FLOW-LITHO: A PHYSICS-GUIDED WORLD MODEL FOR FAITHFUL PROCESS RESPONSES IN MULTI-STAGE COMPUTATIONAL LITHOGRAPHY

Anonymous authors
Paper under double-blind review

ABSTRACT

Computational lithography improves pattern fidelity and manufacturability in advanced integrated circuit (IC) fabrication through modeling and optimization of pattern transfer. However, process responses vary substantially across stages and local regions, making reliable multi-stage lithography modeling challenging. To this end, we propose ***Flow-Litho***, a physics-guided lithography world model (WM) for faithful process-response modeling, which recursively evolves the "***Layout-Mask-Resist Image-After Development Image (ADI)***" pipeline through intra-stage physics correction and cross-stage difference preservation. First, we formulate the heterogeneous multi-stage process as a latent-space flow-matching problem, learning process-conditioned velocity fields and evolving latent states through numerical integration. Second, we develop a physical dynamics correction mechanism that evaluates regional response discrepancies using stage-specific approximate physical operators and couples their gradients to correct the latent velocity field. Finally, we introduce a difference-preserving self-evolving optimization paradigm for joint training, which uses paired supervision for the same layout across process conditions to guide intermediate-state recomposition and downstream replay, enabling the model to preserve process-induced differences while adapting to upstream rollout deviations. Experiments show that Flow-Litho achieves state-of-the-art performance in multi-stage lithography modeling and process-parameter planning. Our Code is publicly available at GitHub.

1 INTRODUCTION

Lithography transfers designed patterns onto wafers through optical imaging, photoresist reactions, and development, while computational lithography models and optimizes this process to improve pattern fidelity and manufacturability (Jiang et al., 2024; Meng et al., 2026). As feature sizes shrink, contour placement and critical dimensions become increasingly sensitive to exposure dose, defocus, and illumination. Evaluating these responses through high-fidelity simulations or wafer experiments incurs substantial computational and manufacturing costs (He et al., 2025). Therefore, developing computational lithography models that balance accuracy and efficiency is essential for pre-manufacturing lithography simulations and process planning.

Recent research has primarily focused on physics-based solvers and learning-based approximations. Physics-based methods enable gradient-driven mask optimization through differentiable optical and resist modeling (Chen et al., 2024a; Wang et al., 2025b), but repeated evaluation across process conditions remains costly. Learning-based methods instead predict mask designs or patterning outcomes from data, using efficient forward inference to approximate physical simulation (Wang et al., 2025a; Jin et al., 2025). These typically treat individual tasks as static mappings, whereas Mask, Resist Image, and After Development Image (ADI) in a lithography workflow are progressively formed based on preceding states. Thus, complete lithography modeling requires moving beyond single-step outcome prediction toward process-conditioned multi-stage state evolution.

World models (WMs) provide a unified framework for modeling sequential processes. By learning environment dynamics and predicting how states evolve under different conditions or actions, WMs enable long-horizon prediction and planning (Hafner et al., 2025). In computational lithog-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

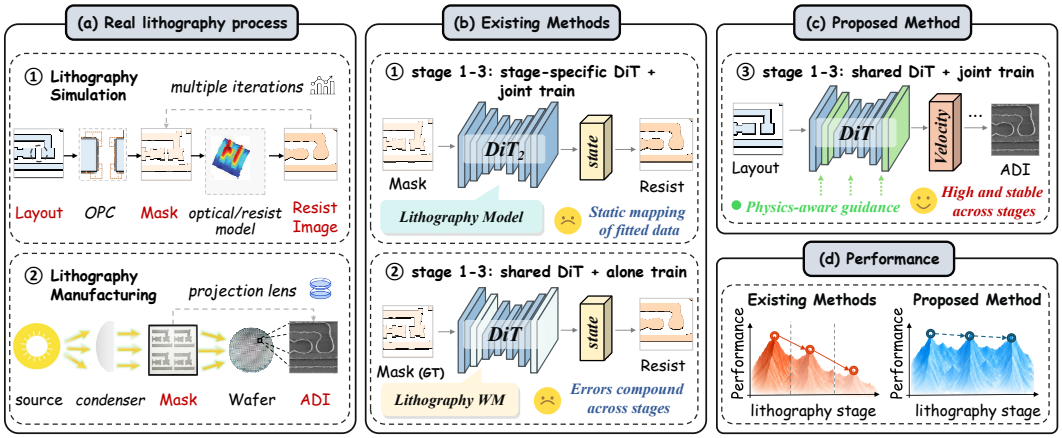


Figure 1: Motivation for Flow-Litho. Comparing lithography workflows and modeling paradigms motivates physics guidance and cross-stage joint learning for faithful multi-stage rollouts.

raphy, Layout, Mask, Resist Image, and ADI can be formulated as successive process-conditioned states, linking mask design, pattern formation, and manufacturing observations within a unified state-evolution framework. Prior work (Jiang et al., 2026) has introduced this paradigm into multi-stage lithography, demonstrating the feasibility of state transitions across the lithography workflow.

However, process-response modeling under the WM paradigm still faces three challenges. **(1) Stage Heterogeneity**: different stages are governed by distinct physical mechanisms and mapping objectives, requiring a unified model to balance cross-stage sharing with stage-specific responses. **(2) Local Non-uniformity**: even within each stage, process variations act unevenly across geometries, making subtle boundary changes easy to dilute under global image-level objectives. **(3) Cross-stage Coupling**: these local responses and evolution deviations propagate downstream, requiring models to accommodate upstream perturbations while preserving genuine process-induced differences.

To overcome these challenges, we propose **Flow-Litho**, as shown in Figure 1, a physics-guided lithography WM for faithful process-response modeling that unifies recursive state evolution along the "Layout-Mask-Resist Image-ADI" pipeline and supports target-driven process-parameter planning. Specifically, Flow-Litho unifies heterogeneous stage dynamics with a shared Diffusion Transformer (DiT) that learns latent velocity fields through flow matching. To account for spatially nonuniform responses, stage-specific approximate physical operators quantify regional response discrepancies. A coupled solve determines gradient-combination weights from these discrepancies and cross-region gradient interactions to correct the latent velocity field. The latent state is then advanced by numerically integrating the corrected velocity field. Furthermore, we introduce a self-evolving optimization paradigm for joint training, in which paired supervision for the same layout across process conditions constrains intermediate-state recomposition to preserve process-induced differences, while downstream replay adapts the model to upstream rollout deviations.

Our contributions are as follows:

- We propose a Flow-Litho, a physics-guided lithography world model that unifies heterogeneous stage dynamics through latent-space flow matching, enabling recursive multi-stage evolution and target-driven process-parameter planning.
- We develop a physical dynamics correction mechanism that coordinates latent velocity updates through a coupled solve based on regional response discrepancies and cross-region gradient interactions, without introducing additional trainable parameters.
- We introduce a self-evolving optimization paradigm that recomposes intermediate states and replays downstream evolution, mitigating state-distribution shift and stabilizing rollout.
- Experiments show that Flow-Litho achieves Mask EPEs of 0.42 nm and 0.83 nm in forward evolution on the 55 nm ID and unseen-process OOD sets, respectively, and 78.29% exact process-parameter recovery in ID planning with 12 candidate rollouts.

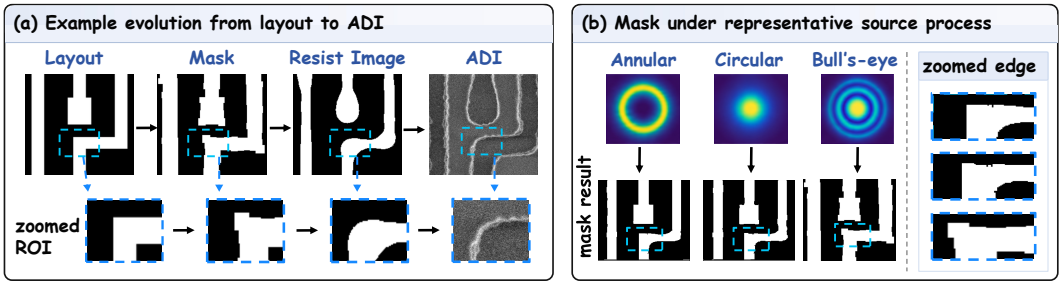


Figure 2: Multi-stage lithography samples and illumination-dependent mask variations. Local comparisons highlight contour evolution and subtle source-dependent boundary changes.

2 PRELIMINARIES

2.1 COMPUTATIONAL LITHOGRAPHY PROCESS

Computational lithography combines inverse mask compensation with forward resist simulation to model pattern transfer, while an observation mapping relates resist patterns to measured ADIs.

In the layout-to-mask stage, optical proximity correction (OPC) adjusts mask geometry to compensate for diffraction and resist effects (Chen et al., 2024a). For a target layout $X^{(0)}$, continuous-mask optimization minimizes the patterning error:

$$\min_{0\le M\le 1} \left\| P(M;p) - Y_{\text{ref}}^{(1)} \right\|_2^2, \tag{1}$$

where M is mask amplitude transmittance, P the approximate patterning operator, and $Y_{\text{ref}}^{(1)} = X^{(0)}$ the desired resist response. The parameters $p = (d, f, T, s)$ specify relative dose, defocus (nm), effective resist threshold, and source category, with s selecting the source image S_s.

In the mask-to-resist image stage, optical imaging determines exposure intensity, and resist reactions and development form the pattern. Under the scalar thin-mask approximation, Abbe imaging (Evanschitzky et al., 2009) and a smooth threshold approximate the developed resist image:

$$Y_{\text{ref}}^{(2)} = P\left(X^{(1)};p\right) = \text{sigmoid}\left(\frac{d\sum_j S_s(q_j)\left|\mathcal{F}^{-1}\left[h_f(\nu+aq_j)\,\mathcal{F}\left(X^{(1)}\right)\right]\right|^2 - cT}{\tau}\right), \tag{2}$$

where $X^{(1)}$ and $X^{(2)}$ denote the mask and simulated developed resist image; $\mathcal{F}$ and $\mathcal{F}^{-1}$ are the 2D Fourier transform and its inverse. Illumination samples $q_j \in [-1,1]^2$ have nonnegative, unit-sum weights $S_s(q_j)$ obtained by discretizing and normalizing the source image. Here, ν is spatial frequency, h_f the defocus-dependent pupil, and a, c, and τ control source shifts, threshold scaling, and response smoothness, respectively.

In the resist-image-to-ADI mapping, simulated resist patterns describe the expected developed geometry, while scanning electron microscopy (SEM) records the fabricated structure (Villarrubia et al., 2015). An empirical combination of region and edge responses approximates the low-frequency ADI appearance:

$$Y_{\text{ref}}^{(3)} = G\left(X^{(2)}\right) = Q\left(\text{clip}_{[0,1]}\left[\beta_0 + \beta_r R\left(X^{(2)}\right) + \beta_e H\left(X^{(2)}\right)\right]\right) \approx Q\left(X^{(3)}\right), \tag{3}$$

where $X^{(3)}$ is measured ADI, G the observation operator, R and H the smoothed region and edge responses, and Q a Gaussian low-pass filter. The offset β_0 and response weights β_r, β_e are calibrated on aligned training pairs and fixed; clipping bounds intensities to $[0, 1]$. Operator discretization and calibration are detailed in Appendix D.

2.2 LITHOGRAPHY DATASET

As shown in Figure 2, the dataset is collected from a 55 nm manufacturing line and contains 280,000 spatially aligned Layout-Mask-Resist Image-ADI samples with corresponding process parameters

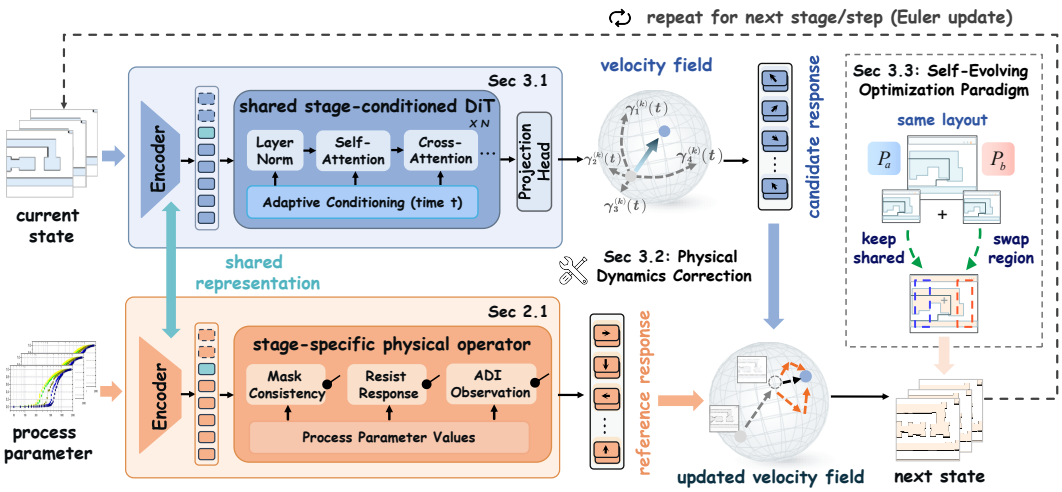


Figure 3: Overview of Flow-Litho. Physics correction and cross-stage difference preservation guide recursive evolution, retaining process responses while adapting to upstream rollout deviations.

and illumination-source images (Jiang et al., 2026). Each image covers a 6000×6000 nm field of view at 512×512 resolution. The data span 36 process configurations combining three illumination sources (annular, circular, and bull's-eye), relative doses of $1.0, 1.2$, defocus values of $0, 50$ nm, and resist thresholds of $0.09231251, 0.1236402, 0.1436665$. Across all 36 conditions, 2,000 layouts are paired, yielding 72,000 sequences. Cross-condition variations primarily manifest as subtle contour shifts and linewidth changes near line ends, corners, and boundaries, and can be quantified in nanometers from the physical pixel size.

2.3 Related Work on Computational Lithography

Computational lithography is evolving from single-stage mask optimization and patterning prediction toward physics-informed multi-stage modeling. For mask optimization, DiffOPC (Chen et al., 2024b) updates mask edges using patterning-error gradients under mask-rule constraints, while OptiCo (Son et al., 2026) embeds optical diffraction into convolution through optical phase kernels. WMs support recursive rollout and goal-directed planning through learned conditional dynamics (Hafner et al., 2025; Zhou et al., 2025). LithoDreamer (Jiang et al., 2026) extends this paradigm to lithography, enabling multi-stage evolution and inverse planning through stage-specific latent constraints and intervention policies. However, faithfully modeling process responses across stages remains challenging. Flow-Litho addresses this challenge by explicitly guiding continuous state evolution with stage-specific approximate physical operators to improve multi-stage evolution fidelity and support process-parameter planning.

3 Method

As shown in Figure 3, Flow-Litho integrates a shared DiT's conditional velocity field after correction by approximate physical responses. GT-pair pretraining is followed by cascaded joint training with difference-preserving recomposition and downstream replay to accommodate upstream deviations.

3.1 Stage-Conditioned Velocity Field Modeling

Flow-Litho models inverse mask compensation, patterning simulation, and simulation-to-measurement mapping as three stage-conditioned transitions indexed by $k = 1, 2, 3$. All stages share a frozen VAE encoder E and decoder D (Rombach et al., 2022), with posterior-mean encoding and decoded images converted to grayscale, rescaled, and clipped to $[0, 1]$. The stage input $X_{\text{in}}^{(k)}$ can be a GT, evolved, or recomposed upstream state; $u_{\text{in}}^{(k)} = E(X_{\text{in}}^{(k)})$ provides the initial latent state and fixed image condition. Source-image features and numerical conditions, encoded from

fixed-template text by frozen BERT E_n (Devlin et al., 2019), condition the velocity:

$$\begin{aligned} e_p &= \mathrm{MLP}_\theta([\mathrm{Flatten}(E(S_s)); E_n(\mathrm{text}(d, f, T))]), \\ v_\theta^{(k)}(u,t) &= \mathrm{DiT}_\theta\Big(u, u_{\mathrm{in}}^{(k)}, e_p, t, k\Big), \end{aligned} \tag{4}$$

where θ denotes trainable parameters; Flatten, text, and $[\cdot;\cdot]$ denote flattening, template conversion, and concatenation, and MLP is a multilayer perceptron. DiT combines a shared backbone with stage-specific projections and output heads (Peebles & Xie, 2023). The current latent state is u; $t \in [0,1]$ measures numerical evolution progress, not physical time, and p remains fixed throughout a rollout, ensuring consistent process conditioning as the latent state evolves toward the stage output.

With $u_{\mathrm{GT}}^{(k)} = E(X^{(k)})$, linear-path flow matching (Lipman et al., 2023) defines the interpolated state and a constant target velocity connecting the stage input to its downstream latent representation:

$$u_t^{(k)} = (1-t)u_{\mathrm{in}}^{(k)} + t u_{\mathrm{GT}}^{(k)}, \qquad v_\star^{(k)} = u_{\mathrm{GT}}^{(k)} - u_{\mathrm{in}}^{(k)}. \tag{5}$$

At uniformly sampled t, the physics-corrected velocity $\widetilde{v}_\theta^{(k)}$ is regressed toward $v_\star^{(k)}$. Rollouts start at $\widehat{u}_0^{(k)} = u_{\mathrm{in}}^{(k)}$ and use Euler integration to propagate the corrected latent dynamics:

$$\begin{aligned} \widehat{u}_{t_{n+1}}^{(k)} &= \widehat{u}_{t_n}^{(k)} + K_k^{-1}\widetilde{v}_\theta^{(k)}\Big(\widehat{u}_{t_n}^{(k)}, t_n\Big), \\ \widehat{X}^{(k)} &= F_\theta^{(k)}(X_{\mathrm{in}}^{(k)}, p) = D\Big(\widehat{u}_1^{(k)}\Big), \quad t_n = n/K_k, \quad n = 0, \ldots, K_k - 1, \end{aligned} \tag{6}$$

where $K_k = 3$ for all stages and $F_\theta^{(k)}$ comprises encoding, corrected integration, and decoding. Starting from $X^{(0)}$, each terminal state conditions the next stage, connecting heterogeneous transitions within a shared latent representation and propagating process-dependent geometric responses.

3.2 Physical Dynamics Correction

Shared latent representations couple local responses, so correcting one region can affect another. We jointly resolve regional response discrepancies against approximate physical references to guide state evolution while accounting for interactions among corrections within the shared latent representation. This couples local physical constraints to the global evolution of the predicted stage.

3.2.1 Regional Response Discrepancies

For the current stage and time, let $v = v_\theta^{(k)}(u,t)$. Extrapolation along this velocity gives a candidate endpoint whose decoded image approximates the stage response for physical comparison:

$$z = u + (1-t)v. \tag{7}$$

Using Equation (1)-Equation (3), we map candidate and reference responses into a common comparison domain for each stage, ensuring discrepancies measure comparable responses:

$$\begin{aligned} \big(Y^{(k)}, Y_{\mathrm{ref}}^{(k)}\big) &= \begin{cases} \big(P(D(z);p), X_{\mathrm{in}}^{(1)}\big), & k = 1, \\ \big(D(z), P(X_{\mathrm{in}}^{(2)};p)\big), & k = 2, \\ \big(Q(D(z)), G(X_{\mathrm{in}}^{(3)})\big), & k = 3, \end{cases} \\ m(x) &= \big[Y^{(k)}(x) - Y_{\mathrm{ref}}^{(k)}(x)\big]^2, \end{aligned} \tag{8}$$

where $Y^{(k)}$ is the candidate response and $m(x)$ its squared discrepancy at pixel x. To capture contour-sensitive linewidths and spacings, horizontal and vertical midlines partition the image into four quadrants. Thresholding $Y_{\mathrm{ref}}^{(k)}$ for $k = 1, 2$ and $X_{\mathrm{in}}^{(3)}$ for $k = 3$ at 0.5 yields C_{bin}. Its morphological gradient (Soille, 2003) defines a boundary band to emphasize contour deviations:

$$\begin{aligned} W &= \mathrm{Dilate}(C_{\mathrm{bin}}) - \mathrm{Erode}(C_{\mathrm{bin}}), \\ e_j &= w_b \frac{\sum_{x\in\Omega_j} W(x)m(x)}{\max(1, \sum_{x\in\Omega_j} W(x))} + w_a \frac{\sum_{x\in\Omega_j} m(x)}{\max(1, |\Omega_j|)}, \quad j = 1, \ldots, 4, \end{aligned} \tag{9}$$

where Ω_j contains valid pixels in quadrant j, and $w_b = w_a = 0.5$. Dilation and erosion each use one 3×3 square neighborhood over the full image. The region-wide term includes boundary pixels, jointly characterizing contour displacement and regional structure at complementary spatial scales.

3.2.2 REGIONALLY COORDINATED VELOCITY CORRECTION

Differentiating each discrepancy through the candidate endpoint and response mapping gives its velocity sensitivity; gradient inner products characterize how a correction derived from one region affects the response discrepancies of other regions through shared latent variables (Liu et al., 2025):

$$g_j = \nabla_v e_j = (1-t)\nabla_z e_j, \qquad B_{ij} = g_i^\top g_j, \tag{10}$$

where g_j is the full-latent gradient for region j and B is the Gram matrix of the four gradients. To first order, a step along $-g_j$ decreases region i's discrepancy when $B_{ij} > 0$ and increases it when $B_{ij} < 0$, distinguishing compatible corrections from conflicting regional directions.

For a velocity correction δ, the first-order regional change is $g_j^\top \delta$. We seek changes close to $-b_j$, where b_j is a desired reduction beyond the reference tolerance. Penalizing large corrections yields a regularized objective balancing regional response reduction against velocity correction magnitude:

$$\begin{aligned} b_j &= \alpha t^2 \max(e_j - \varepsilon_j, 0), \\ \Delta v &= \arg\min_{\delta} \left\{ \frac{1}{2} \sum_{j=1}^{4} (g_j^\top \delta + b_j)^2 + \frac{\lambda}{2} \|\delta\|_2^2 \right\}, \end{aligned} \tag{11}$$

where the stage- and region-specific ε_j is the training-split 90th percentile of discrepancies from GT input-target pairs. We set $\alpha = 0.2$ and $\lambda = 10^{-3}\,\mathrm{tr}(B)/4 + 10^{-8}$, where tr denotes the trace; t^2 attenuates early corrections. Regions within tolerance remain in the objective with $b_j = 0$, penalizing unintended changes induced by corrections to other regions through the shared representation.

The optimality condition $\lambda \Delta v + \sum_j g_j (g_j^\top \Delta v + b_j) = 0$ places the solution in the span of the regional gradients. A gradient expansion yields the coefficient system for regional coordination:

$$(B + \lambda I_4)\mathbf{a} = \mathbf{b}, \qquad \Delta v = -\sum_{j=1}^{4} a_j g_j, \tag{12}$$

where $\mathbf{a} = (a_j)_{j=1}^4$ contains combination coefficients, $\mathbf{b} = (b_j)_{j=1}^4$ contains desired reductions, and I_4 is the identity matrix. Since B is positive semidefinite and $\lambda > 0$, the solve is unique. The resulting correction is added to the base velocity with norm control to limit its influence:

$$\widetilde{v}_\theta^{(k)}(u,t) = v + \eta \Delta v, \qquad \eta = \min\left\{1, \frac{0.2\|v\|_2}{\|\Delta v\|_2 + 10^{-8}}\right\}. \tag{13}$$

Thus, the correction norm is at most 20% of the base-velocity norm. Corrections are recomputed before each update in Equation (6). Training stops gradients through the entire $\eta \Delta v$ while retaining base-velocity and recursive-state gradients, incorporating physical guidance through a first-order surrogate. Appendix E develops the derivation and gradient treatment for cascaded latent evolution.

3.3 DIFFERENCE-PRESERVING SELF-EVOLVING OPTIMIZATION

GT-conditioned and recursive states can differ in distribution (Bengio et al., 2015), while unconstrained cross-process mixing can distort genuine responses. Joint training retains GT-pair supervision and ordinary cascaded learning, augmented with label-constrained recomposition and downstream replay to accommodate evolving upstream deviations under process-specific supervision.

3.3.1 DIFFERENCE-PRESERVING STATE RECOMPOSITION

For one layout, paired conditions $p_a \neq p_b$ have aligned labels $X_q^{(k)}$, where $q \in \{a, b\}$ indexes process branches. Unrecomposed trajectories follow $\widehat{X}_q^{(k)} = F_\theta^{(k)}(\widehat{X}_q^{(k-1)}, p_q)$ from $\widehat{X}_q^{(0)} = X^{(0)}$. At transitions $k \to k+1$, $k = 1, 2$, evolved values are exchanged only at jointly valid, label-consistent locations, distinguishing shared structure from process-dependent geometry:

$$\begin{aligned} A_{ab}^{(k)} &= V_{ab}^{(k)} \odot \mathbf{1}[X_a^{(k)} = X_b^{(k)}], \\ \widetilde{X}_a^{(k)} &= (1 - A_{ab}^{(k)}) \odot \widehat{X}_a^{(k)} + A_{ab}^{(k)} \odot \widehat{X}_b^{(k)}, \end{aligned} \tag{14}$$

where $V_{ab}^{(k)}$ is the jointly valid mask, $A_{ab}^{(k)}$ the exchange mask, and $\odot$ and $\mathbf{1}[\cdot]$ denote element-wise multiplication and an indicator. Branch b is symmetric. Labels gate exchange without supplying pixel values, introducing alternative upstream deviations while preserving label-differing regions.

3.3.2 DOWNSTREAM REPLAY AND JOINT OPTIMIZATION

Each recomposed state undergoes a complete next-stage rollout under its original condition, with the physical reference recomputed from that input. Original downstream labels supervise robustness to recomposition, while unrecomposed-state supervision constrains upstream drift:

$$\begin{aligned} \overline{X}_q^{(k+1)} &= F_\theta^{(k+1)}(\widetilde{X}_q^{(k)}, p_q), \qquad k = 1, 2, \\ \mathcal{L}_{\mathrm{SE}} &= \frac{1}{4} \sum_{k=1}^{2} \sum_{q \in \{a,b\}} \left[\ell(\widehat{X}_q^{(k)}, X_q^{(k)}) + \ell(\overline{X}_q^{(k+1)}, X_q^{(k+1)}) \right], \end{aligned} \tag{15}$$

where $\overline{X}_q^{(k+1)}$ is the replayed endpoint and ℓ is mean absolute error over valid pixels. Replay gradients traverse the frozen codecs' input derivatives and state recursion to update upstream and downstream trainable modules. Fixed labels constrain states that evolve with model updates, promoting adaptation to current upstream deviations while retaining process-induced differences. Joint gradients couple upstream state generation with downstream reconstruction under fixed process conditions, enabling downstream supervision to refine upstream evolution.

4 EXPERIMENTS

We evaluate Flow-Litho on three tasks: process-conditioned forward evolution, target-driven process-parameter planning, and goal-conditioned trajectory planning under fixed process parameters. Section 4 reports the first two, while Appendix C.3 presents the third. Evaluation protocols and metrics are detailed in Appendices A and B, respectively. Supplementary results and visualizations appear in Appendices C.1-C.3, with additional ablations in Appendix C.4.

4.1 EXPERIMENTAL SETTINGS

4.1.1 IMPLEMENTATION DETAILS

Flow-Litho comprises a frozen shared `sd-vae-ft-mse` VAE, a frozen `BERT-base-uncased` encoder, and a randomly initialized shared DiT with 8 blocks, hidden dimension 384, and 6 attention heads. We use AdamW (Loshchilov & Hutter, 2019) with $(\beta_1, \beta_2) = (0.9, 0.999)$, weight decay 0.01, and gradient-norm clipping at 1.0. Pretraining runs for 24,000 updates with 500-step warm-up followed by cosine learning-rate decay from 10^{-4} to 5×10^{-5}; cascaded joint training runs for 8,000 updates at 10^{-5}. Training uses eight NVIDIA A100-80GB GPUs, a batch size of 4 per GPU, and seed 42. Each stage uses three physics-corrected Euler steps throughout both training and inference.

4.1.2 DATASETS AND TRAINING OBJECTIVES

The 55 nm dataset in Section 2.2 provides 260,000 training and 20,000 in-domain (ID) test samples. Out-of-domain (OOD) evaluation includes 3,000 samples under an unseen 55 nm condition (annular source, relative dose 1.0, defocus 0 nm, and threshold 0.119340) and the public 28 nm LithoSim dataset (He et al., 2025). Cross-node evaluation on LithoSim covers only Mask and Resist Image to assess geometric response transfer across manufacturing nodes, as ADI observations are unavailable.

Pretraining combines flow matching and stage-endpoint image supervision under GT upstream inputs, weighted by 1 and 0.1, respectively. Image supervision combines MSE, BCE, soft Dice (Milletari et al., 2016), and edge losses with weights 1, 0.5, 0.5, and 0.1, respectively; ADI uses only MSE and edge losses. Joint training retains both pretraining objectives at their original weights and adds recursive endpoint supervision and the self-evolving loss in Equation (15), each weighted by 0.1. Both training phases apply physical velocity correction with detached correction increments.

4.2 PROCESS-CONDITIONED FORWARD EVOLUTION

Table 1 compares multi-stage forward evolution under ID and OOD settings. On ID data, Flow-Litho improves ADI mIoU by 12.03 percentage points and reduces MSE by 66.3% over DreamerV3, improving both region overlap and grayscale reconstruction. These gains extend to changes in process conditions and technology nodes. Compared with LithoDreamer, Flow-Litho improves Mask

Table 1: Multi-stage forward evolution under ID and OOD conditions (mIoU: %; MSE: 10^{-3}; EPE: nm). GT/R denote ground-truth upstream inputs/recursive rollouts; n/a indicates unsupported stage evaluation. **Best** and second-best values are highlighted.

		Mask			Resist Image			ADI		
Method	Input	mIoU↑	MSE↓	EPE↓	mIoU↑	MSE↓	EPE↓	mIoU↑	MSE↓	EPE↓
55 nm: in-domain										
LithoNet (Shao et al., 2021)	GT	85.70	57.92	46.16	82.14	78.67	33.31	8.91	247.82	13.39
LMLitho (Wang et al., 2025a)	GT	91.95	19.03	7.12	96.35	18.66	6.31	40.96	162.41	5.54
Unitho (Jin et al., 2025)	GT	92.24	25.40	8.69	91.98	25.16	8.62	n/a	n/a	n/a
OptiCo (Son et al., 2026)	GT	92.05	26.08	8.77	93.73	23.91	8.28	n/a	n/a	n/a
DIAMOND (Alonso et al., 2024)	R	89.33	28.83	23.52	91.63	25.28	21.67	65.30	96.71	24.91
DINO-WM (Zhou et al., 2025)	R	90.04	26.51	21.96	93.89	22.47	17.37	69.41	120.51	19.71
DreamerV3 (Hafner et al., 2025)	R	93.01	21.52	12.06	95.20	16.23	9.38	73.05	79.36	15.92
LithoDreamer (Jiang et al., 2026)	R	96.91	13.74	1.58	97.77	10.93	0.96	78.27	30.29	3.74
Flow-Litho (ours)	R	**97.95**	**11.94**	**0.42**	**98.44**	**10.07**	**0.37**	**85.08**	**26.74**	**3.67**
55 nm: unseen process condition										
LithoNet (Shao et al., 2021)	GT	55.95	97.67	8.19	46.67	113.34	8.64	6.07	265.83	13.45
LMLitho (Wang et al., 2025a)	GT	66.58	73.86	28.00	66.44	77.87	7.00	42.46	138.73	75.61
Unitho (Jin et al., 2025)	GT	67.61	70.83	10.54	71.84	94.82	1.93	n/a	n/a	n/a
OptiCo (Son et al., 2026)	GT	67.27	71.32	13.84	73.22	87.37	1.70	n/a	n/a	n/a
DIAMOND (Alonso et al., 2024)	R	73.19	72.48	18.01	74.68	65.22	5.97	60.55	93.83	7.31
DINO-WM (Zhou et al., 2025)	R	75.98	69.31	15.67	76.25	56.86	2.69	64.06	79.88	4.28
DreamerV3 (Hafner et al., 2025)	R	76.07	53.83	10.62	79.51	42.93	2.08	69.32	61.93	4.11
LithoDreamer (Jiang et al., 2026)	R	77.52	37.07	1.77	82.91	27.58	1.06	76.04	33.62	4.05
Flow-Litho (ours)	R	**88.66**	**19.18**	**0.83**	**92.55**	**12.17**	**0.78**	**80.98**	**27.15**	**3.93**
28 nm: cross-node transfer										
LithoNet (Shao et al., 2021)	GT	31.49	154.79	23.49	31.49	138.72	20.52	n/a	n/a	n/a
LMLitho (Wang et al., 2025a)	GT	30.64	167.42	4.63	29.08	188.46	7.90	n/a	n/a	n/a
Unitho (Jin et al., 2025)	GT	33.98	120.35	4.78	30.59	172.49	5.33	n/a	n/a	n/a
OptiCo (Son et al., 2026)	GT	33.94	124.01	5.36	38.52	131.03	5.08	n/a	n/a	n/a
DIAMOND (Alonso et al., 2024)	R	60.23	78.47	8.55	58.26	93.01	9.74	n/a	n/a	n/a
DINO-WM (Zhou et al., 2025)	R	62.37	60.37	5.85	61.32	57.71	5.97	n/a	n/a	n/a
DreamerV3 (Hafner et al., 2025)	R	72.97	47.29	4.90	69.61	50.39	5.47	n/a	n/a	n/a
LithoDreamer (Jiang et al., 2026)	R	**89.75**	26.78	1.82	81.16	33.29	1.37	n/a	n/a	n/a
Flow-Litho (ours)	R	85.92	**23.36**	**1.59**	**88.89**	**17.90**	**0.77**	n/a	n/a	n/a

and Resist Image mIoU by 11.14 and 9.64 percentage points under unseen 55 nm conditions and reduces Resist Image EPE by 43.8% on 28 nm data. Despite a 3.83-point deficit in Mask mIoU, this improvement shows stronger downstream contour fidelity under cross-node transfer.

Table 2 evaluates LRC violation counts in resist images from ID forward rollouts against commercial lithography simulation. Compared with LithoDreamer, Flow-Litho reduces the mean absolute relative deviations in Bridge and EPE violation counts by 84.1% and 42.4%, respectively. Complementing the continuous geometric metrics in Table 1, these results indicate more faithful reproduction of violation statistics when local geometric deviations exceed manufacturing tolerances.

4.3 Target-Driven Process-Parameter Planning

Given a layout and target ADI, we search for process parameters using a combined contour-region discrepancy. GP-BO uses four random initial candidates; protocols and supplementary analyses appear in Appendices A.2 and C.2, respectively. Table 3 compares models and search strategies. With shared GP-BO and 12 rollouts, Flow-Litho improves exact parameter recovery by 22.72 percentage points over LithoDreamer, suggesting that accurate process-response modeling facilitates candidate discrimination. Random search achieves 93.06% process-response mIoU but only 28.13% recovery accuracy, indicating that within-model response similarity does not imply correct parameter identification. Compared with exhaustive search, GP-BO reduces rollout evaluations by two-thirds with a 0.71 nm increase in target EPE, quantifying the trade-off between search budget and target fidelity.

Table 2: LRC violation counts versus commercial simulation under conditions A-E (Appendix B, Table B.1). The final row reports mean absolute relative deviations (%) across conditions.

Condition	Commercial			Unitho (GT)			LithoDreamer (R)			**Flow-Litho (R)**		
	Pinch	Bridge	EPE	Pinch	Bridge	EPE	Pinch	Bridge	EPE	Pinch	Bridge	EPE
A	1289	2784	12732	1219 $_{-5.4\%}$	2629 $_{-5.6\%}$	13009 $_{+2.2\%}$	1217 $_{-5.6\%}$	2724 $_{-2.2\%}$	12920 $_{+1.5\%}$	**1264** $_{-1.9\%}$	**2753** $_{-1.1\%}$	**12898** $_{+1.3\%}$
B	1335	3032	12134	1279 $_{-4.2\%}$	2866 $_{-5.5\%}$	13729 $_{+13.1\%}$	**1334** $_{-0.1\%}$	3095 $_{+2.1\%}$	13160 $_{+8.5\%}$	1330 $_{-0.4\%}$	**3049** $_{+0.6\%}$	**12954** $_{+6.8\%}$
C	1443	2657	5980	2586 $_{+79.2\%}$	3767 $_{+41.8\%}$	12104 $_{+102.4\%}$	1577 $_{+9.3\%}$	2683 $_{+1.0\%}$	5427 $_{-9.2\%}$	**1526** $_{+5.8\%}$	**2669** $_{+0.5\%}$	**6274** $_{+4.9\%}$
D	1441	2795	6096	2808 $_{+94.9\%}$	2965 $_{+6.1\%}$	8764 $_{+43.8\%}$	1407 $_{-2.4\%}$	3849 $_{+37.7\%}$	6621 $_{+8.6\%}$	**1425** $_{-1.1\%}$	**2684** $_{-4.0\%}$	**6447** $_{+5.8\%}$
E	1456	2808	7041	2460 $_{+69.0\%}$	2910 $_{+3.6\%}$	8650 $_{+22.9\%}$	1519 $_{+4.3\%}$	2761 $_{-1.7\%}$	6037 $_{-14.3\%}$	**1431** $_{-1.7\%}$	**2780** $_{-1.0\%}$	**7426** $_{+5.5\%}$
Error (%)	n/a	n/a	n/a	50.53	12.51	36.87	4.33	8.92	8.41	**2.18**	**1.42**	**4.84**

Table 3: Target-driven process planning. Acc_p denotes exact recovery. Target and response agreement use the target ADI and the same model's rollout under annotated parameters, respectively.

Method	Search	Rollouts	$\mathrm{Acc}_p \uparrow$ (%)	Target agreement		Process-response agreement	
				mIoU↑	EPE↓	mIoU↑	EPE↓
Model comparison with shared GP-BO							
DINO-WM (Zhou et al., 2025)	GP-BO	12	42.53	61.87	30.09	90.97	6.82
LithoDreamer (Jiang et al., 2026)	GP-BO	12	55.57	72.53	5.65	93.64	3.95
Flow-Litho	GP-BO	12	**78.29**	**80.41**	**4.08**	**94.93**	**1.89**
Search-strategy comparison with fixed Flow-Litho							
Flow-Litho	Random	12	28.13	78.21	4.95	93.06	2.40
Flow-Litho	Exhaustive	36	**84.02**	**83.76**	**3.37**	**98.51**	**1.35**

4.4 Ablation Studies

Table 4 examines physical velocity correction (PC), self-evolving optimization (SE), and their key mechanisms, with ablation controls detailed in Appendix C.4. Removing PC or SE degrades forward evolution and lowers exact parameter recovery by 31.66 and 31.90 percentage points, respectively. Without gradient coupling, ADI mIoU decreases by only 0.40 points, whereas recovery drops by 23.55 points, indicating that high image overlap alone does not ensure accurate process identification and that regional coordination improves candidate discrimination. Removing difference preservation lowers recovery by 12.67 points, underscoring its importance for process identification.

Table 4: Component and mechanism ablations. $\mathrm{Acc}_p \uparrow$ is exact process-parameter recovery (%).

	Forward Evolution			**Process Planning**	
Variant	Mask	Resist Image	ADI	Acc_p	Target ADI
w/o PC & SE	92.83 / 2.47	93.15 / 1.90	78.76 / 8.54	37.23	68.36 / 11.92
w/o SE	94.82 / 1.61	95.11 / 1.47	81.96 / 6.02	46.39	70.17 / 7.50
w/o PC	95.28 / 1.47	95.84 / 1.20	82.29 / 5.53	46.63	70.42 / 7.39
w/o gradient coupling	97.01 / 0.73	97.84 / 0.63	84.68 / 3.97	54.74	75.33 / 5.89
w/o diff. preservation	97.24 / 0.69	98.06 / 0.52	84.88 / 3.91	65.62	78.36 / 4.72
Flow-Litho (ours)	**97.95 / 0.42**	**98.44 / 0.37**	**85.08 / 3.67**	**78.29**	**80.41 / 4.08**

5 Conclusion

We present Flow-Litho, a physics-guided world model combining stage-conditioned flow matching, coupled regional velocity correction, and difference-preserving self-evolving optimization. These components integrate approximate physical constraints into latent dynamics while preserving process-induced differences across stages and adapting downstream transitions to upstream rollout deviations. Experiments demonstrate improved geometric and observational fidelity across process conditions and technology nodes, alongside more accurate parameter recovery under limited search budgets, supporting unified multi-stage lithography modeling and target-driven process planning.

AI USE STATEMENT

AI tools were used to refine the manuscript's language and check mathematical expressions. The authors are responsible for reviewing and validating all AI-assisted content and take full responsibility for the research decisions, experimental results, and manuscript as a whole.

REFERENCES

Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in Atari. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-1873.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems*, volume 28, 2015.

Guojin Chen, Hao Geng, Bei Yu, and David Z Pan. Open-source differentiable lithography imaging framework. In *DTCO and Computational Patterning III*, volume 12954, pp. 133–142. SPIE, 2024a.

Guojin Chen, Haoyu Yang, Haoxing Mark Ren, Bei Yu, and David Z Pan. Differentiable edge-based OPC. In *Proceedings of the 43rd IEEE/ACM International Conference on Computer-Aided Design*, pp. 1–9, 2024b.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, 2019. doi: 10.18653/v1/N19-1423.

P Evanschitzky, A Erdmann, and T Fühner. Extended abbe approach for fast and accurate lithography imaging simulations. In *25th European Mask and Lithography Conference*, pp. 1–11. VDE, 2009.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640:647–653, 2025. doi: 10.1038/s41586-025-08744-2.

Hongquan He, Zhen Wang, Jingya Wang, Tao Wu, Xuming He, Bei Yu, Jingyi Yu, and Hao Geng. LithoSim: A large, holistic lithography simulation benchmark for AI-driven semiconductor manufacturing. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-5259.

Yuqi Jiang, Xudong Lu, Qian Jin, Qi Sun, Hanming Wu, and Cheng Zhuo. Fabgpt: An efficient large multimodal model for complex wafer defect knowledge queries. In *Proceedings of the 43rd IEEE/ACM International Conference on Computer-Aided Design*, pp. 1–8, 2024.

Yuqi Jiang, Yumeng Liu, Zimu Li, Jinyuan Deng, Qian Jin, Yucheng Cui, Yu Li, Xunzhao Yin, Qi Sun, and Cheng Zhuo. LithoDreamer: A physics-informed world model for multi-stage computational lithography. *arXiv preprint arXiv:2606.26713*, 2026.

Qian Jin, Yumeng Liu, Yuqi Jiang, Qi Sun, and Cheng Zhuo. Unitho: A unified multi-task framework for computational lithography. *arXiv preprint arXiv:2511.10255*, 2025.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Qiang Liu, Mengyu Chu, and Nils Thuerey. ConFIG: Towards conflict-free training of physics informed neural networks. In *International Conference on Learning Representations*, 2025.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Xiaodong Meng, Cai Chen, and Jie Ni. Inverse lithography technology (ILT) under chip manufacture context. *Micromachines*, 17(1):117, 2026. doi: 10.3390/mi17010117.

Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-Net: Fully convolutional neural networks for volumetric medical image segmentation. In *International Conference on 3D Vision*, pp. 565–571, 2016. doi: 10.1109/3DV.2016.79.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

Hao-Chiang Shao, Chao-Yi Peng, Jun-Rei Wu, Chia-Wen Lin, Shao-Yun Fang, Pin-Yen Tsai, and Yan-Hsiu Liu. From IC layout to die photograph: A CNN-based data-driven approach. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 40(5):957–970, 2021. doi: 10.1109/TCAD.2020.3015469.

Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, volume 25, 2012.

Pierre Soille. *Morphological Image Analysis: Principles and Applications*. Springer, 2 edition, 2003.

Young-Han Son, Dong-Hee Shin, Deok-Joong Lee, Hyun Jung Lee, and Tae-Eui Kam. Optical diffraction-based convolution for semiconductor lithography. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12458–12468, 2026.

John S Villarrubia, AE Vladár, Bin Ming, Regis J Kline, Daniel F Sunday, JS Chawla, and Scott List. Scanning electron microscope measurement of width and shape of 10 nm patterned lines using a jmonsel-modeled library. *Ultramicroscopy*, 154:15–28, 2015.

Zhen Wang, Hongquan He, Tao Wu, Xuming He, Qi Sun, Cheng Zhuo, Bei Yu, Jingyi Yu, and Hao Geng. LMLitho: A large vision model-driven lithography simulation framework. In *IEEE/ACM International Conference on Computer-Aided Design*, 2025a. doi: 10.1109/ICCAD66269.2025.11240940.

Zixiao Wang, Jieya Zhou, Su Zheng, Shuo Yin, Kaichao Liang, Shoubo Hu, Xiao Chen, and Bei Yu. TorchResist: Open-source differentiable resist simulator. In *DTCO and Computational Patterning IV*, volume 13425, pp. 397–405. SPIE, 2025b.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. DINO-WM: World models on pre-trained visual features enable zero-shot planning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 79115–79135. PMLR, 2025.

APPENDIX

The appendix presents evaluation protocols, metric definitions, supplementary experiments, physical-operator calibration, and method derivations.

A **Evaluation Protocols** . . . 12
A.1 Process-Conditioned Forward Evolution . . . 12
A.2 Target-Driven Process-Parameter Planning . . . 13
A.3 Fixed-Process Trajectory Planning . . . 14
B **Evaluation Metrics** . . . 14
C **Supplementary Experiments** . . . 16
C.1 Supplementary Evaluation of Forward Evolution . . . 16
C.2 Supplementary Evaluation of Process Planning . . . 16
C.3 Fixed-Process Trajectory Planning . . . 20
C.4 Additional Ablation Studies . . . 21
D **Approximate Physical Operators** . . . 23
D.1 Patterning Operator and Source Discretization . . . 23
D.2 Low-Frequency Observation Operator . . . 24
D.3 Offline Calibration and Numerical Evaluation . . . 24
E **Method Derivation and Numerical Details** . . . 25
E.1 Regional Discrepancies and Tolerance Calibration . . . 25
E.2 Coupled Correction and Magnitude Control . . . 26
E.3 First-Order Training Gradients . . . 26
E.4 Recomposition and Replay Gradients . . . 27

A EVALUATION PROTOCOLS

We distinguish process-conditioned forward evolution, target-driven process-parameter planning, and goal-conditioned trajectory planning with fixed process parameters. Data splits follow the experimental settings in the main text. Network parameters and physical-operator coefficients remain fixed at test time; intermediate labels are used only for evaluation.

A.1 PROCESS-CONDITIONED FORWARD EVOLUTION

Given $X^{(0)}$ and p, the model recursively evolves subsequent states:

$$\widehat{X}^{(0)} = X^{(0)}, \qquad \widehat{X}^{(k)} = F_\theta^{(k)}\left(\widehat{X}^{(k-1)}, p\right), \quad k = 1, 2, 3. \tag{A.1}$$

Flow-Litho uses three Euler steps per stage with physical velocity correction at every step; p remains unchanged throughout the trajectory. Independent stage evaluation with ground-truth upstream inputs is denoted by **GT**, whereas recursive evaluation with model-evolved upstream states is denoted by **R**.

We evaluate each stage using mPA, mIoU, edge F1, MSE, and EPE, as defined in Appendix B. Table 1 reports mIoU, MSE, and EPE; supplementary metrics and qualitative comparisons appear in Appendix C.1. Evaluation on the 28 nm dataset is restricted to Mask and Resist Image.

A.2 TARGET-DRIVEN PROCESS-PARAMETER PLANNING

Given $X^{(0)}$ and a target $X^{(3)}_{\mathrm{tar}}$, this task selects process parameters through model-based target reconstruction. Evaluation uses held-out layout-target ADI pairs, disjoint from training and validation. Each pair defines a query, and the candidate domain comprises the 36 process combinations in Section 2.2. With network and physical-operator parameters fixed, a complete recursive rollout defines the candidate ADI and its geometric objective:

$$\begin{aligned} \widehat{X}^{(3)}(p) &= F^{(3)}_\theta\Big(F^{(2)}_\theta\Big(F^{(1)}_\theta(X^{(0)},p),p\Big),p\Big), \\ J(p) &= \frac{\mathrm{EPE}_{\mathrm{plan}}\Big(\widehat{X}^{(3)}(p),X^{(3)}_{\mathrm{tar}}\Big)}{e_0} + \lambda_{\mathrm{plan}}\Big[1-\mathrm{mIoU}\Big(\widehat{X}^{(3)}(p),X^{(3)}_{\mathrm{tar}}\Big)\Big], \end{aligned} \tag{A.2}$$

where $\mathrm{EPE}_{\mathrm{plan}}$ is the contour discrepancy with unmatched-gauge penalties defined in Appendix B. We fix $e_0 = 1\,\mathrm{nm}$ and $\lambda_{\mathrm{plan}} = 10$ across models, search strategies, and budgets, with mIoU expressed in $[0,1]$. Selection is restricted to evaluated candidates in the prescribed domain $\mathcal{P}$:

$$\widehat{p} = \arg\min_{p\in\mathcal{P}_{\mathrm{eval}}} J(p), \tag{A.3}$$

where $\mathcal{P}_{\mathrm{eval}} \subseteq \mathcal{P}$ contains the evaluated conditions. The planner receives the layout, target ADI, and candidate process vectors; the target-associated process label $p^\star$ and all other condition-indexed image labels are withheld. Comparisons use lithography-adapted models with identical process inputs and three-stage outputs, each evaluated through its own recursive mapping.

Component	**Configuration**
Search domain	The 36 observed process combinations.
Search method	GP-BO (Snoek et al., 2012); expected improvement selects among unevaluated candidates.
GP representation	Min–max-scaled numerical inputs, one-hot source categories, and standardized scores; a constant mean and Matérn-$5/2$ ARD kernel.
GP estimation	Marginal-likelihood fitting by L-BFGS-B, 100 iterations; fixed noise variance 10^{-4} in standardized score units and numerical jitter 10^{-6}.
Rollout budget	12 distinct conditions: 4 shared random initial candidates and 8 sequential acquisitions.
Search comparisons	Random search shares the initial candidates and uniformly samples the remaining conditions; exhaustive search evaluates all 36.
Randomness	Seeds 42, 43, 44; shared initial candidates and matched rollout seeds for stochastic models.

The four initial evaluations provide process-score pairs for a GP posterior over J. Subsequent acquisitions maximize expected improvement:

$$p_{\mathrm{next}} = \arg\max_{p\in\mathcal{P}\setminus\mathcal{P}_{\mathrm{eval}}} \mathbb{E}_{\mathrm{GP}}[\max\{J_{\min} - J(p), 0\}], \tag{A.4}$$

where $J_{\min} = \min_{q\in\mathcal{P}_{\mathrm{eval}}} J(q)$ and the expectation is taken under the GP posterior. Acquisition uses both the estimated score and its uncertainty. Only the selected candidate incurs a model rollout; its resulting score updates the GP before the next acquisition. The final condition minimizes the evaluated scores in Equation (A.3).

One rollout includes the complete three-stage chain and model-specific correction. GP kernel length scales and output variance start at 1; the kernel and mean are refitted to available scores after each acquisition. Within-query caching counts each condition once, and fixed candidate ordering resolves ties. Stochastic-model seeds depend on the query and condition, making scores independent of evaluation order. These settings are shared across searches and model comparisons.

Evaluation distinguishes parameter recovery, target agreement, and process-response agreement. Target agreement compares $\widehat{X}^{(3)}(\widehat{p})$ with $X^{(3)}_{\mathrm{tar}}$; process-response agreement compares it with $\widehat{X}^{(3)}(p^\star)$. For the latter, the reference rollout uses the selected rollout's stochastic inputs and numerical settings, isolating the effect of condition selection within each model. The annotated condition is used only after selection; this reference rollout is excluded from the search budget and never

informs acquisition. Both image comparisons use mIoU and EPE, with exact recovery defined in Appendix B.

All predefined queries are retained, irrespective of recovery success or model score. Results are averaged over targets within each layout, then over layouts and seeds. Confidence intervals use 1,000 bootstrap resamples of layouts, retaining their associated queries. Budgets count distinct complete candidate rollouts, including initial evaluations; post-selection reference rollouts serve evaluation only.

A.3 Goal-Conditioned Trajectory Planning with Fixed Process Parameters

This task evaluates target-conditioned trajectory control given $X^{(0)}$, p, and $X^{(3)}_{\mathrm{tar}}$. Process conditions and model parameters remain fixed, while stage-wise interventions modify the base velocity:

$$\bar{v}_n^{(k)} = v_\theta^{(k)}\left(\widehat{u}_{t_n}^{(k)}, t_n\right) + \xi_n^{(k)}, \tag{A.5}$$

where $\xi_n^{(k)}$ is the intervention at step n of stage k. Recomputing physical correction from $\bar{v}_n^{(k)}$ and integrating the resulting velocity yields $\widehat{X}^{(3)}_\xi$. The intervention sequence $\xi = \{\xi_n^{(k)}\}_{k,n}$ balances target agreement and intervention magnitude:

$$\xi^\star = \arg\min_\xi \left[\ell\left(\widehat{X}^{(3)}_\xi, X^{(3)}_{\mathrm{tar}}\right) + \lambda_\xi \sum_{k=1}^{3} \sum_{n=0}^{K_k-1} \|\xi_n^{(k)}\|_2^2 \right], \tag{A.6}$$

where λ_ξ controls regularization. Each intervention is a zero-initialized $4 \times 64 \times 64$ tensor. Optimization uses Adam for 50 iterations with learning rate 10^{-3}, moment coefficients $(0.9, 0.999)$, $\lambda_\xi = 10^{-4}$, and gradient-norm clipping at 1.0. All stages retain three Euler steps and the first-order correction-gradient convention in Section 3.2.2.

Stage-wise mIoU and EPE assess intermediate-state consistency and terminal target agreement. For the 28 nm dataset, Resist Image replaces ADI as the target, and both the rollout and intervention regularization terminate at stage two. The experiment thus measures the controllability of learned trajectories under fixed process conditions.

B Evaluation Metrics

Evaluation combines pixel agreement, contour fidelity, and lithography-rule consistency under common registration and valid regions. Intensity metrics use images in $[0, 1]$, and geometric distances are converted to nanometers using each dataset's pixel size. Mask and Resist Image foregrounds are defined at threshold 0.5. ADI geometry uses Gaussian smoothing with standard deviation 1 pixel and threshold 0.38, with darker regions as foreground. Marching squares with linear interpolation extract the corresponding contours. These operations are identical for model outputs and targets. Metrics are averaged per image, with planning aggregation following Appendix A.2.

(1) Mean pixel accuracy (mPA). Mean pixel accuracy equally weights foreground and background classification:

$$\mathrm{mPA} = \frac{1}{2}\left(\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} + \frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}}\right), \tag{B.1}$$

where TP, TN, FP, and FN are the true-positive, true-negative, false-positive, and false-negative pixel counts within the valid region. Results are reported as percentages; higher values indicate better classification accuracy.

(2) Mean intersection over union (mIoU). Region agreement is measured by the mean foreground and background intersection over union:

$$\mathrm{mIoU} = \frac{1}{2}\left(\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}+\mathrm{FN}} + \frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}+\mathrm{FN}}\right). \tag{B.2}$$

Higher percentages indicate better region agreement. Classes with zero denominators are excluded from the corresponding averages in Equation (B.1)- Equation (B.2). With matched pixel sets and class aggregation, $\mathrm{mIoU} \le \mathrm{mPA}$.

(3) Edge F1 score. Boundary precision $\mathrm{Prec}_{\mathrm{edge}}$ and recall $\mathrm{Rec}_{\mathrm{edge}}$ measure the fractions of evolved and target contour points within a matching distance of 1 pixel from the opposite contour. Their harmonic mean is:

$$\mathrm{F1}_{\mathrm{edge}} = \frac{2\,\mathrm{Prec}_{\mathrm{edge}}\mathrm{Rec}_{\mathrm{edge}}}{\mathrm{Prec}_{\mathrm{edge}} + \mathrm{Rec}_{\mathrm{edge}}}. \tag{B.3}$$

Results are reported as $100\,\mathrm{F1}_{\mathrm{edge}}$. Both empty contours receive 100%; a single empty contour or absent matches receives 0%.

(4) Mean squared error (MSE). Intensity fidelity is measured before evaluation binarization:

$$\mathrm{MSE} = \frac{1}{|\Omega|}\sum_{x\in\Omega}\left(\widehat{X}^{(k)}(x) - X^{(k)}(x)\right)^2, \qquad \Omega = \bigcup_{j=1}^{4}\Omega_j. \tag{B.4}$$

Lower values indicate smaller intensity deviations. Table 1 reports MSE in units of 10^{-3}.

(5) Edge placement error (EPE). Target-contour gauges are sampled uniformly at 1-pixel intervals. Each gauge is matched to the nearest intersection between the evolved contour and the line through the gauge along the target normal. The mean absolute displacement in physical units is:

$$\mathrm{EPE}_{\mathrm{avg}} = \frac{1}{N_{\mathrm{valid}}}\sum_{i=1}^{N_{\mathrm{valid}}}|\mathrm{EPE}_i|, \tag{B.5}$$

where EPE_i is the signed displacement in nanometers and N_{valid} counts valid matches. Unmatched gauges are excluded, and their fraction characterizes matching completeness; EPE is undefined when no match exists. Common contour interpolation supports subpixel displacement estimates, and lower EPE indicates greater placement fidelity.

For candidate selection, $\mathrm{EPE}_{\mathrm{plan}}$ averages over all target gauges, assigning the physical image-diagonal distance to gauges with no intersection on the evolved contour. When no target contour is extracted, the score is zero if neither image yields a contour and the image-diagonal distance otherwise. This finite scoring convention applies uniformly to all planners; reported EPE retains Equation (B.5).

(6) Lithography rule-check (LRC) violations. Three violation counts are compared with commercial simulation under identical process conditions, using the input layout as the design reference. For a nominal layout linewidth w_0, **#Pinch** records printed linewidths below $0.95w_0$, and **#Bridge** records printed spacings below $0.95w_0$. **#EPE** counts locations where the absolute contour-to-layout displacement reaches or exceeds 2 nm along straight edges or 4 nm in corner regions. Counts are returned by the shared rule deck with identical contour extraction, gauge sampling, region classification, and marker-reporting settings. Mean EPE measures displacement magnitude, whereas #EPE counts violations of the region-specific tolerances. Table 2 evaluates count agreement under the process conditions in Table B.1.

Table B.1: Process conditions used for the LRC comparison in Table 2.

Condition	Source	Threshold	Defocus (nm)	Relative dose
A	Bull's eye	0.09231251	0	1.2
B	Bull's eye	0.1436665	0	1.2
C	Annular	0.09231251	50	1.0
D	Circular	0.1436665	0	1.0
E	Circular	0.09231251	0	1.0

(7) Relative count discrepancy. For each violation category, percentage subscripts in Table 2 report signed deviations from commercial simulation; *Error* averages their absolute values across five conditions:

$$\Delta_i = 100\frac{N_i - N_i^{\mathrm{ref}}}{N_i^{\mathrm{ref}}}, \qquad \mathrm{Error} = \frac{1}{5}\sum_{i=1}^{5}|\Delta_i|, \tag{B.6}$$

where N_i and N_i^{ref} are model-derived and commercial-reference counts. Positive and negative deviations indicate overestimation and underestimation. All reference counts are nonzero, and lower *Error* indicates closer agreement across conditions.

(8) Planning image agreement. Table 3 reports evolved-target and within-model process-response agreement as defined in Appendix A.2. Both use identical preprocessing and metric extraction; process-response agreement quantifies condition sensitivity within each fixed model.

(9) Exact process recovery and candidate coverage. For the 36-condition benchmark, recovery requires matching all four process values:

$$\begin{aligned} \mathrm{Acc}_p &= \frac{100}{N_{\mathrm{q}}} \sum_{i=1}^{N_{\mathrm{q}}} \mathbf{1}[\widehat{p}_i = p_i^\star], \\ \mathrm{Cov}_p &= \frac{100}{N_{\mathrm{q}}} \sum_{i=1}^{N_{\mathrm{q}}} \mathbf{1}[p_i^\star \in \mathcal{P}_{\mathrm{eval},i}], \end{aligned} \tag{B.7}$$

where N_{q} counts queries per layout and seed, and $\mathcal{P}_{\mathrm{eval},i}$ contains the evaluated conditions for query i. Aggregation follows Appendix A.2. Selection from evaluated candidates gives $\mathrm{Acc}_p \leq \mathrm{Cov}_p$; exhaustive coverage is 100%, irrespective of recovery accuracy.

C Supplementary Experiments

C.1 Supplementary Evaluation of Forward Evolution

Table C.1 supplements the forward-evolution evaluation with pixel-classification and boundary-matching metrics. Compared with LithoDreamer, Flow-Litho improves ADI mPA and edge F1 by 3.81 and 3.31 percentage points on ID data, indicating gains in both pixel classification and boundary reconstruction. The ADI mPA advantage increases to 5.99 points under unseen 55 nm process conditions. On 28 nm cross-node data, Mask and Resist Image edge F1 improve by 2.83 and 2.90 points, respectively, consistent with the EPE reductions in Table 1 and supporting improved contour fidelity across technology nodes.

Figure C.1 reveals that the three Euler updates within each stage preserve the principal layout structure while concentrating geometric changes near line ends and corners. Local Mask extensions are followed by smoother Resist Image contours, whereas ADI updates primarily adjust grayscale appearance around the resulting geometry. These distinct changes illustrate how the shared velocity model accommodates stage-specific geometric and observational responses within short rollouts. Across both layouts, corresponding features remain spatially aligned through successive stages, qualitatively supporting coherent propagation of pattern structure.

C.2 Supplementary Evaluation of Process Planning

Search-budget sensitivity. Table C.2 examines how search budget affects parameter recovery and target agreement, following Appendix A.2. At 12 rollouts, GP-BO outperforms random search in exact recovery by 50.16 percentage points, indicating more effective candidate selection under a limited budget. Increasing the GP-BO budget from 8 to 12 rollouts improves recovery by 25.06 points, whereas doubling it from 12 to 24 adds only 3.04 points, revealing diminishing recovery gains. At 36 distinct evaluations, both methods exhaust the candidate set and obtain identical results. However, 100% coverage yields only 84.02% recovery, showing that complete exploration does not ensure that the target-matching objective selects the annotated combination. Candidate coverage and final selection accuracy therefore capture distinct aspects of planning performance.

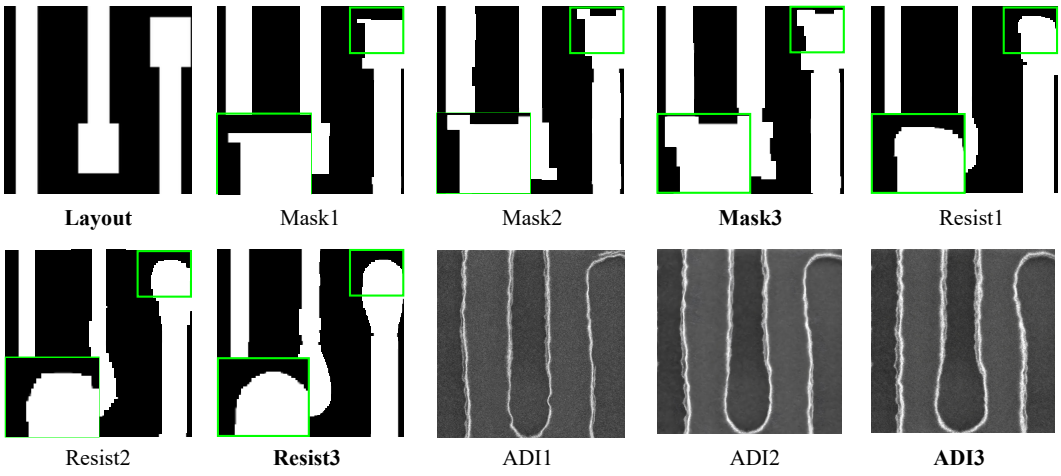
Planning objective. All criteria are evaluated on the same test queries using 12-rollout GP-BO with four shared random initial candidates and eight subsequent acquisitions. Each criterion determines both acquisition and final selection. The combined score follows Equation (A.2), with $\lambda_{\mathrm{plan}} \in \{1, 10, 100\}$ and a default of 10. The contour-only criterion corresponds to $\lambda_{\mathrm{plan}} = 0$.

Table C.1: Supplementary multi-stage forward-evolution results under ID and OOD conditions (mPA and edge F1: %; higher is better). **Best** and second-best values are highlighted.

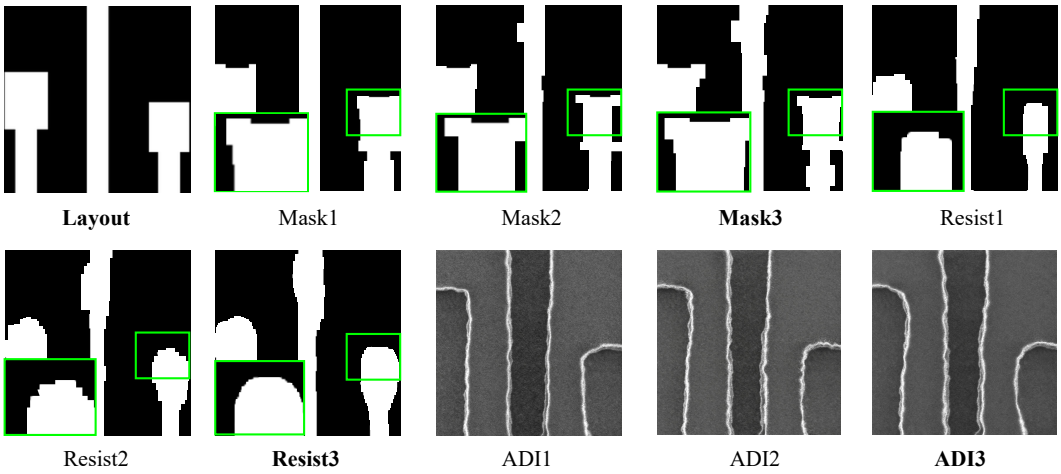
Method	Input	Mask		Resist Image		ADI	
		mPA↑	F1↑	mPA↑	F1↑	mPA↑	F1↑
55 nm: in-domain							
LithoNet (Shao et al., 2021)	GT	93.54	84.29	91.41	74.36	61.17	26.78
LMLitho (Wang et al., 2025a)	GT	96.27	95.99	95.23	96.24	52.56	32.14
Unitho (Jin et al., 2025)	GT	95.71	96.42	95.52	96.02	n/a	n/a
OptiCo (Son et al., 2026)	GT	94.52	93.90	93.89	94.26	n/a	n/a
DIAMOND (Alonso et al., 2024)	R	89.97	86.20	92.82	88.36	71.56	64.21
DINO-WM (Zhou et al., 2025)	R	93.47	93.00	95.68	95.03	73.57	69.99
DreamerV3 (Hafner et al., 2025)	R	95.32	94.97	97.25	96.50	80.76	77.83
LithoDreamer (Jiang et al., 2026)	R	98.69	99.66	99.01	99.73	82.56	80.91
Flow-Litho (ours)	R	**98.84**	**99.69**	**99.45**	**99.79**	**86.37**	**84.22**
55 nm: unseen process condition							
LithoNet (Shao et al., 2021)	GT	91.43	83.81	89.55	76.35	72.24	58.06
LMLitho (Wang et al., 2025a)	GT	90.69	78.36	93.29	84.19	47.45	43.15
Unitho (Jin et al., 2025)	GT	91.01	86.92	94.80	89.28	n/a	n/a
OptiCo (Son et al., 2026)	GT	92.86	90.93	95.78	93.67	n/a	n/a
DIAMOND (Alonso et al., 2024)	R	84.72	78.39	86.04	80.19	68.30	63.96
DINO-WM (Zhou et al., 2025)	R	89.87	84.28	93.51	86.06	71.81	62.20
DreamerV3 (Hafner et al., 2025)	R	92.48	86.53	93.71	88.65	74.74	69.67
LithoDreamer (Jiang et al., 2026)	R	96.92	90.51	98.27	91.33	78.89	75.29
Flow-Litho (ours)	R	**97.84**	**93.69**	**98.85**	**95.41**	**84.88**	**78.07**
28 nm: cross-node transfer							
LithoNet (Shao et al., 2021)	GT	93.06	84.92	93.06	86.39	n/a	n/a
LMLitho (Wang et al., 2025a)	GT	82.49	79.10	92.46	72.31	n/a	n/a
Unitho (Jin et al., 2025)	GT	81.20	73.46	91.71	69.58	n/a	n/a
OptiCo (Son et al., 2026)	GT	80.49	72.56	88.13	84.36	n/a	n/a
DIAMOND (Alonso et al., 2024)	R	71.55	67.70	72.49	70.81	n/a	n/a
DINO-WM (Zhou et al., 2025)	R	73.90	70.74	75.81	74.93	n/a	n/a
DreamerV3 (Hafner et al., 2025)	R	81.66	78.04	85.52	81.83	n/a	n/a
LithoDreamer (Jiang et al., 2026)	R	94.92	87.39	96.99	90.83	n/a	n/a
Flow-Litho (ours)	R	**95.49**	**90.22**	**97.65**	**93.73**	n/a	n/a

Table C.2: Search-budget sensitivity over 36 process combinations with fixed Flow-Litho and scoring function. Nested budgets count complete three-stage rollouts, including four shared random initial evaluations. Acc_p (%) and Cov_p (%) report exact four-parameter recovery and the proportion of queries whose evaluated candidates include the annotated combination, respectively. mIoU (%) and EPE (nm) compare the selected rollout ADI with the target.

Rollouts	$\mathrm{Acc}_p \uparrow$ (%)	$\mathrm{Cov}_p \uparrow$ (%)	mIoU↑ (%)	EPE↓ (nm)
GP-BO				
4	8.07	10.41	71.39	7.39
8	53.23	58.41	79.85	4.26
12	78.29	86.32	80.41	4.08
24	81.33	90.69	82.15	4.06
36	**84.02**	**100.00**	**83.76**	**3.37**
Random search				
4	8.07	10.41	71.39	7.39
8	17.57	22.00	75.98	6.47
12	28.13	37.71	78.21	4.95
24	57.79	67.62	80.95	4.21
36	**84.02**	**100.00**	**83.76**	**3.37**



(a) Layout example 1



(b) Layout example 2

Figure C.1: Three-step Euler rollouts of Flow-Litho for two representative layouts. Within each stage, indices 1-3 denote decoded states after successive integration steps, with step 3 yielding the stage output. Green boxes highlight local contour changes in the Mask and Resist Image.

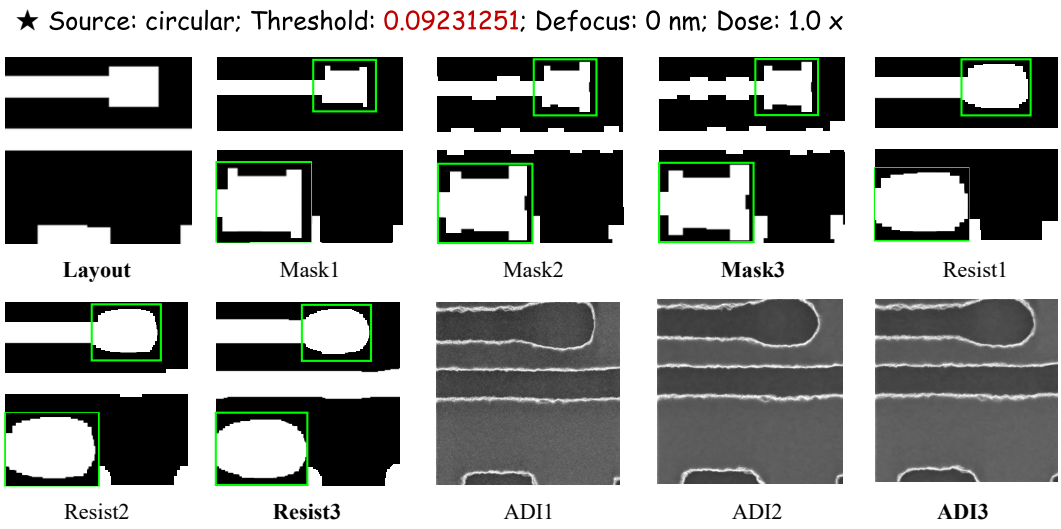
Table C.3 shows that the default combined score improves exact parameter recovery by 6.39 percentage points and reduces target EPE by 23.3% relative to the contour-only criterion, supporting complementary roles for region overlap and contour placement in process identification. The default weight performs best among the tested values, whereas further increasing the region weight degrades both recovery and target agreement. These results favor balancing regional and contour discrepancies for candidate selection under a limited rollout budget.

Qualitative assessment. Figure C.2 compares rollouts under the target-associated condition and an alternative resist threshold for the same layout. Local Mask compensation changes together with the resulting Resist Image contours and ADI boundary appearance, while the overall pattern arrangement remains similar. The persistence of localized differences across stages illustrates how intermediate geometric responses distinguish candidate conditions whose terminal images are globally similar, motivating contour-sensitive target matching.

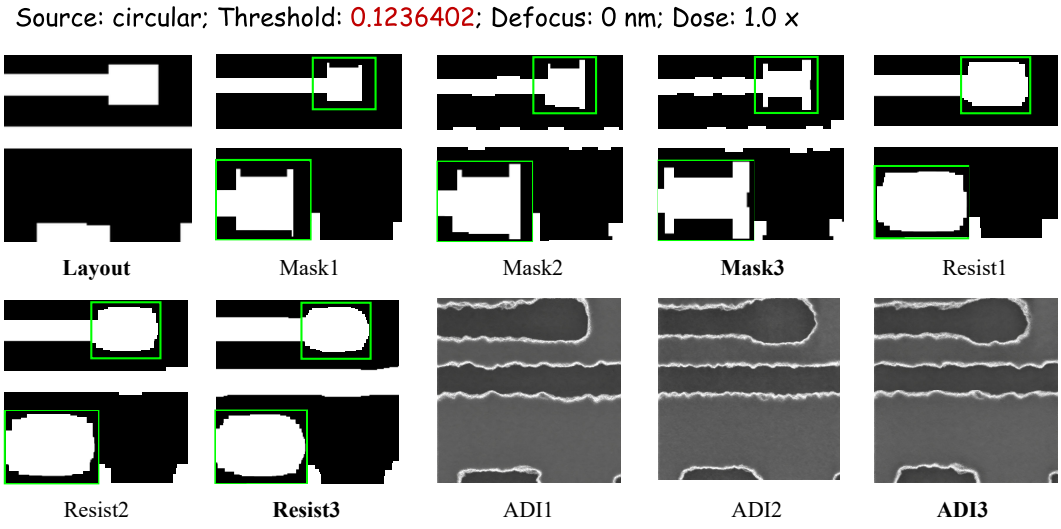
Figure C.3 examines an alternative dose for a second layout. The narrow feature exhibits changes in extent and rounding that remain discernible in the ADI, linking condition-dependent intermediate geometry to the terminal response. These paired trajectories illustrate the local response information available for process planning: region and contour agreement can distinguish candidates even when the layout and overall pattern structure are unchanged.

Table C.3: Scoring-criterion ablation with fixed Flow-Litho and 12-rollout GP-BO. λ_{plan} weights the region term, and Acc_p denotes exact four-parameter recovery (%). Target and process-response agreement report mIoU (%)/EPE (nm) against the target ADI and the same model's rollout under annotated parameters, respectively.

Scoring criterion	λ_{plan}	$\text{Acc}_p \uparrow$ (%)	Target agreement	Process response
Individual criteria				
MAE ℓ	–	52.14	73.97 / 7.53	85.83 / 4.29
Region discrepancy $1 - \text{mIoU}$	–	65.31	76.87 / 6.94	91.06 / 2.57
Contour discrepancy $\text{EPE}_{\text{plan}}/e_0$	0	71.90	78.38 / 5.32	92.04 / 2.43
Combined contour-region criterion				
Combined score	1	72.86	79.25 / 4.71	93.63 / 2.10
Combined score (default)	10	**78.29**	**80.41** / **4.08**	**94.93** / **1.89**
Combined score	100	68.26	77.95 / 6.77	92.28 / 2.23

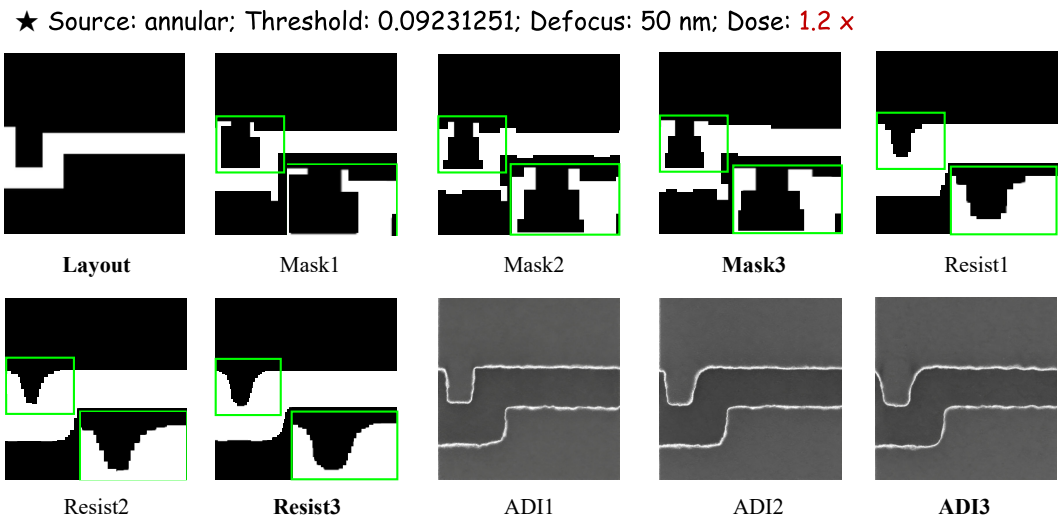


(a) $T = 0.09231251$

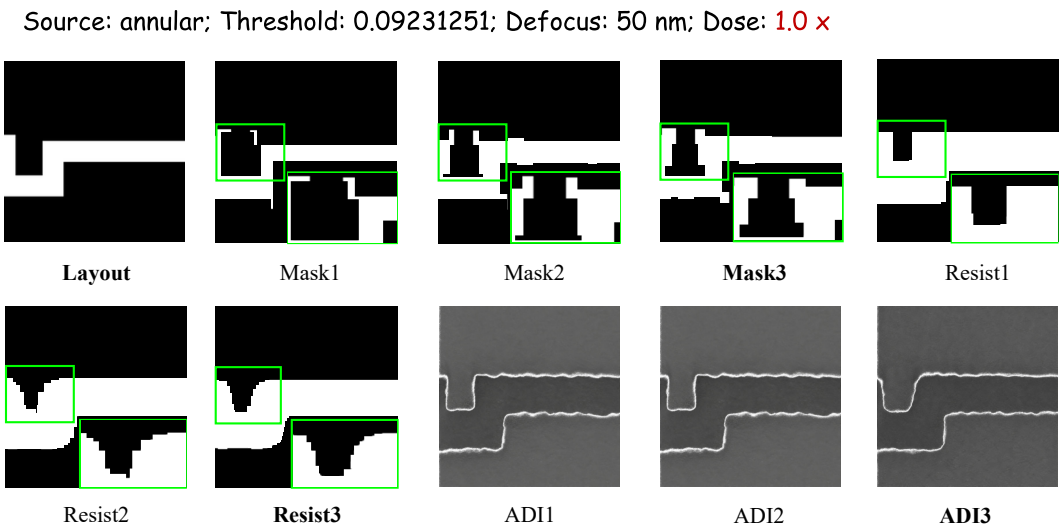


(b) $T = 0.1236402$

Figure C.2: Candidate rollouts under two resist thresholds for the same layout ($d = 1.0$, $f = 0$ nm). The star marks the annotated condition associated with the target ADI. Indices 1-3 denote Euler steps within each stage, and green boxes highlight local geometric responses.



(a) $d = 1.2$



(b) $d = 1.0$

Figure C.3: Candidate rollouts under two exposure doses for the same layout ($T = 0.09231251$, $f = 50$ nm). The star marks the annotated condition associated with the target ADI. Indices 1–3 denote Euler steps within each stage, and green boxes highlight local geometric responses.

C.3 Fixed-Process Trajectory Planning

Evaluation follows Appendix A.3. Terminal targets are ADI for 55 nm and Resist Image for 28 nm; intermediate states are evaluated against their paired labels. The comparisons include DIAMOND (Alonso et al., 2024), DINO-WM (Zhou et al., 2025), DreamerV3 (Hafner et al., 2025), and LithoDreamer (Jiang et al., 2026). DINO-WM and LithoDreamer results are reproduced from Jiang et al. (2026).

Figure C.4 compares goal-conditioned trajectory planning with fixed process parameters. Relative to LithoDreamer, Flow-Litho improves ADI mIoU by 3.48 percentage points on the 55 nm ID set and Mask and Resist Image mIoU by 7.50 and 9.22 points under unseen process conditions, respectively, indicating more accurate reconstruction of intermediate geometry. In 28 nm cross-node transfer, Resist Image EPE decreases by 46.9%, supporting improved contour fidelity across manufacturing nodes. However, ADI mIoU is 2.11 points lower under unseen 55 nm conditions, indicating that improvements in intermediate geometry do not consistently translate into better terminal

observations. These results highlight the benefits of Flow-Litho for geometric trajectory reconstruction under distribution shift, while terminal observation agreement remains condition-dependent.

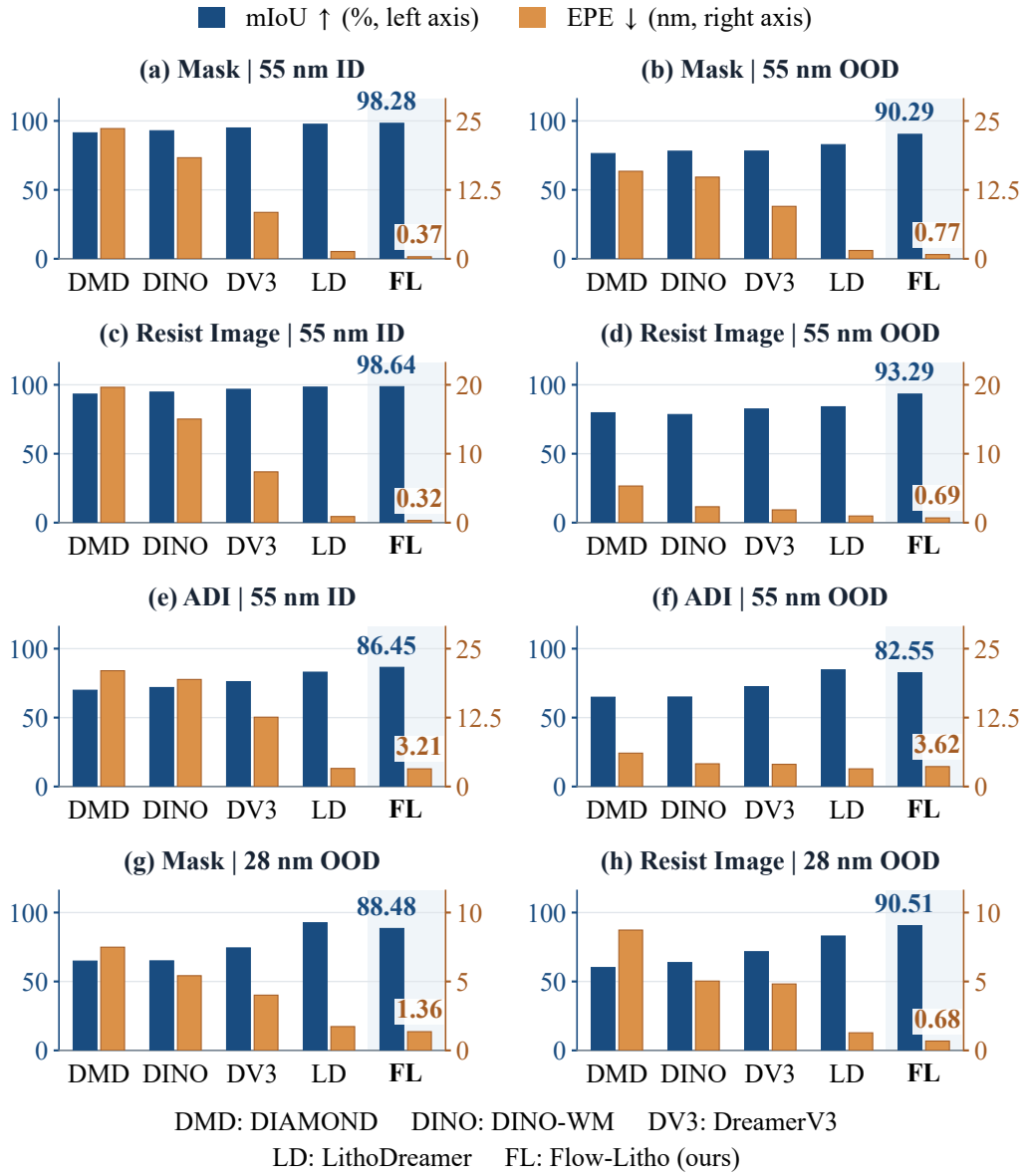


Figure C.4: Stage-wise evaluation of goal-conditioned trajectory planning with fixed process parameters. Panels (a)-(f) compare 55 nm ID and unseen process conditions, while (g)-(h) report 28 nm cross-node results. Dark blue and orange denote mIoU (left axis, %) and EPE (right axis, nm).

C.4 Additional Ablation Studies

These experiments separate the effects of physical guidance, cross-stage learning, and the construction of regional discrepancy. All variants use the data splits, architecture, optimization schedule, and base loss weights in Section 4.1, with three Euler steps per stage. We retain the GT-pair and ordinary cascaded objectives throughout. Forward metrics are evaluated on the 55 nm ID set. Planning uses identical queries and 12-rollout GP-BO with four shared initial candidates, following Appendix A.2. Image entries report mIoU/EPE, and the planning columns measure exact parameter recovery and

agreement between the selected rollout and target ADI, as defined in Appendix B. No ablation changes the planning score or search budget.

For the mechanism controls in Table 4, *w/o gradient coupling* replaces B with its diagonal in Equation (12), retaining the regional gradients, desired reductions, regularization, and norm bound. *w/o difference preservation* randomly permutes $A_{ab}^{(k)}$ within the jointly valid pixels of each quadrant. This preserves the exchange count per quadrant while removing label-consistency gating; paired inputs, branch conditions, replay losses, and loss weights remain unchanged.

C.4.1 Physical Correction versus Loss-Based Supervision

The physical-loss baseline replaces velocity correction with a differentiable training penalty derived from the same response mappings and regional discrepancies:

$$\mathcal{L}_{\text{phys}} = \mathbb{E}_{k,t}\left[\frac{1}{4}\sum_{j=1}^{4} e_j\right], \tag{C.1}$$

where stages are sampled uniformly and $t \sim \mathcal{U}[0,1]$. The penalty is added to the otherwise unchanged objective during both training phases, with its coefficient selected from $\{10^{-3}, 10^{-2}, 10^{-1}, 1\}$ by mean validation mIoU across the three stages. Gradients propagate through the candidate response and frozen physical operators to the base velocity field; inference uses uncorrected velocities. SE is retained in all three variants.

Table C.4 shows that physical-loss supervision improves forward evolution and process planning over the model without physical correction, yet remains less effective than explicit velocity correction. Relative to physical-loss supervision, Flow-Litho improves exact parameter recovery by 19.94 percentage points and reduces target ADI EPE by 31.4%, alongside gains at every rollout stage. These results favor applying physical constraints throughout state evolution over incorporating them solely as a training penalty.

Table C.4: Physical velocity correction versus physical-loss supervision, with self-evolving optimization retained across all configurations.

	Forward Evolution			Process Planning	
Variant	Mask	Resist Image	ADI	Acc_p	Target ADI
w/o PC	95.28 / 1.47	95.84 / 1.20	82.29 / 5.53	46.63	70.42 / 7.39
Physical loss	95.93 / 1.25	96.80 / 0.94	83.42 / 4.47	58.35	74.42 / 5.95
Flow-Litho	**97.95 / 0.42**	**98.44 / 0.37**	**85.08 / 3.67**	**78.29**	**80.41 / 4.08**

C.4.2 Recomposition and Cross-Stage Replay

The recomposition control sets $\widetilde{X}_q^{(k)} = \widehat{X}_q^{(k)}$ while retaining paired sampling, complete downstream replay, and both terms of $\mathcal{L}_{\text{SE}}$. The gradient control instead retains recomposition but stops gradients at the replay input before encoding and physical-reference evaluation. Replay therefore continues to update the downstream mapping without propagating through the recomposed upstream state. All variants retain PC and use identical branch pairs, replay counts, and supervision weights.

Table C.5 shows that removing state recomposition or stopping cross-stage replay gradients lowers exact parameter recovery by 11.83 and 17.98 percentage points, respectively, and reduces forward-evolution accuracy. The former retains full downstream replay, whereas the latter retains recomposed inputs. These comparisons support both exposure to alternative upstream deviations and feedback from downstream responses to upstream dynamics.

Table C.5: State recomposition and cross-stage replay-gradient ablations under matched paired sampling and physical velocity correction.

	Forward Evolution			Process Planning	
Variant	Mask	Resist Image	ADI	Acc_p	Target ADI
w/o recomposition	96.73 / 0.89	97.32 / 0.81	83.56 / 4.03	66.46	74.55 / 7.41
w/o replay gradients	96.38 / 0.95	97.02 / 1.16	82.54 / 5.79	60.31	71.59 / 7.83
Flow-Litho	**97.95 / 0.42**	**98.44 / 0.37**	**85.08 / 3.67**	**78.29**	**80.41 / 4.08**

C.4.3 Regional Discrepancy Construction

We compare the four-quadrant formulation in Equation (9) with a single region spanning all valid pixels, and separately remove the boundary or region-wide term. The boundary-only and region-only variants use $(w_b, w_a) = (1, 0)$ and $(0, 1)$, respectively; the complete model uses $(0.5, 0.5)$. For r regions, the solve uses r gradients and $\lambda = 10^{-3}\,\mathrm{tr}(B)/r + 10^{-8}$, recovering the original definition when $r = 4$. Each variant recalibrates its discrepancy tolerances on the same GT training pairs using the same 90th-percentile rule. All other correction settings and SE remain fixed.

Table C.6 shows that replacing four-quadrant correction with a single region lowers exact parameter recovery by 23.30 percentage points, supporting explicit modeling of local responses. The boundary-only variant yields lower Mask and Resist Image EPE, whereas the region-only variant achieves better ADI agreement and parameter recovery. Combining both terms improves recovery by 9.65 percentage points over the stronger individual term and improves mIoU and EPE at all three stages, supporting complementary roles for boundary placement and regional structure.

Table C.6: Ablations of spatial partitioning and boundary and region response terms, with self-evolving optimization retained.

	Forward Evolution			Process Planning	
Variant	Mask	Resist Image	ADI	Acc_p	Target ADI
Single region	96.56 / 0.73	96.82 / 0.70	82.61 / 4.73	54.99	72.67 / 5.91
Boundary only	96.79 / 0.52	97.41 / 0.47	83.58 / 4.33	62.49	74.52 / 5.35
Region only	96.32 / 0.58	97.07 / 0.51	83.96 / 4.05	68.64	76.68 / 4.75
Flow-Litho	**97.95 / 0.42**	**98.44 / 0.37**	**85.08 / 3.67**	**78.29**	**80.41 / 4.08**

D Implementation and Calibration of Approximate Physical Operators

This appendix specifies the discretization, calibration, and numerical evaluation of the patterning operator P and observation operator G in Section 2.1. Their coefficients are estimated offline using the training split and remain fixed during world-model training and inference.

D.1 Patterning Operator and Source Discretization

Images retain their native 512×512 resolution and are normalized to $[0, 1]$, with brighter mask regions corresponding to higher amplitude transmittance. During rollouts, D supplies candidate images under the image-latent convention in Appendix E, ensuring the same intensity range as the calibration data. Process parameters retain their annotated scales; in particular, f takes its numerical value in nanometers without further normalization. The effective pupil is defined as:

$$h_f(\nu) = \mathbf{1}_{\|\nu\| \le \nu_c} \exp\!\left(-i\pi b f \|\nu\|^2\right), \tag{D.1}$$

where ν_c is the cutoff frequency, b is the effective defocus coefficient, and i is the imaginary unit. The indicator restricts the pupil's frequency support; both ν and a are expressed in cycles per pixel. Table D.1 lists the calibrated coefficients.

Each source image yields a distinct set of weights $S_s(q_j)$. For the 256×256 source images, radial median intensities are computed about the common pixel center $(121.5, 124.5)$ and thresholded at

Table D.1: Fixed coefficients of the approximate patterning operator.

Coefficient	Value	Role
ν_c	0.0125	Effective cutoff frequency
a	0.0075	Illumination-induced frequency shift
b	32	Effective defocus phase coefficient
c	2.175	Resist-threshold scaling
τ	0.05	Fixed soft-threshold temperature

0.58 to estimate the emitting support. The resulting radial support function is linearly interpolated onto a uniform 17×17 grid over $[-1, 1]^2$, including its endpoints, using a normalization radius of 113.5 pixels. Weights outside the unit disk are set to zero, and the remaining weights are normalized to unit sum. The circular, bull's-eye, and annular sources contain 145, 101, and 88 nonzero samples, respectively. This construction approximates source geometry while separating illumination shape from exposure-dose scaling.

Equation (2) is evaluated by summing the weighted squared magnitudes of the source-wise propagated fields, followed by dose scaling and the soft-threshold response. Each mask is zero-padded by 256 pixels on every side, evaluated on a 1024×1024 Fourier grid, and cropped to its original field of view. The forward transform is unnormalized, whereas the inverse transform is normalized by the number of grid points. In the mask stage, $Y_{\mathrm{ref}}^{(1)} = X^{(0)}$ constrains the candidate mask's patterning response; in the resist stage, $Y_{\mathrm{ref}}^{(2)} = P(X^{(1)}; p)$ directly provides the reference. Thus, the two stages apply the same response operator to inverse compensation and forward patterning, respectively, without an additional reference-mask optimization.

D.2 Low-Frequency Observation Operator

The observation prior represents the low-frequency correspondence between simulated resist geometry and measured ADI appearance through region and edge responses:

$$\begin{aligned} R(X^{(2)}) &= K_2 * X^{(2)}, \\ H(X^{(2)}) &= K_1 * \tanh\left(\frac{\|\nabla X^{(2)}\|}{0.25}\right), \\ Q(X^{(3)}) &= K_1 * X^{(3)}, \end{aligned} \tag{D.2}$$

where $*$ denotes two-dimensional convolution, and K_σ is a unit-sum Gaussian kernel with standard deviation σ pixels and radius 3σ; K_1 and K_2 therefore have sizes 7×7 and 13×13. Spatial derivatives use central differences with unit pixel spacing. Both differentiation and Gaussian filtering use replicate padding. For numerical stability, the gradient magnitude is implemented as $\sqrt{(\partial_x X^{(2)})^2 + (\partial_y X^{(2)})^2 + 10^{-12}} - 10^{-6}$, retaining a zero response in uniform regions.

The observation coefficients are fitted by least squares on 500 paired resist-ADI samples from the training split, yielding:

$$(\beta_0, \beta_r, \beta_e) = (0.411003, -0.060164, -0.014975). \tag{D.3}$$

The implementation retains the full calibration precision. Following Equation (3), the region and edge responses are linearly combined, clipped to $[0, 1]$, and filtered by Q to form $Y_{\mathrm{ref}}^{(3)}$. Candidate ADIs are compared in the same low-pass domain. Process conditions affect the reference through $X^{(2)}$; the operator describes a simulation-to-observation correspondence rather than an additional development step.

D.3 Offline Calibration and Numerical Evaluation

The patterning calibration set contains 50 distinct layouts from the training split, each covering all 36 process conditions, for $1{,}800$ mask-resist pairs. A further 20 disjoint layouts provide 720 calibration-validation pairs. Layout grouping and decoded-pixel hashes prevent overlap between the two sets, and a deterministic ordering based on seed 42 fixes their membership. Coefficients are selected exclusively on the calibration set; neither the calibration-validation set nor the final test split participates in fitting.

Calibration minimizes mean pixelwise squared error over the original field of view using the full-image computation described in Appendix D.1. With $\tau = 0.05$ fixed, two rounds of coordinate search are initialized at $(\nu_c, a/\nu_c, 50b\nu_c^2) = (0.015, 0.8, 0.5)$ over the candidates in Table D.2. For each optical candidate, c is selected from $[0.5, 6]$ with spacing 0.25 and subsequently refined within ± 0.25 of the selected value using spacing 0.025.

Table D.2: Candidate values for offline patterning calibration.

Search variable	Candidate values
ν_c	0.005, 0.008, 0.010, 0.0125, 0.015, 0.020, 0.025, 0.04, 0.06, 0.08, 0.12, 0.18, 0.25
a/ν_c	0.4, 0.6, 0.8, 1.0
$50b\nu_c^2$	0, 0.25, 0.5, 1, 2, 4

Table D.3 evaluates both coefficient sets under the same full-image protocol. MSE is computed on normalized intensities. For this operator evaluation, mIoU uses a threshold of 0.5, averages foreground and background IoU within each image, and then averages across images. Calibration reduces validation MSE by 12.46% and increases mIoU by 1.26 percentage points. The preliminary and updated fits differ in both sample size and fitting domain, so the comparison measures the combined effect of recalibration.

Table D.3: Patterning-operator accuracy on 720 calibration-validation pairs.

Coefficient set	MSE↓	mIoU (%)↑
Preliminary calibration	0.033510	90.4564
Calibration on 1,800 pairs	0.029335	91.7129

Across three source types and three cutoff frequencies, numerical consistency checks between the single-precision GPU implementation and double-precision source-wise summation yield a maximum absolute intensity difference of 2.51×10^{-7}.

The full operator G is compared with a region-only fit and a constant-intensity baseline using MSE against $Q(X^{(3)})$. The region-only fit omits H, while the constant baseline uses the mean low-pass ADI intensity of the calibration set. All coefficients are fitted exclusively on calibration data, isolating the contribution of geometric response terms to observation fidelity.

E Derivation and Numerical Details of the Method

This appendix develops the regional correction and gradient approximation underlying Section 3, together with the boundary construction, tolerance calibration, and gradient paths used in paired replay. It complements the physical-response operators in Appendix D.

Image-latent interface. Single-channel state images in $[0, 1]$ are replicated across three channels and affinely mapped to $[-1, 1]$ before VAE encoding. Posterior means define latent representations without stochastic sampling or additional latent rescaling. Raw decoder outputs are averaged across channels, mapped back by the inverse intensity transform, and clipped to $[0, 1]$. These fixed transformations are included in E and D and remain in the input-gradient paths, while the VAE parameters remain frozen. Source images used for conditioning follow the same input intensity normalization.

E.1 Regional Discrepancies and Tolerance Calibration

The four regions are defined by the horizontal and vertical image midlines, with nonoverlapping pixel membership. The geometric mask C_{bin} is formed by the indicator of an intensity greater than or equal to 0.5. Dilation and erosion in Equation (9) are the local maximum and minimum over a 3×3 square neighborhood, respectively, with replicate padding and one application of each operation. Morphology is evaluated before quadrant partitioning, avoiding artificial boundaries at the midlines. The geometric mask and boundary band remain fixed when differentiating the response discrepancy. Only valid pixels contribute to either average; an empty boundary intersection con-

tributes zero, while the region-wide term still supervises valid interior pixels. A wholly invalid quadrant contributes zero discrepancy and zero gradient.

Tolerance calibration evaluates the same regional functional on GT pairs. Specifically, $X_{\text{in}}^{(k)}$ is replaced by $X^{(k-1)}$ and $D(z)$ by $X^{(k)}$ in Equation (8), without latent extrapolation or reconstruction. Let $e_{j,r}^{\text{GT},(k)}$ denote the resulting discrepancy for training sample r in a nonempty region. The tolerance is defined as:

$$\varepsilon_j^{(k)} = \text{Quantile}_{0.9}\left(\left\{e_{j,r}^{\text{GT},(k)}\right\}_r\right). \tag{E.1}$$

The superscript makes explicit the stage dependence suppressed in the local correction equations. The empirical quantile uses linear interpolation between sorted observations at one-based rank $1 + 0.9(N-1)$ for N valid observations. Tolerances are determined exclusively from the training split and then fixed. This calibration accommodates the discrepancy between the approximate prior and paired targets, rather than requiring exact agreement with the prior. It is distinct from fitting the optical and observation coefficients in Appendix D.

E.2 Coupled Least-Squares Correction and Magnitude Control

For fixed current state and conditioning, the local linearization is $e_j(v+\delta) \simeq e_j(v) + g_j^\top \delta$. The unscaled correction solves the following problem:

$$\Delta v = \arg\min_\delta \left\{ \frac{1}{2}\sum_{j=1}^{4}(g_j^\top \delta + b_j)^2 + \frac{\lambda}{2}\|\delta\|_2^2 \right\}. \tag{E.2}$$

Thus, the least-squares fit is performed in velocity space with target changes $-b_j$. Its stationarity condition is:

$$\lambda \Delta v + \sum_{j=1}^{4} g_j(g_j^\top \Delta v + b_j) = 0. \tag{E.3}$$

Substituting $\Delta v = -\sum_j a_j g_j$ and solving $(B + \lambda I_4)\mathbf{a} = \mathbf{b}$ gives $\mathbf{b} - B\mathbf{a} = \lambda \mathbf{a}$, which satisfies Equation (E.3). Since B is a Gram matrix and $\lambda > 0$, the coefficient matrix is positive definite and the velocity-space objective is strictly convex. The four-dimensional solve therefore obtains the unique correction without forming a latent-dimensional Hessian. Coefficients a_j are signed combination weights, not probabilities or independent regional step sizes.

The regularizer $\lambda = 10^{-3}\,\text{tr}(B)/4 + 10^{-8}$ scales with the mean squared gradient norm and includes a positive numerical floor. After the solve, Equation (13) enforces $\|\eta\Delta v\|_2 \le 0.2\|v\|_2$. A zero base velocity therefore receives zero correction, and zero regional gradients also yield zero correction. This scaling bounds the local perturbation while preserving its direction. The predicted regional changes are:

$$g_j^\top(\eta\Delta v) = -\eta(B\mathbf{a})_j = -\eta(b_j - \lambda a_j). \tag{E.4}$$

Consequently, $b_j = 0$ penalizes changes to a region within tolerance without imposing exact invariance; coupling and regularization can still produce a residual change. The linear approximation likewise does not imply monotonic descent of every nonlinear regional discrepancy. Re-evaluation after each state update adapts the next correction to the evolved state.

The four full-latent gradients and their inner products are evaluated separately for each sample. Physical responses and their derivatives use FP32, and the 4×4 positive-definite system is solved by Cholesky factorization in FP64; no explicit matrix inverse is formed. The trace, regularization, solve, and norm scaling are computed per sample, so batch composition does not couple distinct trajectories.

E.3 First-Order Training Gradients

The partial derivative $\nabla_v e_j$ treats the current state, time, and conditioning as fixed arguments and differentiates through z, D, and the relevant response mapping. The numerical correction is recomputed from current network outputs. For parameter optimization, its contribution is detached:

$$\widetilde{v}_\theta^{(k)}(u,t) = v_\theta^{(k)}(u,t) + \text{sg}(\eta\Delta v), \tag{E.5}$$

where sg is the identity in the forward pass and has zero derivative. This has the same forward value as Equation (13), but defines a first-order gradient surrogate rather than the exact derivative through the correction. In particular, gradient computation does not differentiate the regional-gradient construction, coefficient solve, or magnitude scaling. Base-velocity and state-recursion paths remain differentiable, including their dependence on trainable process embeddings and evolved upstream inputs.

For each Euler step, the resulting total derivative satisfies:

$$\frac{\mathrm{d}\widehat{u}^{(k)}_{t_{n+1}}}{\mathrm{d}\theta} = \frac{\mathrm{d}\widehat{u}^{(k)}_{t_n}}{\mathrm{d}\theta} + \frac{1}{K_k}\frac{\mathrm{d}v^{(k)}_\theta(\widehat{u}^{(k)}_{t_n}, t_n)}{\mathrm{d}\theta}. \tag{E.6}$$

The total derivative includes the dependence of the stage input on earlier stages. Frozen E, D, and E_n retain input derivatives where needed, while their parameters remain unchanged. The same numerical correction is used for GT-pair pretraining, cascaded joint training, and inference; the surrogate affects optimization gradients only.

E.4 Recomposition and Replay Gradients

The exchange mask in Equation (14) is evaluated by exact equality of the aligned stored labels within their common valid region. It is distinct from the thresholded geometric mask used for regional scoring. Label-differing pixels retain their own branch values, while label-consistent pixels receive the other branch's evolved values. Both masks and labels are fixed with respect to θ, giving:

$$\frac{\mathrm{d}\widetilde{X}^{(k)}_a}{\mathrm{d}\theta} = (1 - A^{(k)}_{ab}) \odot \frac{\mathrm{d}\widehat{X}^{(k)}_a}{\mathrm{d}\theta} + A^{(k)}_{ab} \odot \frac{\mathrm{d}\widehat{X}^{(k)}_b}{\mathrm{d}\theta}. \tag{E.7}$$

The other branch is symmetric. Re-encoding $\widetilde{X}^{(k)}_q$ supplies both the next stage's initial latent state and image condition, while p_q remains unchanged. Replay traverses all three Euler steps of that next stage, using a reference derived from the recomposed input and the first-order convention of Appendix E.3. Its loss therefore updates the downstream mapping and, through Equation (E.7), both contributing upstream branches.

The two transition-level replays use recompositions of the unrecomposed trajectories: $k = 1$ supervises the replayed Resist Image and $k = 2$ supervises the replayed ADI. They are not recursively recomposed with one another. In Equation (15), the factor $1/4$ averages the four branch-transition pairs, each contributing an upstream and a downstream term. Original downstream labels define robustness supervision under these perturbations, not newly measured outcomes of the recomposed inputs. Retaining ordinary supervision constrains upstream drift, while model-dependent intermediate states expose downstream transitions to current rollout deviations. Paired labels are required only for this training objective; ordinary inference follows Equation (6) from the input layout and process condition.