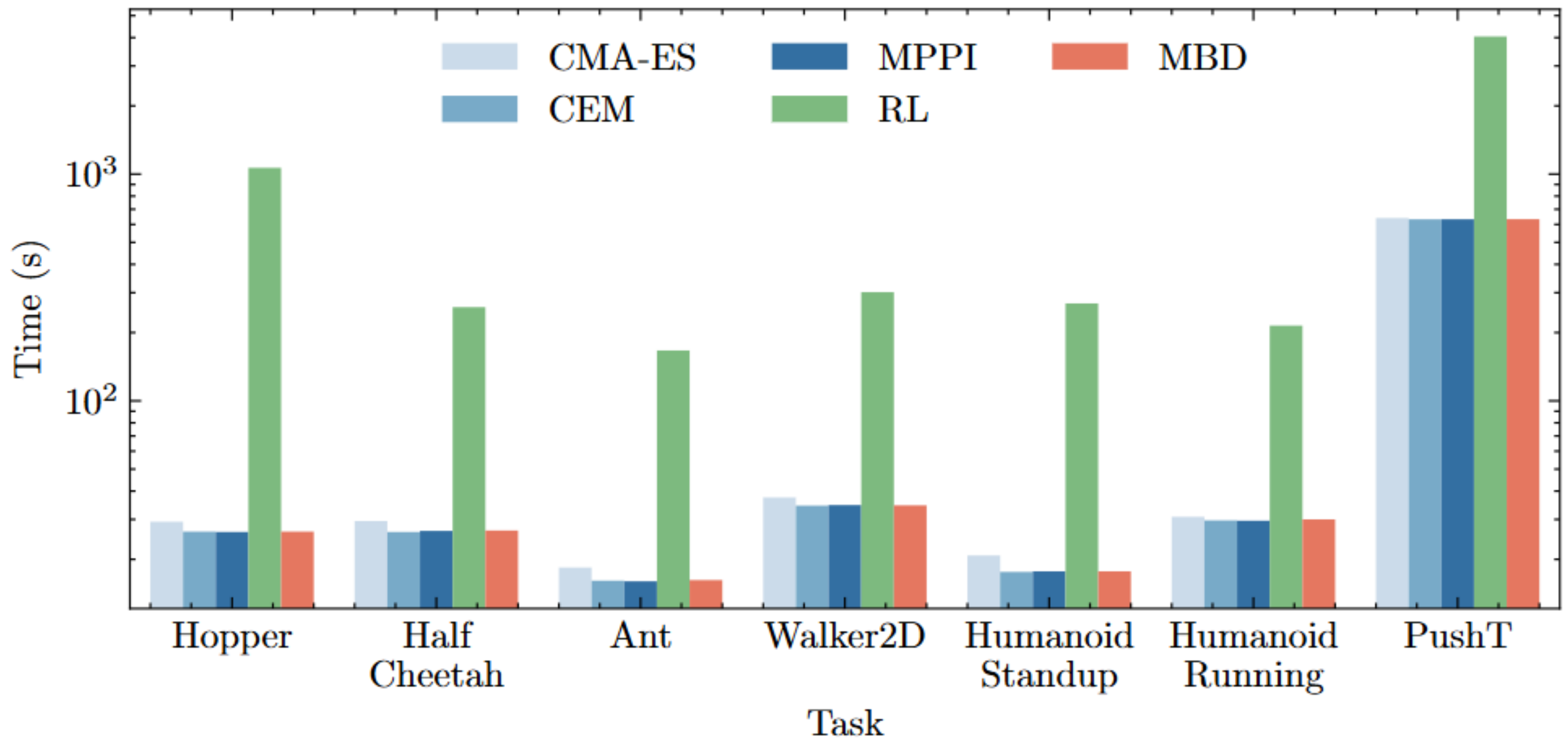


* MBD only plan one open loop trajectory while PPO learns a feedback policy

Results: How does MBD Perform?

MBD for Trajectory Optimization

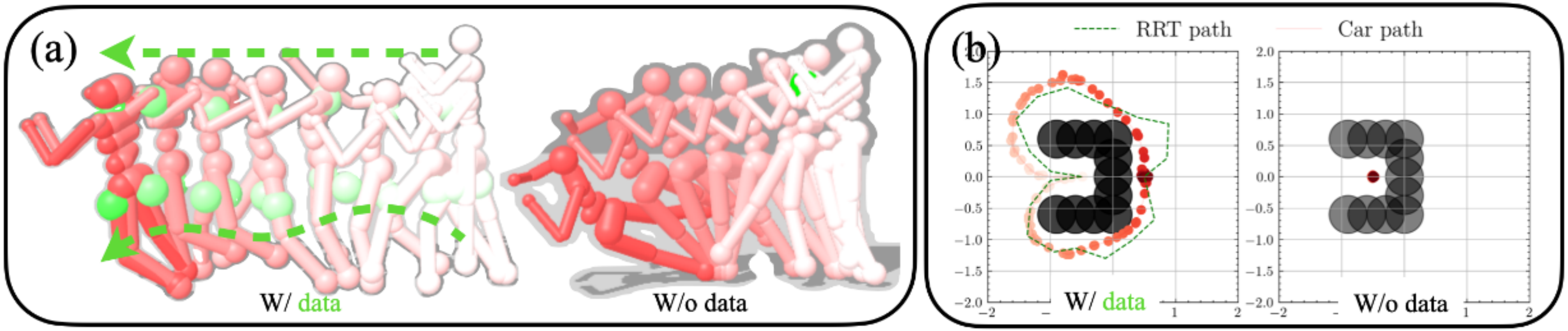
- MBD only requires 10% computational time



Results: How does MBD Perform?

MBD with Diverse Data

- MBD helps regularize humanoid motion and steering to feasible solutions



Takeaway

- Optimization can be **transformed into a sampling problem**, wheregenerative model becomes a powerful tool.
- MBD achieve generalizable trajectory optimization and fast convergence by leveraging **model information**
- MBD can use **diverse quality data** more effectively to generate high-quality trajectories.