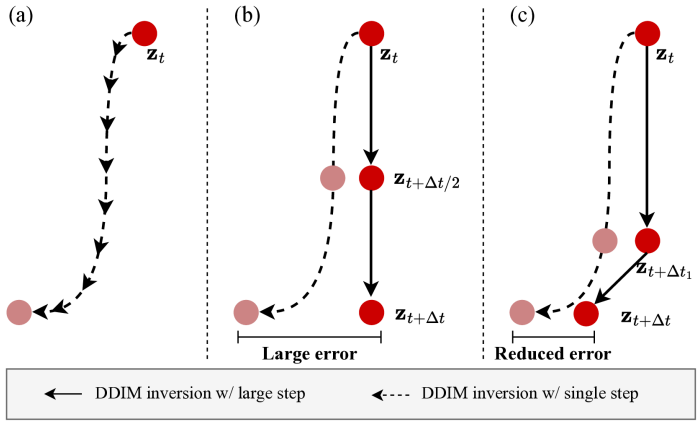
ICML
International Conference On Machine Learning

Timestep Rescheduling in Diffusion Inversion

Shangquan Sun, Ting Gong, Zhirui Liu, Jiamin Wu, Runkai Zhao, Mianxin Liu, Wenqi Ren, Xiaochun Cao

A training-free scheduler that reduces inversion error by reallocating the same timesteps.

No Training | Plug-In | No Extra Inference Cost



Uniform schedule → accumulated drift
TRDI schedule → better reconstruction/editing

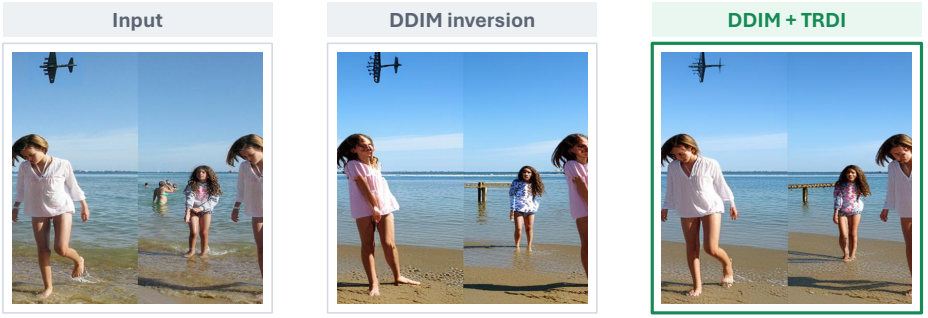
The Bottleneck: Inversion Error Accumulates

DDIM inversion is fast, but each approximate reverse step leaves reconstruction drift.

- Real-image editing first maps an image back to a latent noise trajectory.
- Uniform timestep jumps are convenient, but not equally safe across the diffusion timeline.
- In few-step settings, coarse jumps make local errors visible as structural artifacts.

Question

If the solver and model are fixed, can we reduce error simply by moving the timesteps?

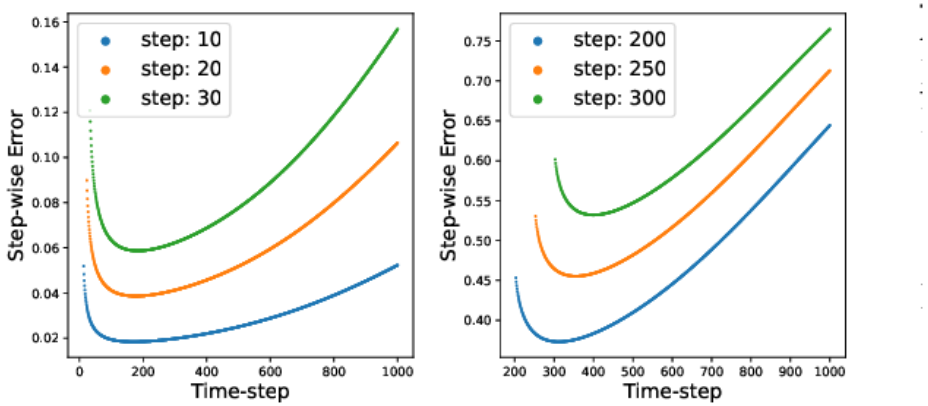


Same 4-step budget on SDXL Turbo; TRDI only changes where the steps are placed.

We treat timestep allocation itself as a controllable part of diffusion inversion.

Observation: Error Is Not Uniform Over Time

The scale of inversion deviation depends strongly on timestep size and position.



Large errors concentrate at the beginning and end of the diffusion timeline.

Error Decomposition

$$\delta(z_t, t, \Delta t) = \| c\alpha(t, \Delta t) \cdot \Delta\varepsilon\theta \|$$

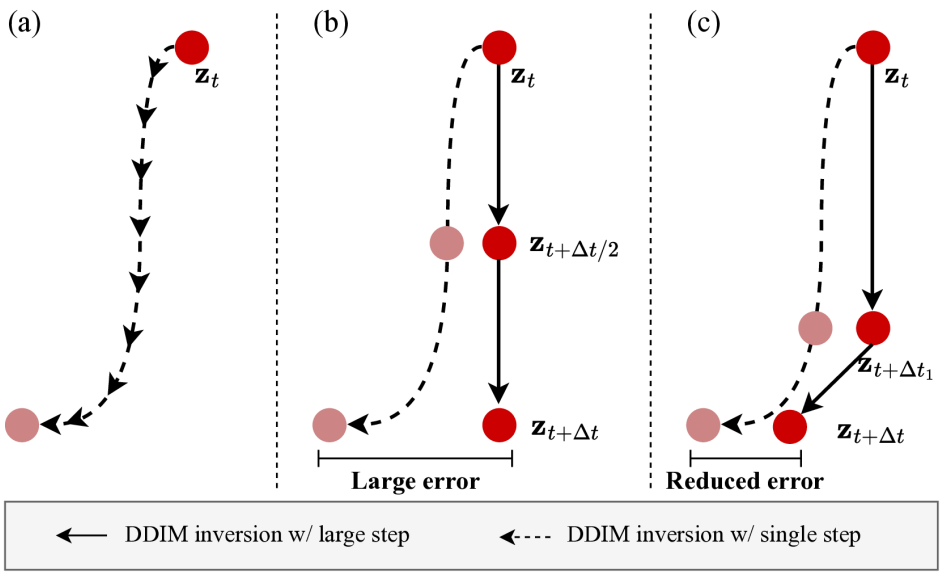
scheduler-dependent coefficient × local prediction difference

We cannot cheaply optimize the true global error, so TRDI uses cα as a lightweight local defect surrogate.

Interpretation: adaptive discretization without changing the solver.

TRDI: Coarse-to-Fine Timestep Rescheduling

Keep the model, solver, and number of steps fixed; only move the timestep locations.



1 Global Rescale

Stretch the original grid with γ to move steps away from high-error jumps.

2 Local DP

Search within a small window and minimize accumulated surrogate error.

3 Reuse Schedule

Precompute once; inference uses the same number of denoising evaluations.

Training-free

Plug-in to DDIM / ReNoise / NPI / GNRI

Zero extra inference cost

The Algorithm Is a Scheduler-Level Retrofit

TRDI optimizes a timetable, not the denoising model.

Algorithm 1 TRDI: Timestep Rescheduling in Diffusion Inversion

Require: number of inversion steps K, noise schedule $\{\alpha_t\}_{t=1}^{T}$ where T is the number of training diffusion steps, original timesteps $\{t_k\}_{k=1}^{K}$, window width d, scaling hyperparameter γ

Ensure: Rescheduled inversion timesteps $\{\hat{t}_k\}_{k=1}^{K}$

Timestep Rescaling:
Update each t_k by Eq. 11.
Dynamic Programming:
Initialize accumulative error map $\mathbf{E} \in \mathbb{R}^{K\times T}$.
Initialize intermediate timestep recorder $\mathbf{R} \in \mathbb{R}^{K\times T}$.
for each step $k \in \{1, \ldots, K\}$ **do**
  for each timestep $t \in \{t_k - d, \ldots, t_k + d\}$ **do**
    if $k = 1$ **then**
      Update $\mathbf{E}[1, t]$ using Eq. 12.
    else
      Update $\mathbf{E}[k, t]$ using Eq. 13.
      Record the $\arg\min$ of Eq. 13 in $\mathbf{R}[k, t]$.
    end if
  end for
end for
Retrieve timesteps:
$\hat{t}_K \leftarrow \min\{\mathbf{E}[K, t]\}_{t=t_K-d}^{t_K+d}$.
for each step $k \in \{K - 1, \ldots, 1\}$ **do**
  $\hat{t}_k \leftarrow \mathbf{R}[k, \hat{t}_{k+1}]$.
end for
return $\{\hat{t}_k\}_{k=1}^{K}$.

Operational View

- Input: original timestep grid and known noise schedule.
- Compute cα(t, Δt) from the scheduler; no model forward pass is needed.
- Windowed DP keeps the schedule close to the original grid while reducing accumulated error.
- The final schedule is reused during inversion and editing.

Offline	0	0
schedule computation	extra trainable parameters	extra inference steps

Results: Consistent Gains Under the Same Budget

TRDI improves reconstruction and editing across multiple inversion backbones.

3.16-5.39%
LPIPS reduction on SD v1.5 reconstruction

24%
structure distance reduction on SDXL editing

4-step
strong gains in few-step SDXL Turbo setting

same K
no extra denoising evaluations

Method	PSNR↑	$SSIM_{\times 10^2}$ ↑	$LPIPS_{\times 10^3}$ ↓
DDIM	20.07	65.11	193.97
w/ Ours	$20.21_{0.70\%}$	$65.73_{0.95\%}$	$187.85_{3.16\%}$
ReNoise	22.35	69.46	166.27
w/ Ours	$22.67_{1.43\%}$	$70.42_{1.38\%}$	$157.30_{5.39\%}$
NPI	20.82	66.22	182.01
w/ Ours	$21.08_{1.25\%}$	$67.05_{1.25\%}$	$175.41_{3.63\%}$
GNRI	22.14	69.72	147.02
w/ Ours	$22.32_{0.81\%}$	$70.39_{0.96\%}$	$141.33_{3.87\%}$

Representative reconstruction table

Visual improvements often occur in localized but perceptually important structure regions.

Summary

Timestep allocation is an overlooked but effective control knob for diffusion inversion.

1 Analyze

- derive a scheduler-dependent error surrogate

2 Reschedule

- global rescale plus local dynamic programming

3 Improve

- better reconstruction and editing with no extra inference cost

Training-free | Plug-in | Same step budget