VFScale: Intrinsic reasoning through verifier-free test-time scalable diffusion model

Tao Zhang*, Jia-Shu Pan*, Ruiqi Feng, Tailin Wu†

Zhejiang University · Westlake University

Speaker: Jia-shu Pan · ICLR 2026 · March 2026

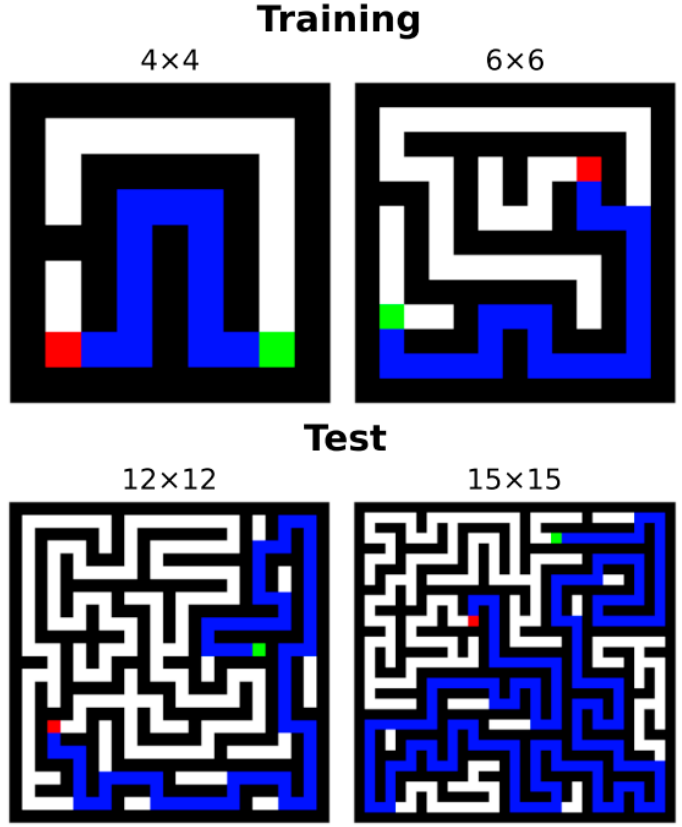
Core idea:

Train the diffusion model's own energy to act as a verifier, then scale reasoning with hybrid search.

No external verifier

Diffusion model

Test-time scaling



train up to 6×6,
solve 15×15 at test time

Outline

From formulation to scalable OOD reasoning

1 **Motivation**

Why use test-time scaling of diffusion model to do intrinsic reasoning?

2 **Setup**

Condition on a problem c, generate a solution x

3 **VFScale**

Train better energy, then search better

4 **Results**

Bottleneck → training effect → test-time scalability

5 **Take-away**

Main message and contact

The key story is: formalize reasoning as conditional diffusion, then show how VFScale unlocks scalable inference.

Motivation

The story starts from the same System-2 question that motivated LLM test-time scaling.

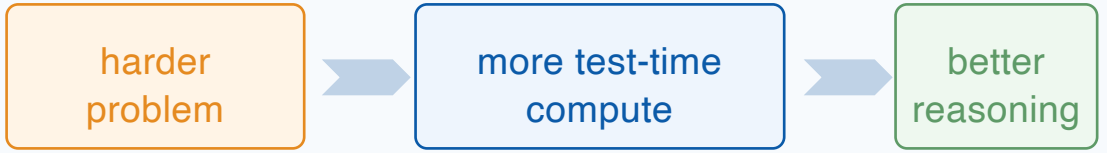
Why care?

- **System 2 thinking** allocates more computation to harder problems without external feedbacks.
- **LLMs** already benefit from extra test-time compute through longer chain-of-thought.
- **Diffusion models** also solve problems iteratively: they refine candidates by denoising.

So diffusion reasoning looks like a natural place to ask whether extra compute can buy better reasoning.

Central question

Can diffusion reasoning also scale with extra test-time compute on hard tasks?

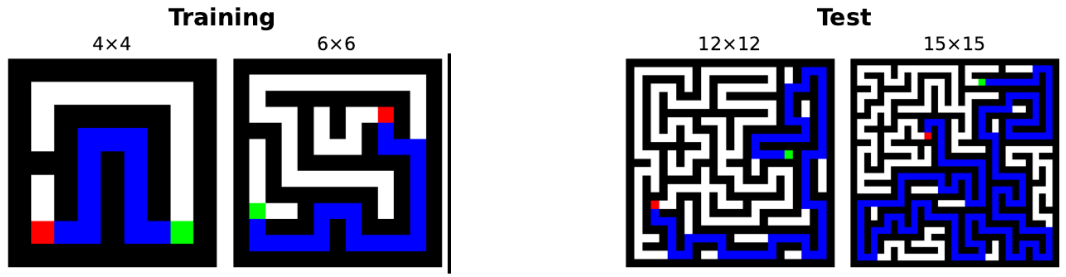


That is exactly the System-2 intuition we want to bring to diffusion reasoning.

Motivation

Can we design a verifier-free test-time scalable diffusion model that achieves scalable intrinsic reasoning on harder task?

Small-train, large-test, no external verifier



Train on small mazes, then test on much larger ones.
The denoising mechanism stays the same, but the difficulty shift is large.

The paper asks for intrinsic reasoning: no extra learned reward model, no external verifier at test time.

Maze size generalization

6×6 → 15×15

Success rate: 100% → 6.3%

Original model, naïve inference

Sudoku conditioning

33 givens → 21 givens

Success rate: 32.0% → 0.0%

Original model, naïve inference

Naïve model fails on harder task.

Diffusion model for reasoning

Forward process: corrupt a clean solution into a noisy state.

$$x_t = \sqrt{\bar{\alpha}_t}\, x_0 + \sqrt{1-\bar{\alpha}_t}\, \epsilon$$

Reverse process: iteratively denoise to recover a candidate solution.

$$\boldsymbol{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}} \frac{\boldsymbol{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_\theta\left(\boldsymbol{x}_t, t\right)}{\sqrt{\bar{\alpha}_t}} + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\boldsymbol{\epsilon}_\theta\left(\boldsymbol{x}_t, t\right) + \sigma_t \boldsymbol{\epsilon}_t, \;\; \boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}).$$

Energy-based view: the score field is the negative gradient of an energy function.

$$\epsilon_\theta(x_t, t) = -\nabla_{x_t} E_\theta(x_t, t)$$

If lower energy really means better candidate quality, then the model's own energy could serve as an intrinsic verifier.

Problem setup

Given a problem c, a conditional diffusion model generates a solution x by iterative denoising.

Conditional diffusion for reasoning

problem c
maze / sudoku

conditional diffusion
learn $\mathbf{E}_{\theta}(c, x_t, t)$
denoise $x_t \rightarrow x_0$

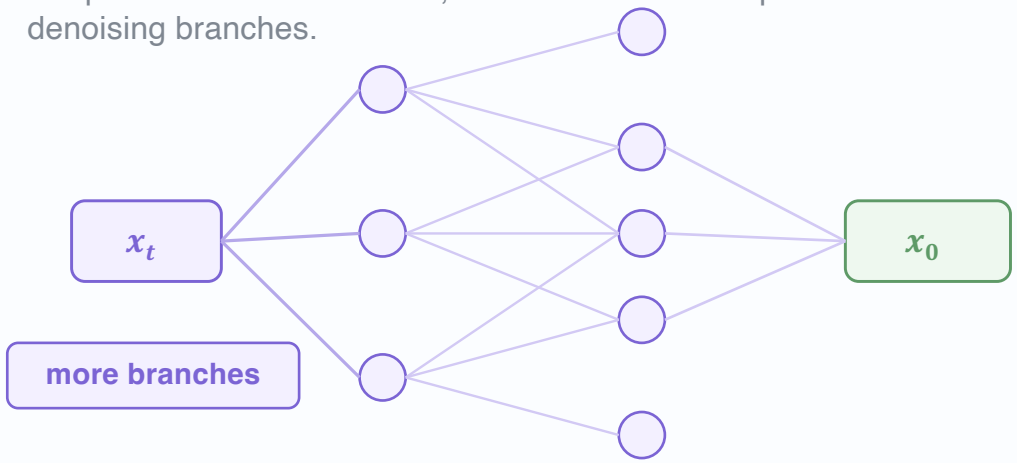
Solution x_0
candidate answer

model $p_{\theta}(x \mid c)$

Each instance is a pair (c, x): condition on the problem c and generate the target solution x.

Test-time scaling knob

Keep the trained model fixed, but increase the sampled denoising branches.



scale by #samples / #branches N

Harder reasoning comes from branching the denoising trajectories, not from adding an external verifier.

Challenges

Once we move to verifier-free number-of-sample scaling with energy-based diffusion model, two bottlenecks appear immediately.

1. The verifier is unreliable

- Number-of-sample scaling only works if the verifier can reliably rank candidate quality.
- But the original learned energy often misranks candidates, so the search keeps the wrong ones.

Original + BoN (energy-guided)

10.9% / 28.1%

Maze / Sudoku

Even GT-guided BoN

17.2% / 29.7%

Maze / Sudoku

2. The search spends budget poorly

- Best-of-N is broad but shallow: it wastes budget on early noisy states.
- Pure MCTS is focused but brittle: it can prune away good branches before energy becomes reliable.
- So the search policy should change with the denoising stage.

VFScale addresses both:
improve the energy landscape during training,
then use a smarter search policy at test time.

Method overview

VFScale has exactly two moving parts: better energy during training, and better budget allocation during search.

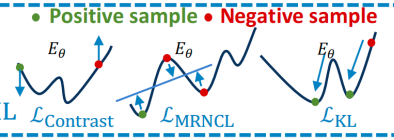
Training: make energy useful for selection

Ours

Training

Denoising loss + Contrastive loss + LRNCL loss + KL loss

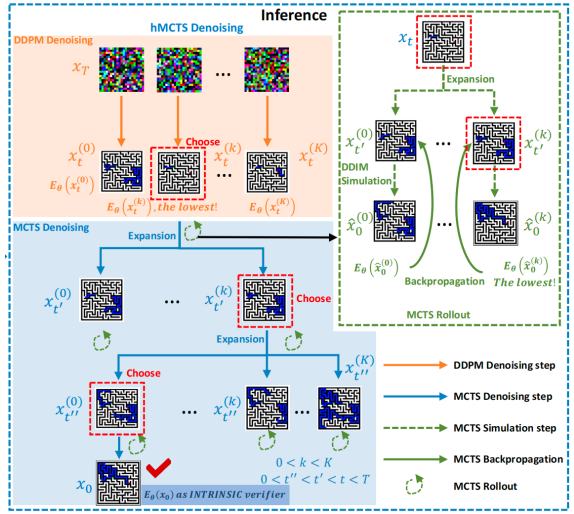
$$\mathcal{L} = \mathcal{L}_{\text{MSE}} + \mathcal{L}_{\text{Contrast}} + \mathcal{L}_{\text{MRNCL}} + \mathcal{L}_{\text{KL}}$$



- MRNCL explicitly orders candidates by corruption level, so energy becomes a quality signal.
- KL regularization smooths the landscape, making search easier instead of more brittle.

Outcome: the model's own energy becomes a usable intrinsic verifier to select better candidate branch.

Inference: spend budget where it matters



hMCTS=BoN + MCTS

- BoN keeps many hypotheses alive in noisy early stages.
- MCTS dives deeper once energy becomes more reliable later in denoising.

Outcome: extra compute is translated into real scaling gains instead of wasted samples.

Methods - Training

The training target is not just a sharp one-step denoiser; it is an energy landscape that search can trust later at test time.

Loss design

$$\mathcal{L} = \mathcal{L}_{\text{MSE}} + \mathcal{L}_{\text{contrast}} + \mathcal{L}_{\text{MRNCL}} + \mathcal{L}_{\text{KL}}$$

MRNCL samples two negatives at different corruption levels and enforces a monotonic energy order.

KL regularizes the denoising distribution so the landscape becomes smoother and easier to search.

Together, the training objective optimizes searchability, not only one-shot reconstruction.

Desired ordering

good sample low E → mid corruption mid E → hard corruption high E

The better the candidate, the lower the energy.

That is exactly the alignment a search algorithm needs during test-time scaling.

This is why VFScale sometimes gives up a little N = 1 sharpness in exchange for much better scalability.

Methods - Inference

The search policy should match the denoising stage: explore when uncertainty is high, exploit when the landscape is already informative.

Hybrid schedule across denoising steps

high noise → low noise

BoN phase

Best-of-N keeps many hypotheses alive when uncertainty is highest.

MCTS phase

MCTS explores local alternatives more deeply once coarse pruning becomes reliable.

switch-over by denoising step →

hMCTS matches the search policy to the denoising stage instead of using one policy everywhere.

Why a hybrid schedule?

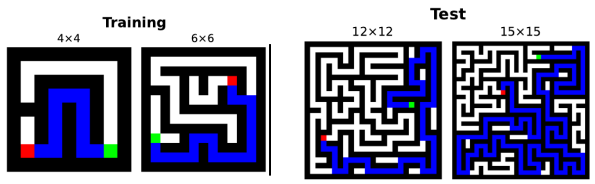
- At high noise, many branches are still plausible, so pruning too early is dangerous.
- At low noise, deeper local exploration becomes worthwhile because energy is more informative.

This is a better exploration–exploitation balance than BoN or MCTS alone.

Experiments: Test-time scaling bottleneck of energy-based diffusion models

Before VFScale, diffusion reasoning fails for two reasons: OOD generalization collapses, and more samples do not fix the ranking/search problem.

OOD collapse



Maze size: 6×6 → 15×15

100% → 6.3%

Sudoku given digits: 33 givens → 21 givens

32.0% → 0.0%

Original + BoN (energy-guided)

10.9%

Maze @ N = 161

28.1%

Sudoku @ N = 321

More samples are present, but the model still cannot rank them well enough.

Even GT-guided BoN

17.2%

Maze @ N = 161

29.7%

Sudoku @ N = 321

So even a stronger verifier is not enough if the search policy remains too weak.

Experiments: the effect of VFScale training methods

Training methods of VFScale reshapes the energy landscape enough to make search reliable.

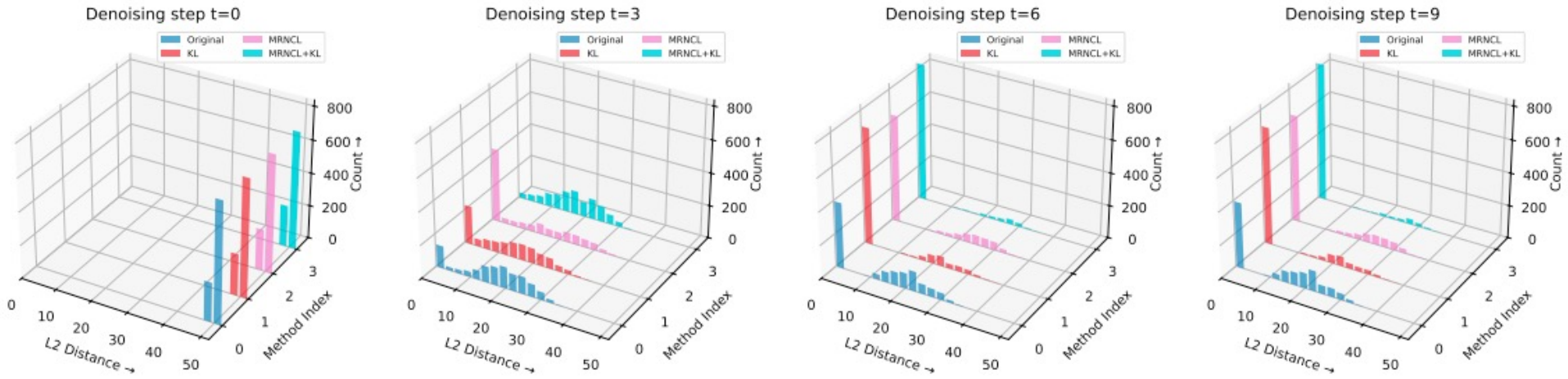


Figure 4: Comparison of the L2 distances between the solutions obtained by different training methods and the ground truth at various denoising steps.

global alignment

73% → 84%

Performance-energy-consistency on Maze (MRNCL only)

same BoN scaling on Maze

10.9% → 70.3%

Original → VFScale training

same BoN scaling on Sudoku

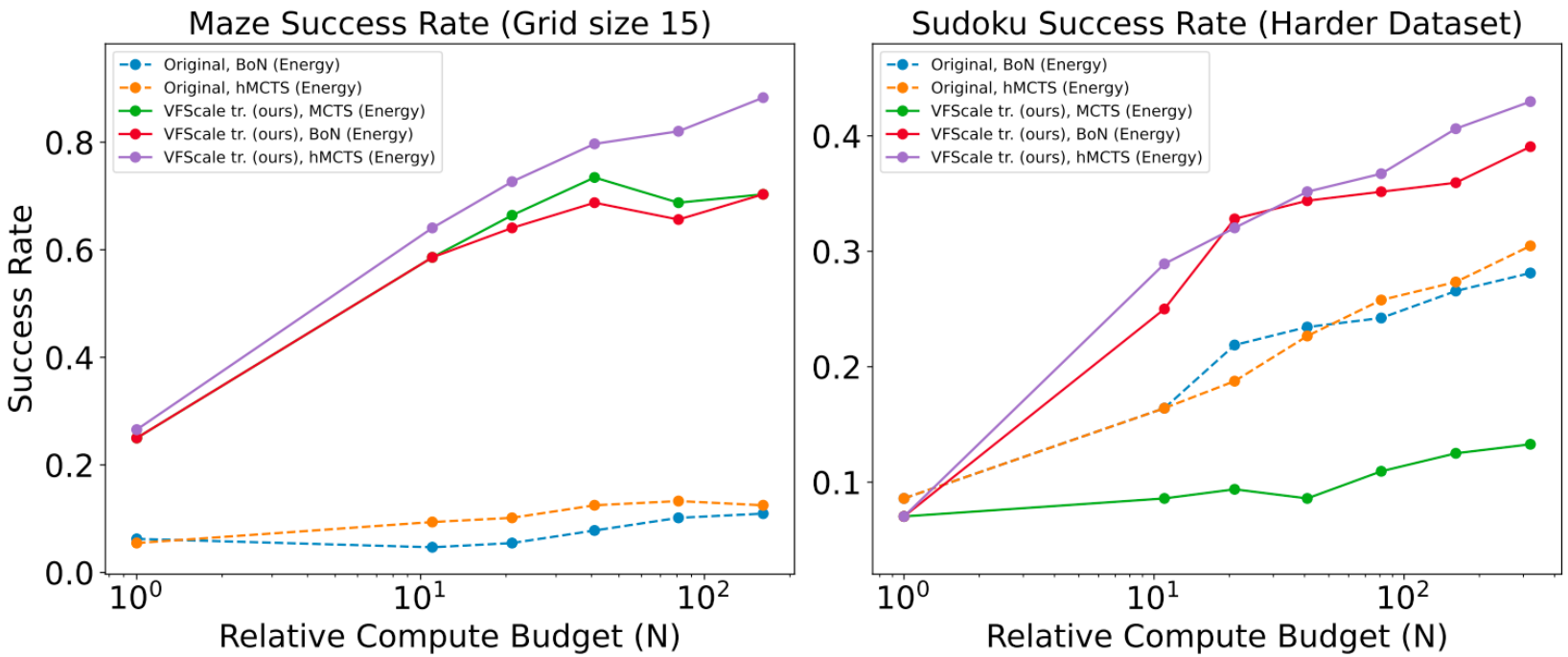
28.1% → 39.1%

Original → VFScale training

VFScale training is what makes the model's intrinsic energy usable for search.

Experiments: test-time scalability of VFScale

With the same compute budget, VFScale + hMCTS keeps improving while the original methods quickly plateau.



88.3%
Maze 15×15, N = 161

43.0%
hard Sudoku, N = 321

+18 pts
over BoN on Maze at the final budget

Take-away & contact

VFScale turns intrinsic energy + hybrid search into real OOD gains.

Three take-aways

- **Diffusion models can scale at test time** without an external verifier.
- **The key is to make energy rank candidate quality** and then use search that matches the denoising stage.
- **On hard OOD Maze and Sudoku, VFScale reaches** 88.3% and 43.0%, respectively.

Tao Zhang*, Jia-Shu Pan*, Ruiqi Feng, Tailin Wu†
Zhejiang University · Westlake University
Contact: wutailin@westlake.edu.cn

Code: github.com/AI4Science-WestlakeU/VFScale

Headline summary

88.3% **Maze 15×15**

43.0% **hard Sudoku**

MRNCL + KL improve the energy landscape
hMCTS spends budget more effectively

Paper

Code

Thank you!
Welcome to contact us after the talk.