MECOGEN: MECHANISTICALLY CONDITIONED GENERATIVE FORECASTING OF SINGLE-CELL DYNAMICS

Anonymous authors
Paper under double-blind review

ABSTRACT

Forecasting single-cell populations beyond the observed time horizon is challenging because cellular states and regulatory programs can transition into unseen regimes, particularly during iPSC differentiation, while observations are limited to heterogeneous, unlabeled population snapshots. We identify a fundamental conditioning bottleneck: although modern generative models can represent complex transcriptional distributions, they require reliable future biological conditions for accurate long-horizon forecasting. To address this challenge, we introduce **MeCoGen** (**Me**chanistically **Co**nditioned **Gen**eration), a mechanistic-data-driven hybrid framework for single-cell forecasting. MeCoGen first uses a stochastic mechanistic model, informed by a subset of core regulatory genes and interactions, to predict a biologically structured future state capturing regulatory dynamics and developmental progression. This predicted state then conditions a generative model that synthesizes future high-dimensional transcriptomic profiles. To capture the sparse and stochastic nature of single-cell expression data, the generator separately learns gene occurrence and expression magnitude using discrete diffusion and conditional flow matching, respectively. We evaluate MeCoGen on three temporal single-cell systems spanning differentiation and cellular reprogramming. Across all datasets, MeCoGen consistently outperforms recent temporal forecasting baselines, while controlled conditioning experiments demonstrate that mechanistically predicted future states substantially improve long-horizon predictions over purely data-driven extrapolation. Ablation studies further validate the benefits of separately modeling gene occurrence and expression magnitude.

1 INTRODUCTION

Induced pluripotent stem cell (iPSC) differentiation and cellular reprogramming are highly dynamic processes shaped by both cell-intrinsic regulatory programs and cell-to-cell interactions, which jointly drive lineage specification and population heterogeneity. Time-resolved single-cell RNA sequencing (scRNA-seq) captures these dynamics as destructive, unpaired snapshots of evolving populations rather than longitudinal measurements of individual cells (Weinreb et al., 2018; Schiebinger et al., 2019; Tong et al., 2020; Herbach et al., 2017). Consequently, inferring underlying spatiotemporal dynamics is fundamentally underdetermined (Weinreb et al., 2018), with the challenge becoming even more pronounced for *out-of-time forecasting*, where future cell-state distributions must be predicted beyond the observed time horizon. Although optimal transport, neural ODE/SDE, diffusion, and flow-based approaches have significantly advanced temporal modeling of single-cell dynamics (Schiebinger et al., 2019; Tong et al., 2020; Huguet et al., 2022; Yeo et al., 2021; Zhang et al., 2024; Jiang & Wan, 2024; Vinyard et al., 2025; Tong et al., 2024b; Luo et al., 2024; Palma et al., 2025), reliable extrapolation remains challenging due to compounding forecast errors and interaction-driven distribution shifts (Zhang et al., 2024; Jiang & Wan, 2024).

A key challenge is that cellular dynamics evolve as cells progress through *developmental states*, corresponding to distinct regulatory programs reflected in gene-expression patterns. This motivates representing cellular state with a compact set of mechanistically linked regulatory genes and extrapolating this core before generating the full transcriptome. In generative forecasting, this state is unobserved at future time points and must be extrapolated before full transcriptomes can be generated. Purely data-driven approaches are vulnerable to distribution shift because cells may enter regulatory regimes that are sparsely represented or absent from the training window. Errors in state

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.

extrapolation therefore propagate to downstream predictions, creating a *conditioning bottleneck* that limits long-horizon forecasting. Reliable out-of-time prediction thus requires not only an expressive generator but also a biologically grounded state representation whose dynamics remain predictive beyond the observed horizon. Existing latent ODE/SDE and probabilistic temporal models (Yeo et al., 2021; Zhang et al., 2024; Jiang & Wan, 2024) typically learn both state dynamics and generative mappings directly from observational data, limiting robustness under distribution shift.

Existing mechanistic models provide complementary structure for extrapolating future cellular states. Stochastic gene regulatory network (GRN) models capture regulatory interactions, transcriptional noise, and state-dependent dynamics (Herbach et al., 2017; Ventre, 2021; Ventre et al., 2023; Dibaeinia & Sinha, 2020; Cannoodt et al., 2021), but typically describe only a subset of the transcriptome. This motivates a hybrid strategy: *extrapolate structured biological dynamics mechanistically and generate unresolved high-dimensional variation conditionally*.

Based on this principle, we introduce **MeCoGen** (**Me**chanistically **Co**nditioned **Gen**eration), a hybrid mechanistic–data-driven framework for out-of-time single-cell forecasting (Figure 1). MeCoGen represents cellular state using a compact mechanistic core comprising regulatory genes linked by prior GRN knowledge and a discrete developmental stage. A stochastic mechanistic model extrapolates this core beyond the observation window by modeling regulatory interactions, state-dependent kinetics, transcriptional stochasticity, developmental-state transitions, and population dynamics. The resulting future core-state distribution conditions a high-dimensional generator that reconstructs the extended transcriptomic profile. Thus, MeCoGen uses mechanistic structure for extrapolation while learning unresolved transcriptomic variation from data.

We further exploit the established hurdle-like structure of single-cell expression, distinguishing gene occurrence from positive-expression magnitude (Finak et al., 2015). For each gene, occurrence (whether it is expressed) and magnitude (how strongly it is expressed when active) capture distinct sources of variation. Instead of modeling both with a single continuous objective, we assign them distinct but coupled generative processes while preserving multigene dependencies within each component. Conditional discrete diffusion captures joint binary occurrence patterns (Austin et al., 2021), whereas conditional flow matching models joint positive-expression magnitudes (Lipman et al., 2023). This hybrid formulation aligns each component with its natural statistical domain, improving preservation of both sparsity structure and continuous expression variation.

We evaluate MeCoGen on branching human iPSC-to-cardiomyocyte differentiation, mouse fibroblast reprogramming, and human pancreatic endocrine differentiation. Across all three systems, MeCoGen consistently improves out-of-time forecasting over recent temporal single-cell baselines across multiple distributional metrics. Conditioning ablations support the proposed conditioning bottleneck: at the longest iPSC-CM forecast horizon, mechanistic conditioning reduces remaining modeled transcriptome W_2 error by 50% and state-distribution error by 83% relative to purely data-driven conditioning. Generator ablations further demonstrate the benefit of separately modeling gene occurrence and positive-expression magnitude.

Our contributions are threefold: 1) We formulate out-of-time single-cell prediction as a *conditioning-limited generative extrapolation* problem and propose mechanistic future conditioning to improve extrapolation beyond observed support. 2) We develop a stochastic structured-state model that extrapolates regulatory and developmental dynamics and provides explicit future-state conditioning for transcriptome generation. 3) We develop a structure-aware conditional generator that instantiates the occurrence–magnitude decomposition with joint discrete diffusion and conditional flow matching, yielding consistent improvements across three temporal single-cell systems.

2 Related Work

Temporal single-cell dynamics and forecasting. Because scRNA-seq provides destructive population snapshots, many methods infer developmental dynamics from cross-sectional data (Weinreb et al., 2018). Trajectory inference and RNA-velocity approaches recover cell-state ordering, branching structure, fate probabilities, or local transition directions within the observed manifold (Trapnell et al., 2014; Qiu et al., 2017; Haghverdi et al., 2016; Street et al., 2018; Setty et al., 2019; Wolf et al., 2019; Saelens et al., 2019; La Manno et al., 2018; Bergen et al., 2020; Qiu et al., 2022; Lange et al., 2022; Li, 2023; Gorin et al., 2022). Population-level approaches based on optimal transport

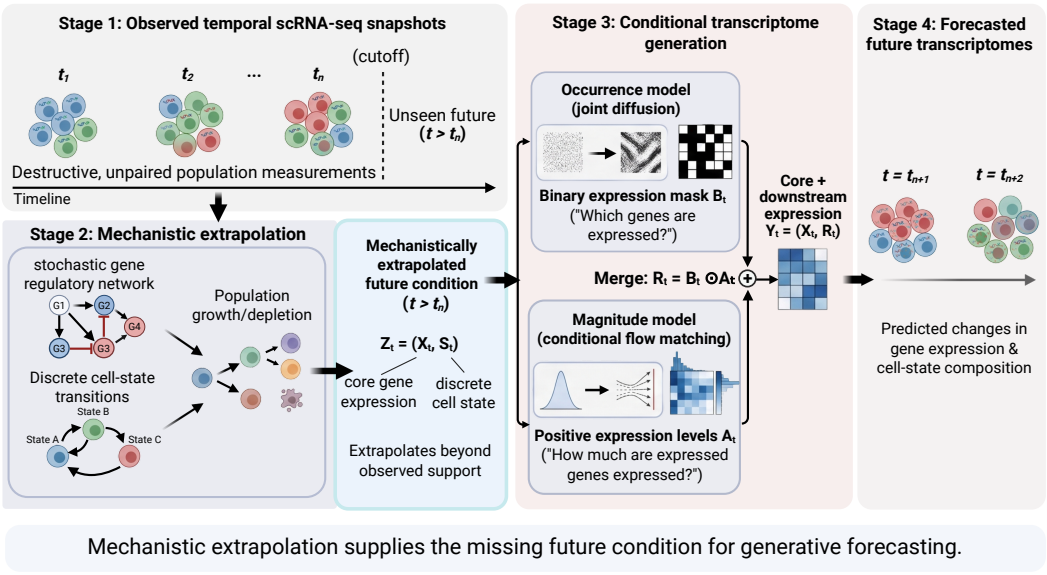

Figure 1: MeCoGen for out-of-time single-cell distribution forecasting. MeCoGen extrapolates a biologically structured future condition $\boldsymbol{z}_t = (\boldsymbol{x}_t, s_t)$ using mechanistic dynamics and conditionally generates the future transcriptome through joint modeling of gene occurrence and positive-expression magnitude.

and continuous flows primarily reconstruct trajectories or interpolate between observed populations (Schiebinger et al., 2019; Tong et al., 2020; Huguet et al., 2022; Klein et al., 2025; Sha et al., 2024; Tong et al., 2024b). A complementary line of work directly models temporal prediction and extrapolation using neural ODE/SDEs, latent stochastic processes, and related generative formulations (Yeo et al., 2021; Zhang et al., 2024; Jiang & Wan, 2024; Vinyard et al., 2025; Jiang et al., 2026; Balik & Lähdesmäki, 2026; BinTayyash et al., 2021; Woicik et al., 2023). Despite this progress, robust prediction beyond the final observed time point remains challenging as forecast horizons extend and cellular populations enter poorly observed regimes (Zhang et al., 2024; Jiang & Wan, 2024).

Mechanistic and biology-informed modeling. Mechanistic models encode gene regulation through stochastic kinetics, promoter switching, and transcriptional bursting (Gillespie, 1977; Peccoud & Ycart, 1995; Raj et al., 2006; Shahrezaei & Swain, 2008; Suter et al., 2011; Munsky et al., 2012). Models such as HARISSA and CARDAMOM infer GRN dynamics from temporal single-cell data and enable forward simulation (Herbach et al., 2017; Ventre, 2021; Ventre et al., 2023), while related frameworks incorporate regulatory, pathway, or graph-based biological priors (Dibaeinia & Sinha, 2020; Cannoodt et al., 2021; Aibar et al., 2017; Fortelny & Bock, 2020; Roohani et al., 2024; Fournié et al., 2026). These approaches provide interpretable and biologically grounded dynamics, but typically operate on a restricted set of genes and do not model the full transcriptome.

Single-cell generation and conditioning. Single-cell generative models capture count structure and cellular heterogeneity using probabilistic latent-variable models, autoencoders, adversarial methods, statistical simulators, and diffusion or flow-based models (Risso et al., 2018; Eraslan et al., 2019; Lopez et al., 2018; Gayoso et al., 2021; Ashuach et al., 2023; Marouf et al., 2020; Sun et al., 2021; Song et al., 2024; Luo et al., 2024; Palma et al., 2025). Conditional models predict cellular responses to perturbations, treatments, and interventions (Lotfollahi et al., 2019; 2023; Hetzel et al., 2022; Bunne et al., 2023; Roohani et al., 2024), while foundation models learn transferable cellular representations (Lotfollahi et al., 2022; Theodoris et al., 2023; Cui et al., 2024; Hao et al., 2024).

3 PRELIMINARIES

Conditional discrete diffusion. Discrete diffusion models define a Markov corruption process over discrete variables and learn a reverse denoising process (Austin et al., 2021). Let X_0 de-

note a discrete sample and C a conditioning variable. The forward process is specified by transition matrices $\{Q_k\}_{k=1}^K$, where $q_k(X_k = j \mid X_{k-1} = i) = [Q_k]_{ij}$. It factorizes as $q(X_{1:K} \mid X_0) = \prod_{k=1}^K q_k(X_k \mid X_{k-1})$, progressively transforming the data distribution of X_0 toward a simple reference distribution. Conditional generation uses the learned reverse process $p_\theta(X_{0:K-1} \mid X_K, C) = \prod_{k=1}^K p_\theta(X_{k-1} \mid X_k, C)$, which generates samples by iteratively denoising from the reference distribution.

Conditional flow matching. Conditional flow matching (CFM) learns a vector field that transports a source distribution p_0 to a target distribution $p_1(\cdot \mid C)$ conditioned on C (Lipman et al., 2023; Tong et al., 2024a). Given samples $U_0 \sim p_0$ and $U_1 \sim p_1(\cdot \mid C)$, CFM constructs the optimal-transport (OT) inspired interpolation $U_\tau = (1-\tau)U_0 + \tau U_1$, $\tau \in [0,1]$, with velocity $U_1 - U_0$. A conditional vector field $v_\psi(U_\tau, \tau; C)$ is trained via

$$\mathcal{L}_{\mathrm{CFM}} = \mathbb{E}\left[\|v_\psi(U_\tau, \tau; C) - (U_1 - U_0)\|_2^2\right] \tag{1}$$

encouraging the learned dynamics to follow the transport path. Generation samples $U_0 \sim p_0$ and integrates $dU_\tau/d\tau = v_\psi(U_\tau, \tau; C)$ from $\tau = 0$ to 1.

4 Hybrid Mechanistic–Generative Forecasting

MeCoGen follows a simple principle: *extrapolate structured biological dynamics of core genes mechanistically and generate unresolved high-dimensional variation conditionally*. The framework comprises two coupled components. First, a compact mechanistic core consists of regulatory genes linked by prior gene-regulatory knowledge and a discrete developmental state summarizing cellular progression. A stochastic multiscale model extrapolates this core beyond the observation window by modeling regulatory interactions, state-dependent kinetics, transcriptional stochasticity, developmental-state transitions, and population dynamics.

Second, the predicted core-state distribution conditions a high-dimensional transcriptome generator. Rather than imposing mechanistic dynamics on the full transcriptome, MeCoGen leverages biological priors where available and learns the remaining variation from data. To accommodate the hurdle-like structure of single-cell expression, the generator separately models gene occurrence and positive-expression magnitude.

Formally, let $\mathcal{D}_{\le t_n} = \{(t_k, \mathcal{Y}_{t_k})\}_{t_k \le t_n}$ denote observed temporal snapshots, where $\mathcal{Y}_{t_k} = \{\boldsymbol{y}_i(t_k)\}_{i=1}^{N_k}$ is the gene-expression population of size N_k observed at time t_k. Our goal is to forecast the distribution of a future cell state $\boldsymbol{y}_t$ at $t > t_n$, optionally conditioned on known experimental inputs $\boldsymbol{u}$. We partition gene expression as $\boldsymbol{y}_t = (\boldsymbol{x}_t, \boldsymbol{r}_t)$, where $\boldsymbol{x}_t \in \mathbb{R}_+^{G_x}$ contains mechanistically modeled core genes and $\boldsymbol{r}_t \in \mathbb{R}_+^{G_r}$ contains the remaining genes, with $G = G_{\boldsymbol{x}} + G_{\boldsymbol{r}}$. Let $s_t \in \{1, \dots, K_s\}$ denote the developmental state of the cell. Together, $\boldsymbol{x}_t$ and s_t define the structured mechanistic state $\boldsymbol{z}_t = (\boldsymbol{x}_t, s_t)$. MeCoGen models the future population distribution as

$$p(\boldsymbol{y}_t \mid \mathcal{D}_{\le t_n}, \boldsymbol{u}) = \sum_{s=1}^{K_s} p_{\mathcal{M}}(\boldsymbol{x}_t, s_t = s \mid \mathcal{D}_{\le t_n}, \boldsymbol{u})\, p_\phi(\boldsymbol{r}_t \mid \boldsymbol{x}_t, s_t = s), \qquad t > t_n. \tag{2}$$

where the mechanistic component $p_{\mathcal{M}}$ extrapolates the future distribution of core-gene expression and developmental state, and the conditional generator p_ϕ generates the remaining transcriptome conditioned on this structured state. This factorization assumes that temporal and experimental effects on $\boldsymbol{r}_t$ are mediated through the structured state $\boldsymbol{z}_t = (\boldsymbol{x}_t, s_t)$. Sections 4.1, 4.2, and 4.3 describe the mechanistic extrapolation model, conditional transcriptome generator, and training procedure, respectively.

4.1 Mechanistic Structured-State Extrapolation

Core gene regulatory dynamics. For each biological system, we define a compact mechanistic core consisting of regulatory genes and signed interactions supported by prior biological evidence and fixed independently of the forecasting targets. The resulting gene-regulatory network (GRN) encodes activating and repressive effects through Hill-type regulatory functions (Santillán, 2008). For

each simulated cell i, core-gene expression evolves according to stochastic production and degradation dynamics. For gene g,

$$dP_{ig} = p_g(\boldsymbol{x}_i(t), s_i(t), \boldsymbol{u}_t)\, dt + \sqrt{\frac{2p_g(\boldsymbol{x}_i(t), s_i(t), \boldsymbol{u}_t)}{\beta_g}}\, dW_{ig}^{\mathrm{prod}}(t), \tag{3}$$

and

$$dD_{ig} = \gamma_g(s_i(t))x_{ig}(t)\, dt + \sqrt{\gamma_g(s_i(t))x_{ig}(t)}\, dW_{ig}^{\mathrm{deg}}(t), \tag{4}$$

with independent Wiener processes, where $p_g(\boldsymbol{x}_i(t), s_i(t), \boldsymbol{u}_t)$ is the GRN-driven production rate of gene g, β_g controls the production-associated bursting stochasticity in the Chemical-Langevin regime, and $\gamma_g(s_i(t))$ is the developmental-state-dependent degradation rate. The production rate depends on regulatory inputs, developmental state, and optional experimental conditions. The net core-gene dynamics are therefore

$$dx_{ig}(t) = dP_{ig} - dD_{ig}. \tag{5}$$

The noise term captures transcriptional stochasticity, with a compound-Poisson formulation used for low-copy genes (Appendix A).

Developmental-state transitions. For each simulated cell i, the developmental state $s_i(t)$ evolves as a topology-constrained continuous-time Markov process on a curated, biologically supported developmental graph. For an admissible transition (s, s'), the transition rate is

$$q_{ss',i}(t) = q_{ss'}^{\max}\Phi_{ss'}(\boldsymbol{x}_i(t); k_{ss'})\eta_{ss',i} + \beta_{ss'}t + \kappa_{ss'}u(t), \tag{6}$$

where $\Phi_{ss'}(\boldsymbol{x}_i(t); k_{ss'}) \in [0, 1]$ is a Hill-type regulatory propensity determined by the transition-specific trigger-gene activity, $\beta_{ss'}t$ captures edge-specific temporal progression, and $\kappa_{ss'}$ captures sensitivity to experimental inputs (see Appendix A). Developmental state modulates regulatory kinetics, whereas core-gene expression governs state transitions.

Population dynamics and structured-state prediction. To model differential expansion and depletion, each simulated trajectory is assigned a time-varying population representation $w_i(t)$,

$$\frac{dw_i(t)}{dt} = r_{s_i(t)}(t)\, w_i(t), \qquad w_i(0) = 1, \tag{7}$$

where $r_s(t)$ is the net growth rate of developmental state s. The normalized representations define the predicted future distribution $p_{\mathcal{M}}(\boldsymbol{x}_t, s_t \mid \mathcal{D}_{\leq t_n}, \boldsymbol{u})$, which provides the future structured condition for transcriptome generation. Dataset-specific choices of $r_s(t)$ and simulation details are provided in Appendices A. Before being used as the generator condition, the simulated population undergoes state-conditioned expression calibration followed by a marginal-preserving discrete diffusion refinement to reduce mismatch between simulated and empirical conditioning distributions. Details are provided in Appendix E.

4.2 Joint Stochastic Transcriptome Generation

The generated expression block $\boldsymbol{r}_t \in \mathbb{R}_+^{G_r}$ is disjoint from the mechanistic core. Because these genes exhibit both substantial sparsity and correlated positive expression, MeCoGen adopts a hurdle decomposition,

$$\boldsymbol{r}_t = \boldsymbol{b}_t \odot \boldsymbol{a}_t, \qquad b_{t,g} = \mathbb{I}[r_{t,g} > 0], \tag{8}$$

where $g \in \{1, \ldots, G_r\}$ indexes the conditionally generated genes, $\boldsymbol{b}_t \in \{0, 1\}^{G_r}$ denotes the multi-gene occurrence pattern, and $\boldsymbol{a}_t \in \mathbb{R}_+^{G_r}$ the positive-expression magnitudes.

Conditioned on the structured state $\boldsymbol{z}_t = (\boldsymbol{x}_t, s_t)$, we factorize the joint distribution as

$$p_\phi(\boldsymbol{b}_t, \boldsymbol{a}_t \mid \boldsymbol{z}_t) = p_\theta(\boldsymbol{b}_t \mid \boldsymbol{z}_t)\, p_\psi(\boldsymbol{a}_t \mid \boldsymbol{b}_t, \boldsymbol{z}_t). \tag{9}$$

This factorization defines a hierarchical conditional generator for $\boldsymbol{r}_t$: first sampling the occurrence pattern $\boldsymbol{b}_t \sim p_\theta(\boldsymbol{b}_t \mid \boldsymbol{z}_t)$, and then sampling the corresponding positive-expression magnitudes $\boldsymbol{a}_t \sim p_\psi(\boldsymbol{a}_t \mid \boldsymbol{b}_t, \boldsymbol{z}_t)$. The resulting pair $(\boldsymbol{b}_t, \boldsymbol{a}_t)$ is mapped deterministically to the expression vector through equation 8, thereby inducing the conditional distribution $p_\phi(\boldsymbol{r}_t \mid \boldsymbol{z}_t)$. Notably, the occurrence mask $\boldsymbol{b}_t$ represents observed gene occurrence and is distinct from the transcriptional bursting dynamics in the mechanistic core.

Joint occurrence diffusion. To capture coordinated sparsity patterns, we model the occurrence mask jointly as $p_\theta(\boldsymbol{b}_t \mid \boldsymbol{z}_t)$ using conditional discrete diffusion. The forward process progressively corrupts $\boldsymbol{b}_t$ by replacing entries with unbiased Bernoulli samples,

$$b_{t,g}^{(k)} = \begin{cases} b_{t,g}^{(0)}, & \text{with probability } \bar{\alpha}_k, \\ \text{Bernoulli}(1/2), & \text{with probability } 1-\bar{\alpha}_k. \end{cases} \tag{10}$$

where $\bar{\alpha}_k$ is the cumulative retention probability at diffusion step k. A conditional denoiser predicts the clean occurrence probabilities $\hat{\boldsymbol{p}}_{0,\theta}^{(k)} = \boldsymbol{p}_\theta(\boldsymbol{b}_t^{(k)}, k, \boldsymbol{z}_t)$, and is trained with binary cross-entropy,

$$\mathcal{L}_{\text{occ}} = \mathbb{E}_{\boldsymbol{b}_t^{(0)}, \boldsymbol{z}_t, k}\left[\text{BCE}\left(\hat{\boldsymbol{p}}_{0,\theta}^{(k)}, \boldsymbol{b}_t^{(0)}\right)\right]. \tag{11}$$

Conditioning on the full corrupted mask enables the model to capture multigene occurrence dependencies. Additional details are provided in Appendix C.

Conditional flow matching for positive magnitudes. Given an occurrence mask $\boldsymbol{b}_t$, MeCoGen models positive-expression magnitudes $\boldsymbol{a}_t$ jointly using conditional flow matching. Positive entries are log-transformed and standardized, $\widetilde{\boldsymbol{a}}_t = \mathcal{T}_+(\boldsymbol{a}_t)$. Given a Gaussian source sample $\boldsymbol{\epsilon} \sim \mathcal{N}(0, I)$, we define the OT-based linear interpolation

$$\boldsymbol{h}_\tau = (1-\tau)\boldsymbol{\epsilon} + \tau\widetilde{\boldsymbol{a}}_t, \qquad \tau \in [0,1]. \tag{12}$$

We train a conditional vector field by matching the target velocity $\widetilde{\boldsymbol{a}}_t - \boldsymbol{\epsilon}$, i.e.,

$$\mathcal{L}_{\text{FM}} = \mathbb{E}\left[\frac{\sum_g b_{t,g}\left[v_{\psi,g}(h_\tau, \tau; b_t, z_t) - (\tilde{a}_{t,g} - \epsilon_g)\right]^2}{\max\left(1, \sum_g b_{t,g}\right)}\right]. \tag{13}$$

Only expressed genes contribute to the loss. At generation time, we sample $\boldsymbol{h}_0 \sim \mathcal{N}(0, I)$ and integrate

$$\frac{d\boldsymbol{h}_\tau}{d\tau} = \boldsymbol{v}_\psi(\boldsymbol{h}_\tau, \tau; \boldsymbol{b}_t, \boldsymbol{z}_t), \qquad \tau \in [0,1], \tag{14}$$

before mapping the endpoint back to the original expression scale, $\boldsymbol{a}_t = \mathcal{T}_+^{-1}(\boldsymbol{h}_1)$. The generated transcriptome therefore follows the hierarchy

$$\boldsymbol{z}_t \longrightarrow \boldsymbol{b}_t \sim p_\theta(\boldsymbol{b}_t \mid \boldsymbol{z}_t) \longrightarrow \boldsymbol{a}_t \sim p_\psi(\boldsymbol{a}_t \mid \boldsymbol{b}_t, \boldsymbol{z}_t) \longrightarrow \boldsymbol{r}_t = \boldsymbol{b}_t \odot \boldsymbol{a}_t. \tag{15}$$

4.3 Sequential Training and Optional Transport Regularization

The occurrence and magnitude models are trained sequentially. We first optimize the conditional occurrence diffusion, $\theta^\star = \arg\min_\theta \mathcal{L}_{\text{occ}}(\theta)$, and freeze the resulting model before training the magnitude flow model. The latter is optimized as

$$\psi^\star = \arg\min_\psi \left[\mathcal{L}_{\text{FM}}(\psi) + \lambda_{\text{OT}}\mathcal{L}_{\text{OT}}(\psi; \theta^\star)\right], \tag{16}$$

where the default MeCoGen model uses $\lambda_{\text{OT}} = 0$. We additionally consider MeCoGen-OT, for which a nonzero λ_{OT} is selected using training-only temporal validation.

For the optional transport term, observed training-cell states $z = (x, s)$ provide the condition; occurrence masks are sampled from the frozen diffusion model $p_{\theta^\star}$, and positive-expression magnitudes are generated by integrating the conditional flow. Combining the generated expression block with its corresponding core expression gives samples in the full modeled expression space $[\boldsymbol{x}, \boldsymbol{r}]$. Let $\widehat{\mu}_{[\boldsymbol{x},\boldsymbol{r}]}$ and $\mu_{[\boldsymbol{x},\boldsymbol{r}]}$ denote the generated and observed empirical distributions in this space, respectively. We define

$$\mathcal{L}_{\text{OT}} = \text{Sinkhorn}_{p=2,\varepsilon}\left(\widehat{\mu}_{[\boldsymbol{x},\boldsymbol{r}]}, \mu_{[\boldsymbol{x},\boldsymbol{r}]}\right), \tag{17}$$

where $\text{Sinkhorn}_{p=2,\varepsilon}$ denotes the entropy-regularized W_2 optimal-transport distance. Because the occurrence model is frozen, this auxiliary objective regularizes the magnitude flow while preserving the learned occurrence model. Implementation details of the Sinkhorn objective and differentiable flow integration are provided in Appendix D.

Table 1: Out-of-time single-cell population forecasting. Results are averaged over D11/D15 for iPSC-CM, D6/D7 for SC-islet, and four rolling temporal splits for WOT-MEF. All metrics are evaluated on the complete modeled core-plus-downstream gene panel; Zero denotes mean per-gene zero-fraction error. For SC-islet, Energy values are reported as $\times 10^3$.

Dataset	Method	MMD$\times 10^3$ ↓	Energy↓	KS↓	Zero↓	W_2^{full}	ΔW_2^{full} ↓
iPSC-CM	*Real–Split*	0.13±0.03	0.004±0.001	0.014±0.001	0.008±0.001	47.15±0.12	–
	MeCoGen	7.38±0.07	0.147±0.001	0.177±0.000	0.054±0.001	57.40±0.05	10.25±0.05
	MeCoGen-OT	**6.98±0.07**	**0.138±0.001**	**0.165±0.001**	**0.048±0.000**	55.55±0.06	**8.41±0.06**
	scIMF	25.03±0.14	0.829±0.007	0.466±0.001	0.407±0.000	33.43±0.04	13.71±0.04
	scNODE	55.02±0.10	1.484±0.003	0.500±0.000	0.407±0.000	46.04±0.01	8.58±0.01
	LGP-OT	45.68±0.37	1.401±0.012	0.434±0.001	0.294±0.000	37.72±0.05	9.43±0.05
SC-islet	*Real–Split*	0.001±0.000	0.037±0.007	0.017±0.001	0.009±0.001	52.26±0.02	–
	MeCoGen	0.033±0.000	0.894±0.011	0.159±0.001	**0.049±0.001**	61.09±0.06	8.83±0.06
	MeCoGen-OT	**0.032±0.000**	**0.881±0.010**	**0.159±0.001**	**0.049±0.001**	60.24±0.06	**7.99±0.06**
	scIMF	0.038±0.001	1.658±0.016	0.322±0.001	0.297±0.000	34.04±0.14	18.22±0.14
	scNODE	0.045±0.000	1.944±0.004	0.318±0.000	0.293±0.000	38.84±0.02	13.41±0.02
	LGP-OT	0.111±0.003	4.179±0.088	0.335±0.005	0.221±0.001	42.26±0.24	9.99±0.24
WOT-MEF	*Real–Split*	0.0020±0.0002	0.052±0.003	0.027±0.001	0.015±0.001	50.85±0.04	–
	MeCoGen	**0.074±0.000**	**1.330±0.006**	**0.139±0.000**	**0.100±0.000**	63.63±0.04	12.75±0.04
	MeCoGen-OT	0.087±0.000	1.390±0.006	0.175±0.000	0.100±0.000	58.40±0.03	7.52±0.03
	scIMF	0.207±0.001	5.044±0.040	0.620±0.001	0.550±0.001	44.75±0.04	9.49±0.04
	scNODE	0.256±0.001	6.051±0.017	0.530±0.000	0.452±0.000	51.02±0.04	**3.48±0.04**
	LGP-OT	0.246±0.002	4.887±0.037	0.405±0.001	0.317±0.001	50.72±0.15	7.99±0.15

5 Experiments and Results

Experimental setup. We evaluate **MeCoGen** on three temporal single-cell systems. For iPSC-to-cardiomyocyte differentiation (iPSC-CM) (Elorbany et al., 2022), models are trained on days 0, 1, 3, 5, and 7 and forecast days 11 and 15 using a 33-gene mechanistic core and a six-state branching developmental topology. For mouse fibroblast reprogramming (WOT-MEF) (Schiebinger et al., 2019), we use 41 core genes and evaluate four rolling out-of-time forecasts. For pancreatic endocrine differentiation (SC-islet) (Veres et al., 2019), models are trained on days 0–5 and forecast days 6 and 7 using 45 core genes. In all datasets, MeCoGen predicts 200 additional transcriptomic genes. We compare against scNODE, scIMF, and LGP-OT using their released reference configurations, with no held-out forecast population used for model fitting or hyperparameter selection. Full split and implementation details are provided in Appendix G.

Following temporal single-cell forecasting benchmarks (Osakwe et al., 2026), we evaluate global distributional agreement using MMD, Energy distance, and Sinkhorn-W_2, gene-wise marginals using mean KS, and sparsity using zero-fraction error. To account for variance contraction in raw transport cost, we additionally report $\Delta W_2^{\text{full}} = K^{-1} \sum_{k=1}^{K} \left| W_{2,k}^{\text{full,model}} - W_{2,k}^{\text{full,Real-Split}} \right|$, the mean deviation from the empirical Real–Split transport reference. See details in Appendix G.

MeCoGen improves out-of-time population forecasting. Table 1 compares MeCoGen and its transport-regularized variant, MeCoGen-OT, against three external baselines. Across all datasets, both variants consistently improve distributional, marginal, and sparsity fidelity. On iPSC-CM, MeCoGen-OT achieves MMD 6.98×10^{-3} and Energy 0.138, compared with 25.03×10^{-3} and 0.829 for the strongest baseline. On SC-islet, MeCoGen-OT attains the best MMD (3.22×10^{-5}) and Energy distance (8.81×10^{-4}), while on WOT-MEF, unregularized MeCoGen achieves the lowest MMD (7.41×10^{-5}) and Energy distance (1.33). The gains are accompanied by substantially improved marginal and sparsity fidelity.

Figure 2 illustrates these gains on iPSC-CM out-of-time forecasts at Day 11 (+4 d) and Day 15 (+8 d). MeCoGen more accurately recovers the developmental-state structure and global geometry of future populations than the external baselines, with the advantage becoming more pronounced at longer horizons. It also better recovers sparse multigene occurrence structure, exhibiting substantially stronger agreement in pairwise gene co-occurrence patterns than the competing methods.

Raw W_2 produces a different ranking from MMD, Energy, and sparsity-based metrics. In high-dimensional sparse transcriptomic data, Sinkhorn transport can assign low cost to overly concentrated or smoothed distributions that fail to preserve zero mass, multimodality, or cross-gene depen-

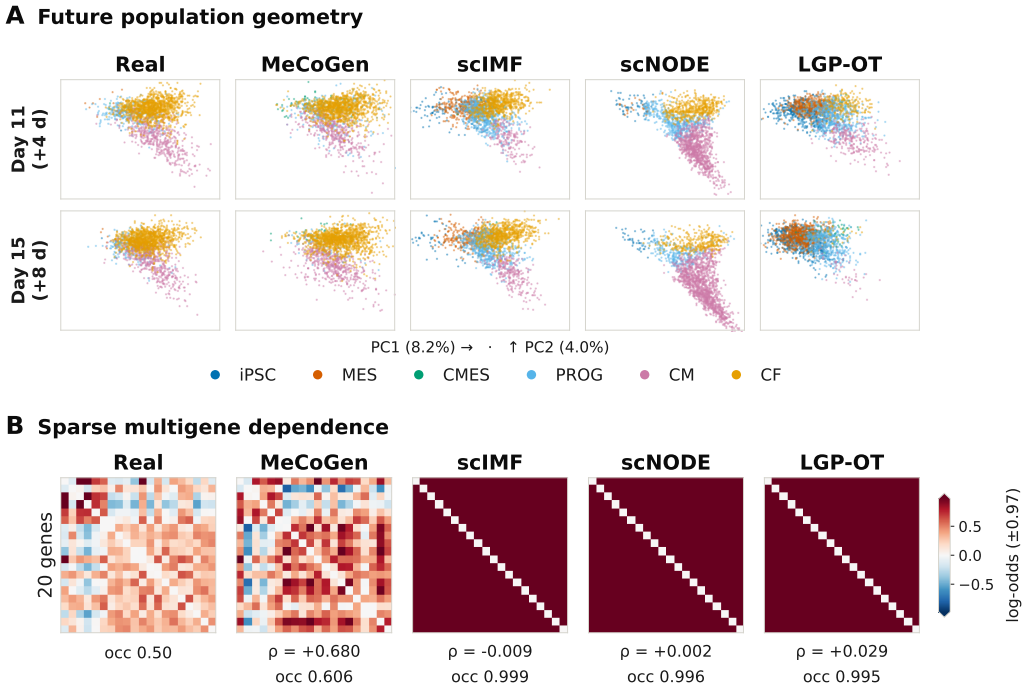

Figure 2: Structural fidelity of out-of-time iPSC-CM forecasts. (A) Day 11 (+4 d) and Day 15 (+8 d) population geometry in a shared training-fitted 2-D PCA basis (PC1+PC2 = 12.2%), colored by developmental state; baseline states are inferred by a training-fitted classifier. (B) Day 15 sparse multigene dependence for a fixed 20-gene panel (Appendix I.4). Heatmaps show pairwise binary co-occurrence log-odds; ρ is their model–real Spearman correlation and *occ* is mean gene occurrence.

Table 2: Controlled conditioning experiment on iPSC-CM using a fixed downstream generator. State TV denotes total-variation distance between predicted and target state distributions. MMD_c, Energy_c, and W_2^c evaluate the core-gene condition; $\mathrm{W}_2^{\mathrm{full}}$ evaluates the resulting full gene panel.

Horizon	Condition	State TV↓	$\mathrm{MMD_c}\times10^3\downarrow$	$\mathrm{Energy_c}\downarrow$	$\mathrm{W_2^c}\downarrow$	$\mathrm{W_2^{full}}$
+4d	Oracle	0.015 ± 0.011	0.0053 ± 0.0014	0.000035 ± 0.000007	0.60 ± 0.01	39.89 ± 0.04
	Mechanism	**0.052 ± 0.009**	**0.93 ± 0.04**	**0.0037 ± 0.0002**	**3.15 ± 0.36**	**46.07 ± 0.29**
	Data-driven	0.481 ± 0.004	3.21 ± 0.01	0.0259 ± 0.0001	6.96 ± 0.02	57.44 ± 0.09
+8d	Oracle	0.015 ± 0.006	0.0036 ± 0.0007	0.000023 ± 0.000003	0.46 ± 0.01	41.49 ± 0.09
	Mechanism	**0.087 ± 0.004**	**1.34 ± 0.05**	**0.0051 ± 0.0002**	**3.66 ± 0.22**	**47.86 ± 0.23**
	Data-driven	0.507 ± 0.004	5.75 ± 0.02	0.0647 ± 0.0002	19.44 ± 0.04	95.31 ± 0.35

dence. Consistent with this effect, the nonzero Real–Split W_2 serves as a useful calibration reference rather than a lower bound. On iPSC-CM and SC-islet, MeCoGen-OT lies closer to this scale than the external baselines while retaining substantially better MMD, Energy, and sparsity fidelity. On WOT-MEF, transport regularization reduces $\Delta W_2^{\mathrm{full}}$ from 12.75 to 7.52, although scNODE remains closest to the Real–Split scale on this metric alone. We therefore report $\Delta W_2^{\mathrm{full}}$ and do not interpret lower raw transport cost as uniformly indicating better distributional fidelity.

Mechanistic conditioning stabilizes long-horizon extrapolation. To evaluate the *conditioning bottleneck* hypothesis, we perform a controlled experiment on iPSC-CM in which the transcriptome generator is held fixed and only the future conditioning state is varied. We compare (i) MeCoGen's mechanistically extrapolated state, (ii) a data-driven condition obtained by forecasting the core genes with scNODE and assigning states with a classifier trained only on observed cells, and (iii) the true target core genes and states as an oracle-conditioning reference.

Table 3: Generative and transport ablations on iPSC-CM, averaged over D11/D15. The first three rows progressively introduce joint occurrence and positive-magnitude modeling; the remaining rows vary λ_{OT} for MeCoGen-OT. All metrics are computed on the downstream genes. Zero denotes mean per-gene zero-fraction error.

Variant	MMD$\times 10^3$ ↓	Energy↓	KS↓	Zero↓	$\mathbf{W_2}$	$\mathbf{\Delta W_2}$ ↓
Independent	69.90 ± 0.08	1.382 ± 0.002	0.320 ± 0.000	0.0462 ± 0.0006	56.69 ± 0.04	12.14 ± 0.04
Joint-Occ.	7.66 ± 0.04	0.158 ± 0.001	0.308 ± 0.001	0.0569 ± 0.0006	43.79 ± 0.04	0.82 ± 0.06
MeCoGen	6.22 ± 0.07	0.123 ± 0.001	0.193 ± 0.000	0.0569 ± 0.0006	50.38 ± 0.06	5.83 ± 0.06
MeCoGen-OT (10^{-3})	5.87 ± 0.08	0.116 ± 0.002	0.183 ± 0.001	0.0505 ± 0.0003	48.59 ± 0.04	4.04 ± 0.04
MeCoGen-OT (10^{-2})	20.01 ± 0.17	0.433 ± 0.004	0.290 ± 0.001	0.0505 ± 0.0003	39.69 ± 0.07	4.86 ± 0.07
MeCoGen-OT (10^{-1})	21.25 ± 0.09	0.468 ± 0.001	0.307 ± 0.000	0.0504 ± 0.0003	38.34 ± 0.05	6.21 ± 0.05

As shown in Table 2, the advantage of mechanistic conditioning grows with forecast horizon. At $+4$ days, it reduces state-distribution error by nearly an order of magnitude relative to the data-driven condition (0.052 versus 0.481). At $+8$ days, the mechanistic core W_2 increases only modestly ($3.15 \rightarrow 3.66$), whereas the data-driven value rises sharply ($6.96 \rightarrow 19.44$). After the same downstream generator, the mechanistic $\mathbf{W}_{\mathbf{2}}^{\mathrm{full}}$ gap to the oracle remains nearly unchanged ($6.18 \rightarrow 6.37$), while the data-driven gap triples ($17.55 \rightarrow 53.82$).

Joint stochastic modeling and transport regularization play complementary roles. We next isolate the contribution of the downstream generator. The Independent variant models gene occurrence and positive-expression magnitude independently, Joint-Occ. introduces joint occurrence modeling via conditional diffusion, and MeCoGen additionally models positive-expression magnitudes jointly through conditional flow matching. Thus, MeCoGen combines joint modeling of both components while preserving their distinct probabilistic structure. The results are shown in Table 3.

Joint occurrence modeling substantially improves distributional fidelity, reducing MMD from 69.90×10^{-3} to 7.66×10^{-3} and Energy distance from 1.382 to 0.158. Joint modeling of positive-expression magnitudes further improves MMD, Energy, and mean KS to 6.22×10^{-3}, 0.123, and 0.193, respectively, demonstrating that both occurrence and magnitude contribute to recovering the stochastic structure of future populations.

The MeCoGen-OT sweep reveals a distinct fidelity–geometry trade-off. Weak regularization at $\lambda_{\mathrm{OT}} = 10^{-3}$ reduces ΔW_2 from 5.83 to 4.04 while retaining strong MMD, Energy, and KS performance (5.87×10^{-3}, 0.116, and 0.183, respectively). Increasing λ_{OT} further lowers raw W_2 but moves farther from the empirical Real–Split reference, with ΔW_2 increasing to 4.86 and 6.21 and substantial degradation in MMD and Energy. A similar trade-off is observed on WOT-MEF, where transport regularization reduces ΔW_2 from 12.75 to 7.52 at a modest cost in distributional fidelity. These results suggest that OT regularization is most effective as a geometry-calibration signal that aligns global population structure, rather than as the primary forecasting objective.

Finally, leave-one-timepoint-out experiments on SC-islet demonstrate robustness to temporal split selection. MeCoGen achieves the best MMD, Energy distance, mean KS, and zero-fraction error across all held-out time points from D2 to D7, including the out-of-time D7 forecast (Appendix H).

6 Conclusions and Discussion

We introduced **MeCoGen**, a hybrid framework that combines mechanistic state extrapolation with conditional transcriptome generation for out-of-time single-cell forecasting. Across three temporal systems, MeCoGen consistently improves long-horizon forecasts and demonstrates that mechanistically grounded state representations are more robust than purely data-driven alternatives under distribution shift. Ablation studies further highlight the benefits of separating gene occurrence from positive-expression magnitude and show that transport-based regularization provides complementary gains in distributional fidelity. While the current framework relies on a compact regulatory state and curated biological priors, it naturally extends to richer regulatory mechanisms, perturbation settings, and multimodal data. Overall, our results support a general strategy for scientific generative forecasting: use mechanistic structure for extrapolation and flexible generative models for unresolved high-dimensional stochastic variation.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript editing, language refinement, organization of scientific writing, and literature search and cross-checking. They were also used to assist with code development and debugging for selected analyses and visualizations. All AI-assisted text, references, code, analyses, and scientific claims were reviewed and verified by the authors against the underlying literature, implementations, data, and experimental results. Generative AI was not used to autonomously conduct experiments, determine the reported results, or make final scientific conclusions. The authors take full responsibility for the content of the manuscript, including all text, claims, analyses, and artifacts produced with the assistance of generative AI.

ETHICS STATEMENT

This work develops computational methods using previously published single-cell transcriptomic datasets. No new human or animal subjects were recruited and no new biological samples were collected as part of this study. The original studies are cited in the manuscript, and ethical oversight and sample collection procedures are described in those publications. The proposed method is intended for computational research and hypothesis generation rather than clinical or diagnostic use.

REPRODUCIBILITY STATEMENT

To support reproducibility, we provide: **(1) Data and preprocessing:** all three temporal single-cell datasets are publicly available, with preprocessing, gene selection, and temporal train–test splits documented in the appendix; **(2) Model specification and training:** the complete formulation of MeCoGen is provided in Section 4, with dataset-specific mechanistic specifications, generator architectures, training procedures, optimization hyperparameters, and transport-regularization settings provided in the appendix; **(3) Baselines and evaluation:** we document baseline implementations and configurations, evaluation metrics, temporal protocols, generation seeds, and uncertainty calculations, with all fitting and calibration restricted to the corresponding training window; and **(4) Code:** code for preprocessing, model training, forecasting, and evaluation will be made publicly available after the review process.

REFERENCES

Sara Aibar, Carmen Bravo González-Blas, Thomas Moerman, Vân Anh Huynh-Thu, Hana Imrichová, Gert Hulselmans, Florian Rambow, Jean-Christophe Marine, Pierre Geurts, Jan Aerts, et al. SCENIC: single-cell regulatory network inference and clustering. *Nature Methods*, 14: 1083–1086, 2017. doi: 10.1038/nmeth.4463. URL https://doi.org/10.1038/nmeth.4463.

Tal Ashuach, Mariano I. Gabitto, Rohan V. Koodli, Giuseppe-Antonio Saldi, Michael I. Jordan, and Nir Yosef. MultiVI: deep generative model for the integration of multimodal data. *Nature Methods*, 20:1222–1231, 2023. doi: 10.1038/s41592-023-01909-9. URL https://doi.org/10.1038/s41592-023-01909-9.

Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems*, volume 34, pp. 17981–17993, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/958c530554f78bcd8e97125b70e6973d-Abstract.html.

Mehmet Yigit Balik and Harri Lähdesmäki. Modeling temporal scrna-seq data with latent gaussian process and optimal transport. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=8RE5i7g4uB.

Volker Bergen, Marius Lange, Stefan Peidli, F. Alexander Wolf, and Fabian J. Theis. Generalizing RNA velocity to transient cell states through dynamical modeling. *Nature Biotechnology*, 38: 1408–1414, 2020. doi: 10.1038/s41587-020-0591-3. URL https://doi.org/10.1038/s41587-020-0591-3.

Nuha BinTayyash, Sokratia Georgaka, S. T. John, Sumon Ahmed, Alexis Boukouvalas, James Hensman, and Magnus Rattray. Non-parametric modelling of temporal and spatial counts data from RNA-seq experiments. *Bioinformatics*, 37(21):3788–3795, 2021. doi: 10.1093/bioinformatics/btab486. URL https://doi.org/10.1093/bioinformatics/btab486.

Charlotte Bunne, Stefan G. Stark, Gabriele Gut, Jacobo Sarabia del Castillo, Mitch Levesque, Kjong-Van Lehmann, Lucas Pelkmans, Andreas Krause, and Gunnar Rätsch. Learning single-cell perturbation responses using neural optimal transport. *Nature Methods*, 20:1759–1768, 2023. doi: 10.1038/s41592-023-01969-x. URL https://doi.org/10.1038/s41592-023-01969-x.

Robrecht Cannoodt, Wouter Saelens, Louise Deconinck, and Yvan Saeys. Spearheading future omics analyses using dyngen, a multi-modal simulator of single cells. *Nature Communications*, 12:3942, 2021. doi: 10.1038/s41467-021-24152-2. URL https://doi.org/10.1038/s41467-021-24152-2.

Haotian Cui, Chloe Wang, Hassaan Maan, Kuan Pang, Fengning Luo, et al. scGPT: toward building a foundation model for single-cell multi-omics using generative AI. *Nature Methods*, 21:1470–1480, 2024. doi: 10.1038/s41592-024-02201-0. URL https://doi.org/10.1038/s41592-024-02201-0.

Payam Dibaeinia and Saurabh Sinha. SERGIO: A single-cell expression simulator guided by gene regulatory networks. *Cell Systems*, 11(3):252–271.e11, 2020. doi: 10.1016/j.cels.2020.08.003. URL https://doi.org/10.1016/j.cels.2020.08.003.

Reem Elorbany, Joshua M. Popp, Katherine Rhodes, Benjamin J. Strober, Kenneth Barr, Guanghao Qi, Yoav Gilad, and Alexis Battle. Single-cell sequencing reveals lineage-specific dynamic genetic regulation of gene expression during human cardiomyocyte differentiation. *PLoS Genetics*, 18(1):e1009666, 2022. doi: 10.1371/journal.pgen.1009666.

Gökcen Eraslan, Lukas M. Simon, Maria Mircea, Nikola S. Mueller, and Fabian J. Theis. Single-cell RNA-seq denoising using a deep count autoencoder. *Nature Communications*, 10:390, 2019. doi: 10.1038/s41467-018-07931-2. URL https://doi.org/10.1038/s41467-018-07931-2.

Jean Feydy, Thibault Séjourné, François-Xavier Vialard, Shun-ichi Amari, Alain Trouvé, and Gabriel Peyré. Interpolating between optimal transport and mmd using sinkhorn divergences. In *The 22nd international conference on artificial intelligence and statistics*, pp. 2681–2690. PMLR, 2019.

Greg Finak, Andrew McDavid, Masanao Yajima, Jingyuan Deng, Vivian Gersuk, Alex K. Shalek, Chloe K. Slichter, Hannah W. Miller, M. Juliana McElrath, Martin Prlic, Peter S. Linsley, and Raphael Gottardo. Mast: a flexible statistical framework for assessing transcriptional changes and characterizing heterogeneity in single-cell rna sequencing data. *Genome Biology*, 16:278, 2015. doi: 10.1186/s13059-015-0844-5.

Nikolaus Fortelny and Christoph Bock. Knowledge-primed neural networks enable biologically interpretable deep learning on single-cell sequencing data. *Genome Biology*, 21:190, 2020. doi: 10.1186/s13059-020-02100-5. URL https://doi.org/10.1186/s13059-020-02100-5.

Clémence Fournié, Elias Ventre, Ulysse Herbach, Aymeric Baradat, Olivier Gandrillon, and Fabien Crauste. Cell trajectory inference based on Schrödinger problem and a mechanistic model of stochastic gene expression. *npj Systems Biology and Applications*, 12:120, 2026. doi: 10.1038/s41540-026-00743-x. URL https://doi.org/10.1038/s41540-026-00743-x.

Adam Gayoso, Zoia Steier, Romain Lopez, Jeffrey Regier, Kristopher L. Nazor, Aaron Streets, and Nir Yosef. Joint probabilistic modeling of single-cell multi-omic data with totalVI. *Nature Methods*, 18:272–282, 2021. doi: 10.1038/s41592-020-01050-x. URL https://doi.org/10.1038/s41592-020-01050-x.

Daniel T. Gillespie. Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry*, 81(25):2340–2361, 1977. doi: 10.1021/j100540a008. URL https://doi.org/10.1021/j100540a008.

Gennady Gorin, Meichen Fang, Tara Chari, and Lior Pachter. RNA velocity unraveled. *PLOS Computational Biology*, 18(9):e1010492, 2022. doi: 10.1371/journal.pcbi.1010492. URL https://doi.org/10.1371/journal.pcbi.1010492.

Laleh Haghverdi, Maren Büttner, F. Alexander Wolf, Florian Buettner, and Fabian J. Theis. Diffusion pseudotime robustly reconstructs lineage branching. *Nature Methods*, 13:845–848, 2016. doi: 10.1038/nmeth.3971. URL https://doi.org/10.1038/nmeth.3971.

Minsheng Hao, Jing Gong, Xin Zeng, Chiming Liu, Yucheng Guo, Xingyi Cheng, Taifeng Wang, Jianzhu Ma, Xuegong Zhang, and Le Song. Large-scale foundation model on single-cell transcriptomics. *Nature Methods*, 21:1481–1491, 2024. doi: 10.1038/s41592-024-02305-7. URL https://doi.org/10.1038/s41592-024-02305-7.

Ulysse Herbach, Arnaud Bonnaffoux, Thibault Espinasse, and Olivier Gandrillon. Inferring gene regulatory networks from single-cell data: a mechanistic approach. *BMC Systems Biology*, 11:105, 2017. doi: 10.1186/s12918-017-0487-0. URL https://doi.org/10.1186/s12918-017-0487-0.

Leon Hetzel, Simon Böhm, Niki Kilbertus, Stephan Günnemann, Mohammad Lotfollahi, and Fabian J. Theis. Predicting cellular responses to novel drug perturbations at a single-cell resolution. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/aa933b5abc1be30baece1d230ec575a7-Abstract-Conference.html.

Guillaume Huguet, Daniel Sumner Magruder, Alexander Tong, Oluwadamilola Fasina, Manik Kuchroo, Guy Wolf, and Smita Krishnaswamy. Manifold interpolating optimal-transport flows for trajectory inference. In *Advances in Neural Information Processing Systems*, volume 35, pp. 29705–29718, 2022. doi: 10.52202/068431-2154. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/bfc03f077688d8885c0a9389d77616d0-Abstract-Conference.html.

Qi Jiang and Lin Wan. A physics-informed neural SDE network for learning cellular dynamics from time-series scRNA-seq data. *Bioinformatics*, 40(Supplement_2):ii120–ii127, 2024. doi: 10.1093/bioinformatics/btae400. URL https://doi.org/10.1093/bioinformatics/btae400.

Qi Jiang, Longquan Li, Lei Zhang, and Lin Wan. Learning collective multicellular dynamics with an interacting mean field neural sde model. *PLoS Computational Biology*, 22(1):e1013916, 2026. doi: 10.1371/journal.pcbi.1013916.

Dominik Klein, Giovanni Palla, Marius Lange, Michal Klein, Zoe Piran, Matthias Gander, Laetitia Meng-Papaxanthos, Michael Sterr, Aimée Bastidas-Ponce, Marta Tarquis-Medina, et al. Mapping cells through time and space with moscot. *Nature*, 638:1065–1075, 2025. doi: 10.1038/s41586-024-08453-2. URL https://doi.org/10.1038/s41586-024-08453-2.

Gioele La Manno, Ruslan Soldatov, Amit Zeisel, Emelie Braun, Hannah Hochgerner, Viktor Petukhov, Katja Lidschreiber, Maria E. Kastriti, Peter Lönnerberg, Alessandro Furlan, et al. RNA velocity of single cells. *Nature*, 560:494–498, 2018. doi: 10.1038/s41586-018-0414-6. URL https://doi.org/10.1038/s41586-018-0414-6.

Marius Lange, Volker Bergen, Michal Klein, Manu Setty, Bernhard Reuter, Mostafa Bakhti, Heiko Lickert, Meshal Ansari, Janine Schniering, Herbert B. Schiller, et al. CellRank for directed single-cell fate mapping. *Nature Methods*, 19:159–170, 2022. doi: 10.1038/s41592-021-01346-6. URL https://doi.org/10.1038/s41592-021-01346-6.

Qian Li. scTour: a deep learning architecture for robust inference and accurate prediction of cellular dynamics. *Genome Biology*, 24:149, 2023. doi: 10.1186/s13059-023-02988-9. URL https://doi.org/10.1186/s13059-023-02988-9.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Romain Lopez, Jeffrey Regier, Michael B. Cole, Michael I. Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature Methods*, 15:1053–1058, 2018. doi: 10.1038/s41592-018-0229-2. URL https://doi.org/10.1038/s41592-018-0229-2.

Mohammad Lotfollahi, F. Alexander Wolf, and Fabian J. Theis. scGen predicts single-cell perturbation responses. *Nature Methods*, 16:715–721, 2019. doi: 10.1038/s41592-019-0494-8. URL https://doi.org/10.1038/s41592-019-0494-8.

Mohammad Lotfollahi, Mohsen Naghipourfar, Malte D. Luecken, Matin Khajavi, Maren Büttner, Melanie Wagenstetter, Žiga Avsec, Adam Gayoso, Nir Yosef, Marta Interlandi, et al. Mapping single-cell data to reference atlases by transfer learning. *Nature Biotechnology*, 40:121–130, 2022. doi: 10.1038/s41587-021-01001-7. URL https://doi.org/10.1038/s41587-021-01001-7.

Mohammad Lotfollahi, Anna Klimovskaia Susmelj, Carlo De Donno, Leon Hetzel, Yuge Ji, Ignacio L. Ibarra, Sanjay R. Srivatsan, Mohsen Naghipourfar, Riza M. Daza, Beth Martin, et al. Predicting cellular responses to complex perturbations in high-throughput screens. *Molecular Systems Biology*, 19(6):e11517, 2023. doi: 10.15252/msb.202211517. URL https://doi.org/10.15252/msb.202211517.

Erpai Luo, Minsheng Hao, Lei Wei, and Xuegong Zhang. scDiffusion: conditional generation of high-quality single-cell data using diffusion model. *Bioinformatics*, 40(9):btae518, 2024. doi: 10.1093/bioinformatics/btae518. URL https://doi.org/10.1093/bioinformatics/btae518.

Mohamed Marouf, Pierre Machart, Vikas Bansal, Christoph Kilian, Daniel S. Magruder, Christian F. Krebs, and Stefan Bonn. Realistic in silico generation and augmentation of single-cell RNA-seq data using generative adversarial networks. *Nature Communications*, 11:166, 2020. doi: 10.1038/s41467-019-14018-z. URL https://doi.org/10.1038/s41467-019-14018-z.

Brian Munsky, Gregor Neuert, and Alexander van Oudenaarden. Using gene expression noise to understand gene regulation. *Science*, 336(6078):183–187, 2012. doi: 10.1126/science.1216379. URL https://doi.org/10.1126/science.1216379.

Adrien Osakwe, Eric H. Huang, and Yue Li. scTimeBench: A streamlined benchmarking platform for single-cell time-series analysis. *Bioinformatics*, 42(Supplement_2):btag414, 2026. doi: 10.1093/bioinformatics/btag414. URL https://doi.org/10.1093/bioinformatics/btag414.

Alessandro Palma, Till Richter, Hanyi Zhang, Manuel Lubetzki, Alexander Tong, Andrea Dittadi, and Fabian Theis. Multi-modal and multi-attribute generation of single cells with CFGen. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/13079a5dd5c994a7e5f0cc691e0338e8-Abstract-Conference.html.

Jean Peccoud and Bernard Ycart. Markovian modeling of gene-product synthesis. *Theoretical Population Biology*, 48(2):222–234, 1995. doi: 10.1006/tpbi.1995.1027. URL https://doi.org/10.1006/tpbi.1995.1027.

Xiaojie Qiu, Qi Mao, Ying Tang, Li Wang, Raghav Chawla, Hannah A. Pliner, and Cole Trapnell. Reversed graph embedding resolves complex single-cell trajectories. *Nature Methods*, 14:979–982, 2017. doi: 10.1038/nmeth.4402. URL https://doi.org/10.1038/nmeth.4402.

Xiaojie Qiu, Yan Zhang, James D. Martin-Rufino, Chen Weng, Sébastien Hosseinzadeh, Dian Yang, Ania N. Pogson, Marco Y. Hein, K. Hoi Joseph Min, Lijia Wang, et al. Mapping transcriptomic vector fields of single cells. *Cell*, 185(4):690–711.e45, 2022. doi: 10.1016/j.cell.2021.12.045. URL https://doi.org/10.1016/j.cell.2021.12.045.

Arjun Raj, Charles S. Peskin, Daniel Tranchina, Diana Y. Vargas, and Sanjay Tyagi. Stochastic mrna synthesis in mammalian cells. *PLOS Biology*, 4(10):e309, 2006. doi: 10.1371/journal.pbio.0040309. URL https://doi.org/10.1371/journal.pbio.0040309.

Davide Risso, Fanny Perraudeau, Svetlana Gribkova, Sandrine Dudoit, and Jean-Philippe Vert. A general and flexible method for signal extraction from single-cell RNA-seq data. *Nature Communications*, 9:284, 2018. doi: 10.1038/s41467-017-02554-5. URL https://doi.org/10.1038/s41467-017-02554-5.

Yusuf Roohani, Kexin Huang, and Jure Leskovec. Predicting transcriptional outcomes of novel multigene perturbations with GEARS. *Nature Biotechnology*, 42:927–935, 2024. doi: 10.1038/s41587-023-01905-6. URL https://doi.org/10.1038/s41587-023-01905-6.

Wouter Saelens, Robrecht Cannoodt, Helena Todorov, and Yvan Saeys. A comparison of single-cell trajectory inference methods. *Nature Biotechnology*, 37:547–554, 2019. doi: 10.1038/s41587-019-0071-9. URL https://doi.org/10.1038/s41587-019-0071-9.

Moisés Santillán. On the use of the hill functions in mathematical models of gene regulatory networks. *Mathematical Modelling of Natural Phenomena*, 3(2):85–97, 2008.

Geoffrey Schiebinger, Jian Shu, Marcin Tabaka, Brian Cleary, Vidya Subramanian, Aryeh Solomon, Joshua Gould, Siyan Liu, Stacie Lin, Peter Berube, Lia Lee, Jenny Chen, Justin Brumbaugh, Philippe Rigollet, Konrad Hochedlinger, Rudolf Jaenisch, Aviv Regev, and Eric S. Lander. Optimal-transport analysis of single-cell gene expression identifies developmental trajectories in reprogramming. *Cell*, 176(4):928–943.e22, 2019. doi: 10.1016/j.cell.2019.01.006. URL https://doi.org/10.1016/j.cell.2019.01.006.

Manu Setty, Vaidotas Kiseliovas, Jacob Levine, Adam Gayoso, Linas Mazutis, and Dana Pe'er. Characterization of cell fate probabilities in single-cell data with Palantir. *Nature Biotechnology*, 37:451–460, 2019. doi: 10.1038/s41587-019-0068-4. URL https://doi.org/10.1038/s41587-019-0068-4.

Yutong Sha, Yuchi Qiu, Peijie Zhou, and Qing Nie. Reconstructing growth and dynamic trajectories from single-cell transcriptomics data. *Nature Machine Intelligence*, 6:25–39, 2024. doi: 10.1038/s42256-023-00763-w. URL https://doi.org/10.1038/s42256-023-00763-w.

Vahid Shahrezaei and Peter S. Swain. Analytical distributions for stochastic gene expression. *Proceedings of the National Academy of Sciences*, 105(45):17256–17261, 2008. doi: 10.1073/pnas.0803850105. URL https://doi.org/10.1073/pnas.0803850105.

Dongyuan Song, Qingyang Wang, Guanao Yan, Tianyang Liu, Tianyi Sun, and Jingyi Jessica Li. scDesign3 generates realistic in silico data for multimodal single-cell and spatial omics. *Nature Biotechnology*, 42:247–252, 2024. doi: 10.1038/s41587-023-01772-1. URL https://doi.org/10.1038/s41587-023-01772-1.

Kelly Street, Davide Risso, Russell B. Fletcher, Diya Das, John Ngai, Nir Yosef, Elizabeth Purdom, and Sandrine Dudoit. Slingshot: cell lineage and pseudotime inference for single-cell transcriptomics. *BMC Genomics*, 19:477, 2018. doi: 10.1186/s12864-018-4772-0. URL https://doi.org/10.1186/s12864-018-4772-0.

Tianyi Sun, Dongyuan Song, Wei Vivian Li, and Jingyi Jessica Li. scDesign2: a transparent simulator that generates high-fidelity single-cell gene expression count data with gene correlations captured. *Genome Biology*, 22:163, 2021. doi: 10.1186/s13059-021-02367-2. URL https://doi.org/10.1186/s13059-021-02367-2.

David M. Suter, Nacho Molina, David Gatfield, Kim Schneider, Ueli Schibler, and Felix Naef. Mammalian genes are transcribed with widely different bursting kinetics. *Science*, 332(6028):472–474, 2011. doi: 10.1126/science.1198817. URL https://doi.org/10.1126/science.1198817.

Christina V. Theodoris, Ling Xiao, Anant Chopra, Mark D. Chaffin, Zain Al Sayed, Matthew C. Hill, Heather Mantineo, Elizabeth M. Brydon, Zexian Zeng, X. Shirley Liu, and Patrick T.

Ellinor. Transfer learning enables predictions in network biology. *Nature*, 618:616–624, 2023. doi: 10.1038/s41586-023-06139-9. URL https://doi.org/10.1038/s41586-023-06139-9.

Alexander Tong, Jessie Huang, Guy Wolf, David Van Dijk, and Smita Krishnaswamy. TrajectoryNet: A dynamic optimal transport network for modeling cellular dynamics. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9526–9536. PMLR, 2020. URL https://proceedings.mlr.press/v119/tong20a.html.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Transactions on Machine Learning Research*, 2024a. URL https://openreview.net/forum?id=CD9Snc73AW.

Alexander Y. Tong, Nikolay Malkin, Kilian Fatras, Lazar Atanackovic, Yanlei Zhang, Guillaume Huguet, Guy Wolf, and Yoshua Bengio. Simulation-free Schrödinger bridges via score and flow matching. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 1279–1287. PMLR, 2024b. URL https://proceedings.mlr.press/v238/tong24a.html.

Cole Trapnell, Davide Cacchiarelli, Jonna Grimsby, Prapti Pokharel, Shuqiang Li, Michael Morse, Niall J. Lennon, Kenneth J. Livak, Tarjei S. Mikkelsen, and John L. Rinn. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nature Biotechnology*, 32:381–386, 2014. doi: 10.1038/nbt.2859. URL https://doi.org/10.1038/nbt.2859.

Elias Ventre. Reverse engineering of a mechanistic model of gene expression using metastability and temporal dynamics. *In Silico Biology*, 14:89–113, 2021. doi: 10.3233/ISB-210226. URL https://doi.org/10.3233/ISB-210226.

Elias Ventre, Ulysse Herbach, Thibault Espinasse, Gérard Benoit, and Olivier Gandrillon. One model fits all: Combining inference and simulation of gene regulatory networks. *PLOS Computational Biology*, 19(3):e1010962, 2023. doi: 10.1371/journal.pcbi.1010962. URL https://doi.org/10.1371/journal.pcbi.1010962.

Adrian Veres, Aubrey L. Faust, Henry L. Bushnell, Elise N. Engquist, Jennifer Hyoje-Ryu Kenty, George Harb, Yeh-Chuin Poh, Elad Sintov, Mads Gürtler, Felicia W. Pagliuca, Quinn P. Peterson, and Douglas A. Melton. Charting cellular identity during human in vitro beta-cell differentiation. *Nature*, 569(7756):368–373, 2019. doi: 10.1038/s41586-019-1168-5.

Michael E. Vinyard, Anders W. Rasmussen, Ruitong Li, Allon M. Klein, Gad Getz, and Luca Pinello. Learning cell dynamics with neural differential equations. *Nature Machine Intelligence*, 7:1969–1984, 2025. doi: 10.1038/s42256-025-01150-3. URL https://doi.org/10.1038/s42256-025-01150-3.

Caleb Weinreb, Samuel Wolock, Betsabeh K. Tusi, Merav Socolovsky, and Allon M. Klein. Fundamental limits on dynamic inference from single-cell snapshots. *Proceedings of the National Academy of Sciences*, 115(10):E2467–E2476, 2018. doi: 10.1073/pnas.1714723115. URL https://doi.org/10.1073/pnas.1714723115.

Addie Woicik, Mingxin Zhang, Janelle Chan, Jianzhu Ma, and Sheng Wang. Extrapolating heterogeneous time-series gene expression data using Sagittarius. *Nature Machine Intelligence*, 5:699–713, 2023. doi: 10.1038/s42256-023-00679-5. URL https://doi.org/10.1038/s42256-023-00679-5.

F. Alexander Wolf, Philipp Hamey, Mireya Plass, Jordi Solana, Joakim S. Dahlin, Berthold Göttgens, Nikolaus Rajewsky, Lukas Simon, and Fabian J. Theis. PAGA: graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells. *Genome Biology*, 20:59, 2019. doi: 10.1186/s13059-019-1663-x. URL https://doi.org/10.1186/s13059-019-1663-x.

Grace Hui Ting Yeo, Sachit D. Saksena, and David K. Gifford. Generative modeling of single-cell time series with PRESCIENT enables prediction of cell trajectories with interventions. *Nature Communications*, 12:3222, 2021. doi: 10.1038/s41467-021-23518-w. URL https://doi.org/10.1038/s41467-021-23518-w.

Jiaqi Zhang, Erica Larschan, Jeremy Bigness, and Ritambhara Singh. scNODE: generative model for temporal single cell transcriptomic data prediction. *Bioinformatics*, 40(Supplement_2):ii146–ii154, 2024. doi: 10.1093/bioinformatics/btae393. URL https://doi.org/10.1093/bioinformatics/btae393.

A MECHANISTIC MODEL SPECIFICATION

This section provides the dataset-specific parameterization and numerical details of the structured-state model in Section 4.1. For every temporal split, all parameters described below are estimated, selected, or calibrated using only the permitted training populations.

A.1 REGULATORY PRODUCTION AND BURSTING STOCHASTICITY

Normalized Hill regulation. For gene g, let $\mathcal{A}_g$ and $\mathcal{I}_g$ denote its activating and inhibitory regulators, respectively. The GRN-driven production rate is represented by the normalized Hill function

$$p_g(\boldsymbol{x}, s, \boldsymbol{u}) = g_{s,g}(\boldsymbol{u})\, \alpha_g \frac{\rho_g + \sum_{j \in \mathcal{A}_g} k_{jg} x_j^{h_g}}{1 + \sum_{j \in \mathcal{A}_g} x_j^{h_g} + \sum_{\ell \in \mathcal{I}_g} x_\ell^{h_g}}, \tag{18}$$

where ρ_g is the basal regulatory term, α_g sets the production scale, k_{jg} specifies the strength of activating interaction $j \to g$, h_g is the Hill coefficient, and $g_{s,g}(\boldsymbol{u})$ captures developmental-state- and protocol-dependent modulation.

Chemical-Langevin bursting. For genes assigned to the Chemical-Langevin (CLE) regime, transcriptional bursting is represented through the parameter β_g, giving the production-noise term

$$\sqrt{\frac{2p_g(\boldsymbol{x}, s, \boldsymbol{u})}{\beta_g}}\, dW_g^{\mathrm{prod}}(t). \tag{19}$$

Thus, the production-associated variance rate in this branch is $2p_g/\beta_g$. When raw count information is available, β_g is informed by gene-level count dispersion. Dataset-specific regime assignments and stochastic parameter settings are given in Appendix A.2.

Numerical integration. The CLE branch is integrated using Euler–Maruyama. Diffusion amplitudes are evaluated using $\max(p_g, 0)$ and $\max(x_g, 0)$, with a numerical floor of 10^{-8} inside the square root. After each step, expression states are projected onto the nonnegative domain, $x_g \leftarrow \max(0, x_g)$, so transient negative Euler proposals are not propagated into subsequent drift or diffusion evaluations.

A.2 LOW-COPY COMPOUND-POISSON BURSTING

For genes assigned to the low-copy regime, the continuous CLE production update is replaced by an explicit compound-Poisson bursting process:

$$N_g \sim \mathrm{Poisson}\left(\frac{p_g}{b_g}\Delta t\right), \qquad J_g \mid N_g \sim \mathrm{Gamma}(N_g, b_g), \tag{20}$$

where b_g is the mean burst size (the scale parameter of the Gamma distribution). This parameterization preserves the mean production increment, $\mathbb{E}[J_g] = p_g \Delta t$, while giving $\mathrm{Var}(J_g) = 2p_g b_g \Delta t$.

The expression state is updated as

$$x_g(t + \Delta t) = x_g(t) e^{-\gamma_g(s_i(t))\Delta t} + J_g. \tag{21}$$

Values below 4×10^{-4} are snapped to zero after the update. This threshold is numerical and does not introduce an additional biological state.

The CLE parameter β_g and the compound-Poisson parameter b_g are distinct parameterizations of production-associated bursting stochasticity: β_g controls the continuous production-noise variance in the CLE branch, whereas b_g is the mean discrete burst size in the low-copy branch. The two branches are mutually exclusive for a given gene.

Regime assignment. The stochastic regime is fixed for each core gene rather than selected dynamically from its instantaneous expression level. WOT-MEF and SC-islet use the low-copy branch for all core genes (41/41 and 45/45, respectively). For iPSC-CM, nine genes (*OCT4*, *TBXT*, *MESP1*, *NKX2-5*, *GATA4*, *TNNT2*, *MYL7*, *MEF2C*, and *NANOG*) use the low-copy branch, while the remaining 24 core genes use the CLE branch. The WOT-MEF assignment was selected from candidate low-copy sets using training-day mean per-gene KS and then held fixed across temporal splits.

When raw count information is available, we define the dispersion-derived stochastic scale $\eta_g = \frac{2}{\mathrm{Fano}_g}$. For genes assigned to the CLE branch, we set $\beta_g = \eta_g$, whereas for genes assigned to the low-copy branch, we set $b_g = \eta_g$. Thus, the same dispersion-derived numerical scale is used under the two branch-specific stochastic parameterizations.

dataset	core genes	low-copy	CLE	stochastic parameter
WOT-MEF	41	41	0	shared $b_g = 0.05$
SC-islet	45	45	0	$b_g = \eta_g$
iPSC-CM	33	9	24	$b_g = \eta_g$ or $\beta_g = \eta_g$, by regime

For WOT-MEF, raw UMI counts are unavailable, so the low-copy branch instead uses a shared value $b_g = 0.05$. Its training-day calibration is weakly sensitive over the tested range $b_g \in [0.02, 0.1]$.

A.3 State-Transition Hazard Parameterization

For each admissible developmental edge $e = (s, s')$, the transition rate is

$$q_{e,i}(t) = q_e^{\max} \Phi_e(\boldsymbol{x}_i; k_e) \eta_{e,i} + \beta_e t + \kappa_e u(t), \tag{22}$$

where the transition-specific regulatory propensity is represented by a Hill function,

$$\Phi_e(\boldsymbol{x}_i; k_e) = \frac{s_{e,i}^H}{k_e^H + s_{e,i}^H}, \qquad s_{e,i} = \sum_{g \in \mathcal{G}_e} x_{i,g}, \tag{23}$$

with $\mathcal{G}_e$ denoting the set of trigger genes associated with transition e, k_e the corresponding half-activation threshold, and H the Hill coefficient. The term $\beta_e t$ captures edge-specific temporal progression, while $\kappa_e u(t)$ represents protocol-dependent modulation where applicable.

The multiplicative factor $\eta_{e,i}$ is a cell-specific log-normal random effect drawn once for each simulated cell and retained over its trajectory, thereby representing persistent rather than step-wise heterogeneity. Transitions not contained in the curated developmental topology are assigned zero rate. The topology, trigger-gene sets, and protocol-sensitive edges are specified separately for each biological system.

A.4 Population Expansion and Depletion

When differential population expansion or depletion is included, each simulated cell is assigned a population weight evolving according to

$$\frac{dw_i(t)}{dt} = r_{s_i(t)}(t) w_i(t). \tag{24}$$

The state-dependent net rate is parameterized using up to two logistic phases,

$$r_s(t) = r_s^{(1)} \sigma\left(\frac{t - t_s^{(1)}}{\tau}\right) + r_s^{(2)} \sigma\left(\frac{t - t_s^{(2)}}{\tau}\right), \tag{25}$$

where $r_s^{(1)}$ and $r_s^{(2)}$ are signed amplitudes, $t_s^{(1)}$ and $t_s^{(2)}$ are state-specific temporal midpoints, and τ is a shared transition width. Positive amplitudes represent net expansion and negative amplitudes net depletion.

The rate is re-evaluated at every integration step from the cell's current state. Consequently, both continuous temporal progression and discrete state transitions can alter a cell's subsequent population contribution.

At prediction time, cells are sampled according to

$$\bar{w}_i(t) = \frac{w_i(t)}{\sum_j w_j(t)} \tag{26}$$

while retaining $(\boldsymbol{x}_i, s_i)$ jointly. For datasets using a fixed simulated population without an explicit growth/depletion component, $w_i(t) \equiv 1$ and this step reduces to uniform joint sampling.

B MECHANISTIC MODEL FITTING AND BIOLOGICAL PRIORS

This section describes how the mechanistic model is fitted to the observed training populations. We distinguish fixed biological priors from fitted kinetic parameters, define the calibration objective and optimization procedure, and specify the dataset-specific developmental-state and gene regulatory network (GRN) topologies. For each temporal split, fitted parameters are estimated using only observations within the corresponding training window.

B.1 FIXED PRIORS AND FITTED PARAMETERS

The mechanistic model separates structural biological priors from kinetic parameters estimated from data.

Structural biological priors. The core-gene panel, developmental-state set and admissible transitions, transition-specific trigger genes, signed GRN topology, Hill exponents, and low-copy gene assignments are specified before kinetic parameter fitting and held fixed across temporal splits. These quantities define the biological structure of the simulator and are not optimized against the forecast target. Dataset-specific developmental-state graphs and GRN specifications are given in Section B.4.

Fitted kinetic parameters. For each temporal split, a low-dimensional parameter vector θ is estimated from the corresponding training populations. The fitted parameters control developmental-state transition intensities, transition gating, cell-to-cell heterogeneity, protocol effects where applicable, and population growth or depletion. Table 4 summarizes the parameter blocks.

Table 4: Free parameters of the mechanistic model. E denotes the number of admissible developmental-state transitions and S_g the number of states with fitted population-growth parameters.

Parameter block	Meaning	Dimension	Datasets
$\log q_{\max,e}$	maximum transition intensity	E	all
$\log \lambda_0$	state-transition leak intensity	1	WOT-MEF, iPSC-CM
$\log \sigma_{\mathrm{het}}$	cell-level transition heterogeneity	1	all
$\log k_{\mathrm{trig},e}$	trigger half-saturation	E	all
$\log \kappa_c$	protocol-condition effects	$3/1/0$	WOT-MEF/iPSC-CM/SC-islet
β_e	trigger-score exponent adjustment	E	WOT-MEF, iPSC-CM
$(r_{\max,s}, t_{0,s})$	population growth/depletion parameters	$2S_g$	WOT-MEF

The resulting parameter vectors contain 33 parameters for WOT-MEF, 18 for iPSC-CM, and 13 for SC-islet. For SC-islet, the EndoPrec→SC-α transition is inactive in the adopted state graph, leaving 11 active optimization coordinates.

For WOT-MEF, the protocol terms comprise $\kappa_{2\mathrm{i}}$ on MET→IPS, κ_{serum} on Epithelial→Neural, and $\kappa_{\mathrm{serum,epi}}$ on MET→Epithelial. iPSC-CM uses a single κ_{WntC59} term. SC-islet uses no protocol-specific covariate because its two experimental arms are treated as replicates of the same differentiation protocol.

B.2 CALIBRATION OBJECTIVE

The mechanistic simulator defines a stochastic population process and does not provide a tractable likelihood over observed cells. We therefore estimate its kinetic parameters by matching simulated and observed developmental-state compositions.

Let $f_{ds}(\theta)$ denote the simulated fraction of cells in developmental state s at training day d, obtained by propagating a population of N_{cal} simulated cells, and let $\hat{f}_{ds}$ denote the corresponding empirical fraction among state-annotated cells. The calibration objective is

$$\mathcal{L}(\theta) = \frac{1}{|\mathcal{D}_{\mathrm{tr}}|S} \sum_{d\in\mathcal{D}_{\mathrm{tr}}} \sum_{s=1}^{S} \left(\sqrt{f_{ds}(\theta)} - \sqrt{\hat{f}_{ds}} \right)^2, \tag{27}$$

where $\mathcal{D}_{\mathrm{tr}}$ contains only the observed training days for the corresponding temporal split. The square-root transformation reduces the dominance of highly abundant states and increases sensitivity to lower-frequency fates.

For WOT-MEF, state-composition discrepancies are evaluated under the corresponding culture condition at each training day; combinations absent from either the simulated or observed population are excluded rather than zero-filled. For SC-islet, the objective additionally includes a small worst-case residual penalty and is averaged over two fixed simulation seeds. Specifically,

$$\mathcal{L}_{\mathrm{islet}}(\theta) = \frac{1}{2}\sum_{k=1}^{2}\left[\overline{e^{(k)}} + 0.05\max_{d,s} e_{ds}^{(k)}\right], \tag{28}$$

where $e_{ds}^{(k)} = (\sqrt{f_{ds}^{(k)}(\theta)} - \sqrt{\hat{f}_{ds}})^2$. Day 0 is treated as the initial condition and excluded from this objective.

We use $N_{\mathrm{cal}} = 200$ simulated cells for WOT-MEF and iPSC-CM and $N_{\mathrm{cal}} = 300$ for SC-islet. Simulation seeds are fixed during optimization (one seed for WOT-MEF and iPSC-CM and two for SC-islet), making the numerical objective deterministic with respect to θ.

B.3 Optimization and initialization

Because the calibration objective is evaluated through stochastic population simulation with discrete state transitions, we use derivative-free optimization.

WOT-MEF and iPSC-CM. Equation 27 is minimized using adaptive Nelder–Mead (`scipy.optimize.minimize`) with $x_{\mathrm{tol}} = 3 \times 10^{-2}$ and $f_{\mathrm{tol}} = 10^{-2}$. WOT-MEF uses a budget of 150 iterations and at most 3000 objective evaluations, whereas iPSC-CM uses 300 iterations.

SC-islet. We use cyclic coordinate search over the 11 active coordinates. For each coordinate j, candidate offsets

$$\delta \in \{-1, -0.5, -0.25, 0, 0.25, 0.5, 1\}$$

are evaluated in log-parameter space and the best candidate is retained when it decreases the objective by more than 10^{-4}. Three complete sweeps are performed, with the step grid halved after each sweep.

Initialization. SC-islet uses a training-data-derived initialization. For each developmental edge, $q_{\max,e}$ is initialized from the observed training-window change in destination-state abundance relative to source-state occupancy, while $k_{\mathrm{trig},e}$ is initialized from the corresponding trigger-gene activity in source-state cells.

All fitted parameter vectors are stored separately for each temporal split and simulator version. No held-out forecast population is used in the calibration objective or optimization.

B.4 Dataset-specific biological priors

The mechanistic model uses two forms of dataset-specific biological structure: a developmental-state transition graph and a signed regulatory network over the mechanistic core genes. Both are fixed before kinetic parameter fitting.

B.4.1 Developmental-state topology

Each admissible developmental transition is associated with a set of regulatory trigger genes whose normalized expression modulates the transition propensity through the Hill-type gating function described in Appendix A.3. Transitions driven primarily by an experimental input are indicated separately.

Table 5: Dataset-specific developmental-state transition priors. Each admissible transition and its trigger-gene set are fixed before kinetic parameter fitting. `None` denotes transitions driven primarily by an experimental input rather than an endogenous trigger score.

Dataset	Transition	Trigger genes
WOT-MEF	Stromal → MET	`None` (Dox)
	MET → Stromal	*Zeb1, Zeb2, Snai1, Snai2, Twist1*
	MET → IPS	*Nanog, Tfap4, Irf6, Cbfa2t3, Gli2, Ovol1, Esrrb*
	MET → Epithelial	*Cdh1, Epcam*
	Epithelial → Trophoblast	*Cdx2, Elf5, Gata3*
	Epithelial → Neural	*Pax6, Sox1, Nes*
iPSC-CM	IPSC → MES	`None` (CHIR pulse)
	MES → CMES	*TBXT, EOMES*
	CMES → PROG	*MESP1, KDR, PDGFRA*
	PROG → CM	*NKX2-5, GATA4, MEF2C*
	PROG → CF	*PBX1, PITX2, VIM*
SC-islet	Prog → EndoProg	`None` (Notch; *HES1*-repressed)
	EndoProg → EndoPrec	*NEUROG3, NEUROD1*
	EndoPrec → SC-β	*NKX6-1, ISL1, PDX1*
	EndoPrec → SC-EC	*LMX1A, TPH1*
	EndoPrec → SC-α	*ARX, IRX2* (inactive)
	EndoPrec → SST	*HHEX, SST*

The WOT-MEF topology contains a branching developmental graph rooted in the stromal state together with a MET→Stromal reverse transition representing reversion. The iPSC-CM topology explicitly branches from the progenitor state into cardiomyocyte (CM) and cardiac-fibroblast (CF) fates. For SC-islet, the EndoPrec→SC-α transition is inactive in the final specification and is excluded from parameter optimization.

B.4.2 Gene-regulatory topology

Only regulatory interactions included in the dataset-specific prior GRN enter the production functions in the mechanistic simulator. Each interaction is specified by its regulator, target, and sign; genes without an incoming regulatory edge use state-dependent basal production. The adopted networks contain 21 regulatory terms over 41 core genes for WOT-MEF, 39 terms over 33 genes for iPSC-CM, and 14 terms over 45 genes for SC-islet. The Hill exponent is fixed to $h = 2.0$ for WOT-MEF and SC-islet and $h = 2.8$ for iPSC-CM.

For each activating interaction $j \to g$, the half-saturation constant is determined from the training expression scale of the corresponding upstream regulator(s). Repressive interactions enter multiplicatively through the repression gates defined in Appendix E.

For WOT-MEF, the regulatory prior additionally includes the exogenous OSKM input during the Dox window and a state-dependent endogenous pluripotency component after Dox withdrawal. The SC-islet network includes the Notch–*HES1*–*SOX9*–*NEUROG3* regulatory module and downstream endocrine regulators. Dataset-specific protocol inputs affect only the explicitly designated regulatory or developmental transitions above.

C Conditional Generator Implementation

This appendix provides implementation details for the two conditional models introduced in Section 4.2: the joint occurrence model $p_\theta(\boldsymbol{b} \mid \boldsymbol{z})$ and the positive-magnitude model $p_\psi(\boldsymbol{a} \mid \boldsymbol{b}, \boldsymbol{z})$. The generator models $G_r = 200$ downstream genes disjoint from the mechanistic core-gene set. Both models are conditioned on the structured state $\boldsymbol{z} = (\boldsymbol{x}, s)$ and receive no explicit biological-time or day variable; temporal information therefore enters the generator only through the mechanistically extrapolated condition.

C.1 Joint Occurrence Diffusion

Forward process. The occurrence model operates jointly on the complete binary vector $\boldsymbol{b} \in \{0,1\}^{G_r}$. We use the binary corruption process in equation 10 with $K = 50$ diffusion steps and a cosine schedule

$$\bar{\alpha}_k = \frac{f(k)}{f(0)}, \qquad f(k) = \cos^2\left[\frac{k/K + \varsigma}{1+\varsigma}\frac{\pi}{2}\right], \qquad \varsigma = 0.008, \qquad k = 0, \ldots, K. \tag{29}$$

Thus, $\bar{\alpha}_0 = 1$ and $\bar{\alpha}_K \approx 0$. For each gene g, the corresponding corrupted-state marginal is

$$p\left(b_g^{(k)} = 1 \mid b_g^{(0)}\right) = \bar{\alpha}_k b_g^{(0)} + \frac{1-\bar{\alpha}_k}{2}. \tag{30}$$

Conditional denoiser. The denoiser receives the corrupted occurrence vector, diffusion-step embedding, developmental-state embedding, and core-gene expression:

$$\boldsymbol{\ell}_\theta^{(k)} = \mathrm{MLP}_\theta\left([\boldsymbol{b}^{(k)}, \mathrm{emb}_{\sin}(k), \mathrm{emb}_{\mathrm{state}}(s), \boldsymbol{x}]\right), \qquad \widehat{\boldsymbol{p}}_0^{(k)} = \sigma\left(\boldsymbol{\ell}_\theta^{(k)}\right). \tag{31}$$

Here $\widehat{\boldsymbol{p}}_0^{(k)} \in [0,1]^{G_r}$ denotes the predicted probabilities of the clean occurrence mask $\boldsymbol{b}^{(0)}$. The MLP has three hidden layers of width 256 with GELU activations, a 32-dimensional diffusion-step embedding, and a 16-dimensional developmental-state embedding. Its G_r output logits jointly predict the clean occurrence pattern. Unlike the D3 refinement in Appendix E.2, this model is not residual: the downstream genes have no mechanistic occurrence mask on which to anchor the prediction.

During training, k is sampled uniformly from $\{1, \ldots, K\}$ and the clean mask $\boldsymbol{b}^{(0)}$ is corrupted according to equation 10. The model is optimized using the binary cross-entropy objective in equation 11. We use Adam with an initial learning rate of 2×10^{-3}, cosine annealing, batch size 4096, and 6×10^4 optimization steps.

Reverse sampling. Generation starts from the terminal binary reference distribution

$$\boldsymbol{b}^{(K)} \sim \mathrm{Bernoulli}(1/2)^{\otimes G_r}. \tag{32}$$

For $k = K, \ldots, 1$, the denoiser predicts the clean-state probabilities $\widehat{\boldsymbol{p}}_0^{(k)}$. For each coordinate g and $c \in \{0,1\}$, define

$$m_c^{(k-1)} = \bar{\alpha}_{k-1} c + \frac{1-\bar{\alpha}_{k-1}}{2}, \tag{33}$$

and compute the exact forward-process posterior

$$q_{g,c}^{(k-1)} = \frac{q_k(b_g^{(k)} \mid 1)\, m_c^{(k-1)}}{q_k(b_g^{(k)} \mid 1)\, m_c^{(k-1)} + q_k(b_g^{(k)} \mid 0)\,[1 - m_c^{(k-1)}]}, \tag{34}$$

where

$$q_k(y \mid r) = \alpha_k \mathbf{1}[y = r] + \frac{1-\alpha_k}{2}. \tag{35}$$

The reverse probability is

$$\pi_g^{(k-1)} = \widehat{p}_{0,g}^{(k)} q_{g,1}^{(k-1)} + \left(1 - \widehat{p}_{0,g}^{(k)}\right) q_{g,0}^{(k-1)}, \tag{36}$$

from which

$$b_g^{(k-1)} \sim \mathrm{Bernoulli}\left(\pi_g^{(k-1)}\right). \tag{37}$$

The final sample $\boldsymbol{b} = \boldsymbol{b}^{(0)}$ is supplied to the positive-magnitude model. In contrast to D3, which partially corrupts and refines an existing mechanistic occurrence mask, the downstream occurrence model generates the complete mask from binary noise.

C.2 Positive-Only Normalization

Positive-expression magnitudes are normalized separately from occurrence. For each downstream gene g, let $\mu_{+,g}$ and $\sigma_{+,g}$ denote the mean and standard deviation of $\log(1 + a_g)$ computed using positive training entries only. For an active coordinate ($b_g = 1$), the normalized magnitude is

$$\widetilde{a}_g = \frac{\log(1 + a_g) - \mu_{+,g}}{\sigma_{+,g}}. \tag{38}$$

Zero-expression entries are excluded from both the normalization statistics and the magnitude loss. For inactive coordinates ($b_g = 0$), for which no positive magnitude is defined, we set the flow target to the placeholder value $\tilde{a}_{1,g} = 0$.

The inverse transformation for active coordinates is

$$a_g = \max\left\{0,\, \exp(\sigma_{+,g}\tilde{a}_g + \mu_{+,g}) - 1\right\}, \qquad b_g = 1. \tag{39}$$

The nonnegativity projection prevents numerical flow trajectories outside the positive-expression support from producing negative expression values. The complete downstream expression vector is reconstructed as $r = b \odot a$, so inactive coordinates are deterministically zero.

C.3 Conditional Positive-Magnitude Flow

Velocity-field architecture. Conditioned on the sampled occurrence mask $\boldsymbol{b}$ and structured state $\boldsymbol{z} = (\boldsymbol{x}, s)$, the vector field receives the current normalized magnitude, flow-time embedding, occurrence mask, developmental-state embedding, and core-gene expression:

$$\boldsymbol{v}_\psi\left(\widetilde{\boldsymbol{a}}_\tau, \tau; \boldsymbol{b}, \boldsymbol{z}\right) = \mathrm{MLP}_\psi\left([\widetilde{\boldsymbol{a}}_\tau, \boldsymbol{b}, \mathrm{emb}_{\sin}(\tau), \mathrm{emb}_{\mathrm{state}}(s), \boldsymbol{x}]\right). \tag{40}$$

The network has three hidden layers of width 512 with SiLU activations, a 32-dimensional flow-time embedding, and a 16-dimensional developmental-state embedding. Providing the complete occurrence mask $\boldsymbol{b}$ allows the vector field to model dependencies between positive-expression magnitudes and the joint occurrence pattern in addition to the structured mechanistic condition.

Training. Let $\widetilde{\boldsymbol{a}}_1$ denote the normalized positive-expression vector obtained from the training data using equation 38. We sample $\widetilde{\boldsymbol{a}}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and $\tau \sim \mathrm{Unif}[0, 1]$, and construct the linear probability path

$$\widetilde{\boldsymbol{a}}_\tau = (1 - \tau)\widetilde{\boldsymbol{a}}_0 + \tau\widetilde{\boldsymbol{a}}_1. \tag{41}$$

The Gaussian source is sampled over all coordinates without masking. Consequently, for an inactive coordinate ($b_g = 0$), whose target is $\tilde{a}_{1,g} = 0$, the interpolation path reduces to

$$\tilde{a}_{\tau,g} = (1 - \tau)\tilde{a}_{0,g}, \qquad b_g = 0. \tag{42}$$

Thus, inactive coordinates contain linearly decaying Gaussian noise rather than being fixed at zero along the flow path. The complete occurrence mask b is provided explicitly to the vector field, allowing it to distinguish active from inactive coordinates. The vector field is trained with the masked flow-matching objective in equation 13, whose target velocity is $\widetilde{\boldsymbol{a}}_1 - \widetilde{\boldsymbol{a}}_0$. Only coordinates active in $\boldsymbol{b}$ contribute to the loss. We use Adam with an initial learning rate of 2×10^{-4}, cosine annealing, batch size 2048, and 3×10^4 optimization steps.

Sampling and inverse transformation. At inference, we initialize $\widetilde{\boldsymbol{a}}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and integrate the flow ODE in equation 14 from $\tau = 0$ to 1 using 50 uniform Euler steps:

$$\widetilde{\boldsymbol{a}}_{\tau+\Delta\tau} = \widetilde{\boldsymbol{a}}_\tau + \Delta\tau\, \boldsymbol{v}_\psi\left(\widetilde{\boldsymbol{a}}_\tau, \tau; \boldsymbol{b}, \boldsymbol{z}\right), \qquad \Delta\tau = \frac{1}{50}. \tag{43}$$

The resulting endpoint $\tilde{a}_1$ is transformed gene-wise to the original expression scale using equation 39, yielding a. All coordinates are integrated during sampling; inactive coordinates are neither clamped nor re-noised during the flow and are discarded only in the final reconstruction $r = b \odot a$.

C.4 SEQUENTIAL TRAINING

The occurrence diffusion is trained first,

$$\theta^\star = \arg\min_\theta \mathcal{L}_{\mathrm{occ}}(\theta), \tag{44}$$

and is subsequently frozen while the positive-magnitude flow is optimized. Thus, gradients from the flow-matching objective do not update the occurrence model. When $\lambda_{\mathrm{OT}} > 0$, the optional transport objective is applied only during magnitude-flow training after the occurrence model has been frozen. Its implementation is described in Appendix D.

D TRANSPORT-REGULARIZATION IMPLEMENTATION

The headline MeCoGen configuration uses $\lambda_{\mathrm{OT}} = 0$. Transport-regularized MeCoGen-OT variants are evaluated only to study how explicit population-level transport alignment interacts with stochastic distributional fidelity.

For an OT-regularized update, occurrence masks are sampled from the frozen occurrence model. The magnitude flow then generates downstream expression, and the resulting population representation is formed by concatenating the structured core and downstream blocks,

$$\widehat{\boldsymbol{y}} = [\boldsymbol{x}, \widehat{\boldsymbol{r}}]. \tag{45}$$

For generated and observed training mini-populations,

$$\widehat{\mu} = \frac{1}{N}\sum_{i=1}^{N} \delta_{\widehat{\boldsymbol{y}}_i}, \qquad \mu = \frac{1}{M}\sum_{j=1}^{M} \delta_{\boldsymbol{y}_j}, \tag{46}$$

we compute the $p = 2$ entropically regularized Sinkhorn discrepancy used in equation 17.

The implementation uses Sinkhorn blur 0.05 and scaling 0.5. To reduce the cost of differentiating through population generation, the OT-training branch integrates the magnitude flow using 10 differentiable Euler steps. Gradients are propagated through this integration to the magnitude network while the occurrence network remains frozen. Standard generation and evaluation use the 50-step integration described in Appendix C.3.

When

$$\lambda_{\mathrm{OT}} = 0, \tag{47}$$

the transport branch is skipped entirely. For the MeCoGen-OT sensitivity analysis, we evaluate

$$\lambda_{\mathrm{OT}} \in \left\{0,\, 10^{-3},\, 10^{-2},\, 10^{-1}\right\}. \tag{48}$$

No held-out future population is used in the transport objective; both the generated and reference populations used for this regularizer correspond only to the training window.

E POST-MECHANISTIC CALIBRATION

The mechanistic simulator (M0) produces a future population of core-gene expression vectors and developmental states while accounting for differential population expansion and depletion. Before this population is used to condition the transcriptome generator, MeCoGen applies two training-calibrated refinements. **M3** calibrates gene-wise occurrence and expression marginals, whereas **D3** refines multigene occurrence dependence while preserving the state-conditioned occurrence targets established by M3. The complete interface is

$$\mathrm{M0} \rightarrow \text{growth-adjusted resampling} \rightarrow \mathrm{M3} \rightarrow \mathrm{D3} \rightarrow \boldsymbol{z} = (\boldsymbol{x}^{\mathrm{D3}}, s). \tag{49}$$

For every forecasting split, M0, M3, and D3 use only cells within the corresponding training window; no held-out future population is used for mechanistic fitting or calibration. Throughout this section, $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{iG_c}) \in \mathbb{R}_+^{G_c}$ denotes the simulated core-gene expression vector for cell i after growth-adjusted resampling, $s_i \in \{1, \ldots, K_s\}$ its mechanistically predicted developmental state, and $b_{ig} = \mathbb{I}[x_{ig} > 0]$ the occurrence indicator for core gene g. Real training-cell quantities are distinguished by the superscript real.

E.1 M3: MARGINAL EXPRESSION CALIBRATION

M3 corrects residual discrepancies in gene occurrence and expression magnitude without modifying the developmental states predicted by M0. The two corrections use different levels of conditioning: occurrence rates and newly activated expression values are calibrated within developmental state, whereas the quantile correction for existing positive magnitudes is estimated gene-wise after pooling across states.

State-conditioned occurrence calibration. Let $\mathcal{I}_s^{\mathrm{tr}} = \{i : s_i^{\mathrm{real}} = s\}$ denote the set of labeled training cells in developmental state s. For each state s and core gene g, the target occurrence probability is

$$\pi_{sg} = \frac{1}{|\mathcal{I}_s^{\mathrm{tr}}|} \sum_{i \in \mathcal{I}_s^{\mathrm{tr}}} \mathbb{I}[x_{ig}^{\mathrm{real}} > 0]. \tag{50}$$

We also retain the state- and gene-specific empirical positive-expression pool

$$\mathcal{P}_{sg} = \left\{ x_{ig}^{\mathrm{real}} : i \in \mathcal{I}_s^{\mathrm{tr}},\ x_{ig}^{\mathrm{real}} > 0 \right\}, \tag{51}$$

which is used when calibration activates an expression value that was zero in the mechanistic prediction.

Gene-wise magnitude calibration. Magnitude calibration is performed separately for each core gene using positive-expression values only, while pooling cells across developmental states. For each observed training day d, let $Q_{g,d}^{\mathrm{real},+}(u)$ and $Q_{g,d}^{\mathrm{mech},+}(u)$ denote the empirical quantile functions of the positive real and mechanistically simulated expression values for gene g, respectively. At quantile levels $u_m \in \{0.02, \ldots, 0.98\}$ on a 49-point uniform grid, we compute

$$\delta_g^{(d)}(u_m) = Q_{g,d}^{\mathrm{real},+}(u_m) - Q_{g,d}^{\mathrm{mech},+}(u_m). \tag{52}$$

Mechanistic and real populations are compared only within the same experimental arm.

The day-specific corrections are aggregated as

$$\delta_g(u) = \sum_{d \in \mathcal{D}_{\mathrm{tr}}} \widetilde{\omega}_d \, \delta_g^{(d)}(u), \qquad \widetilde{\omega}_d = \frac{\sqrt{n_d}}{\sum_{d' \in \mathcal{D}_{\mathrm{tr}}} \sqrt{n_{d'}}}, \tag{53}$$

where $\mathcal{D}_{\mathrm{tr}}$ is the set of observed training days and n_d is the number of training cells used at day d. The pooled correction is linearly interpolated between grid points and held constant outside the grid; genes with insufficient positive observations use $\delta_g \equiv 0$. The square-root weighting gives greater weight to better-sampled days while preventing differences in cell count from dominating the pooled correction.

The correction is applied only to mechanistically predicted positive entries. For a simulated cell with $x_{ig} > 0$, let R_{ig}^{+} denote the rank of x_{ig} among the positive simulated values for gene g in the corresponding population, and let N_g^{+} denote the number of such positive values. We define

$$u_{ig}^{+} = \frac{R_{ig}^{+} - \frac{1}{2}}{N_g^{+}}, \qquad \widetilde{x}_{ig} = \max\left\{0,\, x_{ig} + \delta_g(u_{ig}^{+})\right\}, \qquad x_{ig} > 0. \tag{54}$$

Thus, occurrence and positive-expression magnitude are calibrated separately: π_{sg} controls the state-conditioned probability of expression, whereas δ_g aligns the conditional positive-expression distribution.

Combining occurrence and magnitude calibration. For each simulated cell and gene, M3 first draws a calibrated occurrence indicator

$$b_{ig}^{\mathrm{M3}} \sim \mathrm{Bernoulli}(\pi_{s_i g}). \tag{55}$$

The calibrated expression is then

$$x_{ig}^{\mathrm{M3}} = \begin{cases} \widetilde{x}_{ig}, & & b_{ig} = 1,\ b_{ig}^{\mathrm{M3}} = 1, \\ \zeta_{ig}, & \zeta_{ig} \sim \mathrm{Unif}(\mathcal{P}_{s_i g}), & b_{ig} = 0,\ b_{ig}^{\mathrm{M3}} = 1, \\ 0, & & b_{ig}^{\mathrm{M3}} = 0. \end{cases} \tag{56}$$

Here $\mathrm{Unif}(\mathcal{P}_{sg})$ denotes uniform sampling from the finite empirical pool $\mathcal{P}_{sg}$. Thus, positive entries that remain active retain their corrected mechanistic magnitudes, newly activated entries receive values from the corresponding state- and gene-specific training pool, and inactive entries are set to zero.

M3 therefore combines state-conditioned occurrence calibration with a gene-wise magnitude correction pooled across developmental states. Because the occurrence indicators are sampled independently across genes, however, M3 does not by itself recover multigene occurrence dependence.

E.2 D3: Marginal-Preserving Dependence Refinement

D3 addresses this remaining dependence mismatch. Although M3 calibrates each state–gene occurrence probability π_{sg}, independent Bernoulli sampling does not preserve the joint occurrence distribution across core genes. D3 therefore learns multigene occurrence structure while retaining the calibrated state-conditioned occurrence targets $\{\pi_{sg}\}$.

Residual discrete diffusion. Let $\boldsymbol{b}_i^{(0)} = (b_{i1}^{(0)}, \ldots, b_{iG_c}^{(0)}) \in \{0,1\}^{G_c}$ denote the clean binary occurrence vector of a labeled training cell. We train a state-conditioned discrete diffusion model on these vectors. At diffusion step $k \in \{1, \ldots, K\}$, each coordinate is retained with cumulative probability $\bar{\alpha}_k$ and otherwise replaced by an unbiased Bernoulli draw,

$$b_{ig}^{(k)} = \begin{cases} b_{ig}^{(0)}, & \text{with probability } \bar{\alpha}_k, \\ \mathrm{Bernoulli}(1/2), & \text{with probability } 1 - \bar{\alpha}_k, \end{cases} \tag{57}$$

where $\{\bar{\alpha}_k\}_{k=1}^{K}$ follows the cosine diffusion schedule.

Rather than predicting clean occurrence probabilities from scratch, the denoiser learns a residual correction to the M3 target logits:

$$\widehat{p}_{ig}^{(0)} = \sigma\left(\mathrm{logit}\left[\mathrm{clip}(\pi_{s_i g}, \epsilon_\pi, 1 - \epsilon_\pi)\right] + \rho_{\vartheta, g}(\boldsymbol{b}_i^{(k)}, k, s_i)\right), \tag{58}$$

where ρ_ϑ is an MLP conditioned on diffusion step and developmental state, $\sigma(\cdot)$ is the logistic sigmoid, and ϵ_π is a small numerical constant that prevents infinite logits when $\pi_{s_i g}$ is exactly zero or one. The output layer of ρ_ϑ is initialized to zero, so the initial denoiser reproduces the M3 target occurrence probabilities and subsequently learns residual multigene structure. Training minimizes

$$\mathcal{L}_{\mathrm{D3}} = \mathbb{E}\left[\mathrm{BCE}\left(\widehat{\boldsymbol{p}}_i^{(0)}, \boldsymbol{b}_i^{(0)}\right)\right]. \tag{59}$$

Partial-noise refinement. At prediction time, D3 refines the M3 occurrence pattern rather than generating a new mask independently. Let $\boldsymbol{b}_i^{\mathrm{M3}}$ denote the occurrence vector obtained from M3. It is corrupted to an intermediate diffusion step $k_0 = \lceil \eta K \rceil$ with $\eta = 0.3$, and the learned reverse process is then applied from k_0 to zero:

$$\boldsymbol{b}_i^{\mathrm{M3}} \longrightarrow \boldsymbol{b}_i^{(k_0)} \longrightarrow \widehat{\boldsymbol{p}}_i^{(0)}. \tag{60}$$

This partial-noise construction keeps the refinement anchored to the M3 occurrence pattern while allowing the denoiser to reorganize multigene co-occurrence structure.

Marginal-preserving decoding. Sampling each coordinate independently from $\widehat{\boldsymbol{p}}_i^{(0)}$ would generally perturb the calibrated occurrence targets $\{\pi_{sg}\}$. Instead, for every state–gene pair (s, g), the denoiser probabilities are used to rank the N_s simulated cells currently assigned to state s. We set

$$k_{sg} = \mathrm{round}(N_s \pi_{sg}) \tag{61}$$

and activate gene g in the k_{sg} highest-ranked cells,

$$b_{ig}^{\mathrm{D3}} = \mathbb{I}\left[i \in \mathrm{TopK}_{k_{sg}} \left\{ \widehat{p}_{jg}^{(0)} : s_j = s \right\} \right]. \tag{62}$$

D3 therefore enforces the calibrated occurrence count implied by π_{sg} for each state–gene pair while changing which cells are active. This preserves the calibrated marginal targets while allowing cross-gene co-occurrence patterns to change. Ties at the Top-k selection boundary, if any, are broken uniformly at random.

Finally, expression magnitudes are reattached using the same rule as in M3: genes that remain active retain their M3 magnitudes, newly activated genes are sampled from $\mathcal{P}_{s_i g}$, and inactive genes are set to zero. Denoting the resulting core-expression vector by $\boldsymbol{x}_i^{\mathrm{D3}}$, the final structured condition is

$$\boldsymbol{z}_i = (\boldsymbol{x}_i^{\mathrm{D3}}, s_i). \tag{63}$$

Neither M3 nor D3 modifies the developmental state s_i predicted by the mechanistic transition model.

E.3 Training-Split Isolation and Forecasting Interface

For each forecasting split, M0 is re-estimated using only the observed training days for that split. The growth-adjusted simulated population is resampled before M3 and D3 are applied, and both calibration stages are fitted exclusively from labeled training-day cells. The complete forecasting branch is therefore

$$\mathrm{M0} \rightarrow \text{growth-adjusted resampling} \rightarrow \mathrm{M3} \rightarrow \mathrm{D3} \rightarrow (\boldsymbol{x}^{\mathrm{D3}}, s), \tag{64}$$

with no access to the held-out forecast population during mechanistic fitting, calibration, or simulation.

F TRAINING-ONLY SELECTION OF THE TRANSPORT REGULARIZATION

MeCoGen-OT augments the conditional generator with a Sinkhorn transport regularizer,

$$\mathcal{L}_{\text{MeCoGen-OT}} = \mathcal{L}_{\text{FM}} + \lambda_{\text{OT}} \mathcal{L}_{\text{OT}}, \tag{65}$$

where λ_{OT} controls the strength of population-level transport regularization. We consider

$$\lambda_{\text{OT}} \in \{0, 10^{-3}, 10^{-2}, 10^{-1}\}.$$

All values reported for MeCoGen-OT in Table 1 use λ_{OT} selected exclusively from data within the corresponding training window; the outer forecast targets are not accessed during model selection.

Selection criterion. For each inner temporal validation task v and candidate λ, we evaluate four complementary lower-is-better criteria: MMD, Energy distance, mean KS, and

$$\Delta W_{2,v}^{\text{full}}(\lambda) = \left| W_{2,v}^{\text{full}}(\lambda) - W_{2,v}^{\text{Real-Split}} \right|,$$

where the Real–Split reference is computed from two disjoint subsets of the real validation-day population using the same full-gene transport pipeline.

Because these metrics have different numerical scales, each criterion $E_{j,v}$ is min–max normalized across the candidate values within the same validation task:

$$\widetilde{E}_{j,v}(\lambda) = \frac{E_{j,v}(\lambda) - \min_{\lambda'} E_{j,v}(\lambda')}{\max_{\lambda'} E_{j,v}(\lambda') - \min_{\lambda'} E_{j,v}(\lambda') + \epsilon}. \tag{66}$$

The validation score is then

$$S_v(\lambda) = \frac{1}{4} \left(\widetilde{\text{MMD}}_v + \widetilde{\text{Energy}}_v + \widetilde{\text{KS}}_v + \widetilde{\Delta W_2^{\text{full}}}_v \right). \tag{67}$$

When multiple inner validation tasks are available, $S_v(\lambda)$ is averaged over tasks and the candidate with the smallest mean score is selected. Thus, OT strength is selected by a balanced distributional criterion rather than by minimizing raw W_2 alone.

G Implementation Details

G.1 Data, Preprocessing, and Temporal Splits

WOT-MEF. We use the Waddington-OT MEF-to-iPSC reprogramming time course (Schiebinger et al., 2019) and restrict all experiments to the Dox+2i arm; the serum arm is excluded so that each split represents a single protocol trajectory. After restricting to cells with state annotations where required, the dataset contains $83{,}407$ cells.

We use the provided $\log(1 + x)$ normalized expression matrix rather than raw UMI counts. Consequently, dispersion-based burst calibration from raw counts is not available for WOT-MEF; the reported mechanistic model instead uses the shared burst prior described in Section 4.1. The transcriptome is partitioned into two disjoint blocks: $G_c = 41$ core regulatory genes used by the mechanistic model and $G_r = 200$ downstream target genes selected from the dataset's highly variable genes.

Blind temporal evaluation. All main evaluations are strict temporal extrapolations: models are fitted only to observations at or before a cutoff and evaluated on a later unseen population. We use four rolling splits:

Split	Training window	Target	State-label coverage
A	$\leq$ D8	D9	51.7%
B	$\leq$ D9.5	D10.5	86.0%
C	$\leq$ D12	D15	99.6%
D	$\leq$ D16	D18	99.7%

The lower state-label coverage at D9 reflects the annotations available in the original dataset and affects only state-dependent analyses.

Every fitted component—including M0 parameters, M3/D3 calibration, conditional generators, normalization statistics, and PCA/scaler bases used for evaluation—is estimated independently from the corresponding split's training window. No held-out target population enters model fitting or calibration. For horizon analyses, a single model fit is evaluated at multiple subsequent times so that forecast distance varies while the training data remain fixed.

G.2 External Baselines

We evaluate the released implementations of scNODE (Zhang et al., 2024), scIMF (Jiang et al., 2026), and LGP-OT (Balik & Lähdesmäki, 2026), using only data-loading adapters required to apply the same temporal splits and $G_c + G_r = 241$-gene evaluation panel. Each baseline is trained once per split and evaluated at the corresponding future targets.

- **scNODE**: latent dimension 50, drift network $[50, 50]$, 10 epochs $\times$ 100 iterations, batch size 32, learning rate 10^{-3}, 200 pretraining iterations, and 2000 simulated cells.
- **scIMF**: neural SDE in a 50-dimensional PCA space, with the PCA basis fitted from training cells only; 3000 epochs and 3000 generated cells.
- **LGP-OT**: latent dimension 32, decoder width 50, $M = 10$ basis functions, 1250 epochs, batch size 256, learning rate 10^{-3}, and 2000 simulated cells.

These settings follow the authors' reference configurations rather than hyperparameters tuned on the held-out targets. The headline MeCoGen model uses $\lambda_{\mathrm{OT}} = 0$ and therefore requires no transport-regularization hyperparameter selection; the training-only validation procedure in Appendix F is used only for the MeCoGen-OT variant. For the controlled conditioning experiment, scNODE is additionally trained only on the G_c core genes and paired with a gradient-boosted classifier $p(s \mid \boldsymbol{x})$ fitted on training-day cells. This produces the data-driven structured condition while leaving the downstream MeCoGen generator unchanged.

G.3 Evaluation Metrics

We evaluate complementary aspects of marginal, sparsity, joint-distribution, and transport fidelity. Let $\widehat{\mathcal{Y}} = \{\widehat{\boldsymbol{y}}_i\}_{i=1}^n$ and $\mathcal{Y} = \{\boldsymbol{y}_j\}_{j=1}^m$ denote generated and real target populations, respectively, over G evaluated genes.

Marginal and sparsity metrics. For each gene g, let $\widehat{F}_g$ and F_g denote the empirical cumulative distribution functions of generated and real expression. We report the mean two-sample Kolmogorov–Smirnov statistic

$$\mathrm{KS} = \frac{1}{G}\sum_{g=1}^{G} \sup_x \left|\widehat{F}_g(x) - F_g(x)\right|. \tag{68}$$

To assess preservation of expression sparsity, we define the per-gene zero fraction as

$$z_g(\mathcal{Y}) = \frac{1}{m}\sum_{j=1}^{m} \mathbf{1}[y_{j,g} = 0],$$

and report the mean absolute zero-fraction error

$$E_{\mathrm{zero}} = \frac{1}{G}\sum_{g=1}^{G} \left|z_g(\widehat{\mathcal{Y}}) - z_g(\mathcal{Y})\right|. \tag{69}$$

Positive-only variants are computed analogously after restricting each gene to its expressed entries.

Maximum mean discrepancy. Joint distribution fidelity is measured using the squared maximum mean discrepancy

$$\begin{aligned}\mathrm{MMD}^2 = &\frac{1}{n(n-1)}\sum_{i\neq i'} k(\widehat{\boldsymbol{y}}_i, \widehat{\boldsymbol{y}}_{i'}) + \frac{1}{m(m-1)}\sum_{j\neq j'} k(\boldsymbol{y}_j, \boldsymbol{y}_{j'}) \\ &- \frac{2}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m} k(\widehat{\boldsymbol{y}}_i, \boldsymbol{y}_j),\end{aligned} \tag{70}$$

using the Gaussian kernel

$$k(\boldsymbol{y}, \boldsymbol{y}') = \exp\left(-\frac{\|\boldsymbol{y} - \boldsymbol{y}'\|_2^2}{2\sigma^2}\right). \tag{71}$$

The bandwidth σ is set to $1.0\times$ the median pairwise-distance heuristic computed from the real target population, and the same convention is used across the three datasets in the main benchmark. Additional bandwidth sweeps assess sensitivity to local versus global distributional structure.

Energy distance. We additionally report the empirical Energy distance

$$\begin{aligned}\mathrm{Energy} = &\frac{2}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m} \|\widehat{\boldsymbol{y}}_i - \boldsymbol{y}_j\|_2 \\ &- \frac{1}{n^2}\sum_{i=1}^{n}\sum_{i'=1}^{n} \|\widehat{\boldsymbol{y}}_i - \widehat{\boldsymbol{y}}_{i'}\|_2 - \frac{1}{m^2}\sum_{j=1}^{m}\sum_{j'=1}^{m} \|\boldsymbol{y}_j - \boldsymbol{y}_{j'}\|_2.\end{aligned} \tag{72}$$

Energy distance is evaluated in the pooled expression representation. Where reported, standardized variants use gene-wise statistics fitted under the corresponding evaluation protocol.

Transport metrics. Transport quantities are computed using `geomloss` (Feydy et al., 2019) in double precision with debiasing enabled. For empirical measures

$$\widehat{\mu} = \frac{1}{n}\sum_{i=1}^{n} \delta_{\widehat{\boldsymbol{y}}_i}, \qquad \mu = \frac{1}{m}\sum_{j=1}^{m} \delta_{\boldsymbol{y}_j},$$

we first define the entropically regularized transport cost

$$\mathrm{OT}_{\varepsilon,2}(\alpha,\beta) = \min_{\pi\in\Pi(\alpha,\beta)} \left\{ \int \|\boldsymbol{y}-\boldsymbol{y}'\|_2^2 \, d\pi(\boldsymbol{y},\boldsymbol{y}') + \varepsilon \,\mathrm{KL}(\pi \,\|\, \alpha\otimes\beta) \right\}, \tag{73}$$

where $\Pi(\alpha,\beta)$ denotes the set of couplings with marginals α and β. The debiased Sinkhorn divergence is then

$$\begin{aligned} S_{\varepsilon,2}(\alpha,\beta) = \; & \mathrm{OT}_{\varepsilon,2}(\alpha,\beta) - \frac{1}{2}\,\mathrm{OT}_{\varepsilon,2}(\alpha,\alpha) \\ & - \frac{1}{2}\,\mathrm{OT}_{\varepsilon,2}(\beta,\beta). \end{aligned} \tag{74}$$

We report

$$W_2^{\mathrm{Sink}} = S_{\varepsilon,2}\left(\widehat{\mu},\mu\right). \tag{75}$$

Following the evaluation convention used in Jiang et al. (2026), we report the direct debiased $p=2$ Sinkhorn quantity and denote it by W_2^{Sink}; no additional square root is applied.

Real–Split calibration. To characterize the finite-sample scale of each metric, the real target population is partitioned into two disjoint subsets and evaluated through the same scoring pipeline. We refer to this empirical reference as *Real–Split*; it is a calibration reference rather than a mathematical lower bound.

For transport geometry, we additionally report the target-wise deviation from the corresponding Real–Split reference,

$$\Delta W_2^{(m)} = \frac{1}{|\mathcal{T}|}\sum_{t\in\mathcal{T}} \left| W_{2,t}^{(m)} - W_{2,t}^{\text{Real-Split}} \right|, \tag{76}$$

where $\mathcal{T}$ denotes the set of forecast targets for the corresponding dataset. This distinction is useful because a model can attain a smaller raw transport distance through variance contraction or centralization while deviating more from the empirical geometry of independently sampled real populations.

G.4 Optimization and Compute

The occurrence diffusion and magnitude flow are trained with Adam and cosine learning-rate annealing; architecture- and model-specific settings are provided in Appendix C. Mechanistic parameters are refitted for each temporal split using only the corresponding training observations, with warm starts used where applicable.

Compute. Experiments were run on NVIDIA B200 GPUs. A complete MeCoGen fit, including mechanistic calibration and both downstream generative models, requires on the order of minutes per split; baseline runtimes range from approximately one to forty minutes depending on the method.

H SC-ISLET LEAVE-ONE-TIMEPOINT-OUT ROBUSTNESS

To test sensitivity to the temporal split, we perform leave-one-timepoint-out (LOTO) evaluation on SC-islet. For each target day D2–D7, the corresponding population is removed and each model is refit using the remaining permitted observations. D2–D6 are interpolation tasks because observations remain available on both sides of the held-out day, whereas D7 additionally tests extrapolation beyond the final training time.

Table 6 compares MeCoGen with the same three external baselines used in the main benchmark. MeCoGen achieves the lowest MMD, Energy, mean KS, and zero-fraction error among the four methods at every held-out day. The advantage therefore persists across substantially different temporal splits and remains present at D7, where prediction requires genuine out-of-time extrapolation.

For full-gene W_2, we additionally report two empirical real–real references. *Real-Split* compares two disjoint subsets of cells from the same differentiation and measures finite-sample variability, whereas *Real-Rep.* compares independent differentiation replicates at the same day and additionally captures between-replicate biological and experimental variation. MeCoGen remains close to these empirical transport scales across the LOTO experiments, whereas the three external baselines consistently obtain raw W_2 values substantially below both real–real references. Together with their larger MMD, Energy, KS, and sparsity errors, this reproduces the metric discrepancy observed in the main benchmark and shows that it is not specific to a single temporal split.

Table 6: Leave-one-timepoint-out (LOTO) robustness on SC-islet. Each column denotes the held-out target day. LOTO-D2–D6 withhold an intermediate time point while retaining observations on both sides, whereas LOTO-D7 withholds the final time point and therefore requires temporal extrapolation. MMD is reported as $\times 10^3$ using the same $1.0\times$ median-heuristic bandwidth as the main benchmark. MMD, Energy, KS, and Zero are evaluated on the full 245-gene core-plus-downstream expression space; values are mean $\pm$ SD over three test-time generation seeds from a single fitted model. W_2 denotes the same full-gene Sinkhorn metric used in Table 1; per-fold W_2 values are from a single generation seed. Real-Rep. compares independent differentiation replicates at the same day, and Real-Split compares two disjoint subsets of the same differentiation. Lower is better for MMD, Energy, KS, and Zero; raw W_2 is interpreted relative to the real-data references.

Metric	Method	D2	D3	D4	D5	D6	D7
Energy$\times 10^3$	**MeCoGen**	**0.51±0.02**	**1.05±0.03**	**0.30±0.00**	**0.28±0.02**	**0.87±0.02**	**0.63±0.02**
	scIMF	2.32±0.00	2.45±0.07	1.68±0.03	1.62±0.03	1.52±0.01	1.47±0.01
	scNODE	2.25±0.00	2.66±0.01	1.68±0.00	2.22±0.00	1.68±0.00	2.35±0.00
	LGP-OT	2.67±0.02	2.54±0.05	2.00±0.02	1.81±0.05	2.12±0.09	3.85±0.13
MMD$\times 10^3$	**MeCoGen**	**0.020±0.001**	**0.040±0.001**	**0.011±0.000**	**0.010±0.001**	**0.031±0.001**	**0.023±0.001**
	scIMF	0.058±0.000	0.070±0.002	0.034±0.001	0.033±0.001	0.032±0.000	0.030±0.000
	scNODE	0.053±0.000	0.076±0.000	0.033±0.000	0.047±0.000	0.035±0.000	0.059±0.000
	LGP-OT	0.063±0.001	0.067±0.002	0.041±0.001	0.033±0.001	0.042±0.002	0.098±0.003
mean KS	**MeCoGen**	**0.135±0.001**	**0.179±0.002**	**0.132±0.001**	**0.125±0.001**	**0.153±0.001**	**0.151±0.001**
	scIMF	0.335±0.001	0.351±0.001	0.319±0.001	0.308±0.001	0.327±0.001	0.311±0.001
	scNODE	0.321±0.000	0.338±0.000	0.313±0.000	0.309±0.001	0.320±0.000	0.337±0.000
	LGP-OT	0.291±0.002	0.288±0.002	0.263±0.001	0.234±0.003	0.271±0.004	0.317±0.005
Zero	**MeCoGen**	**0.034±0.001**	**0.052±0.001**	**0.027±0.000**	**0.024±0.000**	**0.046±0.001**	**0.042±0.001**
	scIMF	0.296±0.000	0.308±0.000	0.292±0.000	0.273±0.000	0.304±0.000	0.290±0.000
	scNODE	0.291±0.000	0.299±0.000	0.292±0.000	0.263±0.000	0.298±0.000	0.311±0.000
	LGP-OT	0.209±0.001	0.234±0.001	0.212±0.002	0.195±0.001	0.229±0.002	0.208±0.000
Full-gene W_2	*Real-Rep.*	59.01	60.02	51.68	52.52	56.81	54.81
	Real-Split	54.26	50.92	50.50	50.29	51.26	52.49
	MeCoGen	56.67	59.71	56.89	56.08	60.38	59.44
	scIMF	36.83	36.23	31.98	31.62	32.23	32.45
	scNODE	38.42	40.93	35.57	36.14	36.56	42.93
	LGP-OT	36.94	35.62	33.32	32.92	33.96	39.10

I Additional Qualitative and Diagnostic Results

The main text summarizes the principal structural results in a compact multi-panel figure. Here we provide the corresponding full diagnostic views, including both forecast horizons, the complete conditioning analysis, additional gene-level occurrence–magnitude examples, the full dependence analysis, and qualitative inspection of the mechanistic extrapolation. Unless otherwise stated, all analyses use the iPSC-CM setting trained on days 0, 1, 3, 5, and 7 and evaluated on held-out days 11 and 15.

I.1 WOT-MEF Results Across Rolling Forecasts

Table 7 disaggregates the WOT-MEF results across the four rolling out-of-time forecasts used in Table 1. MeCoGen consistently improves MMD, Energy distance, mean KS, and zero-fraction error across all four forecast targets, showing that the aggregate results are not driven by a single temporal split.

Table 7: WOT-MEF forecasting performance across individual rolling temporal splits. Each column corresponds to a separately fitted model evaluated at its held-out future target. Lower is better for all metrics.

Metric	Method	≤D8→D9	≤D9.5→D10.5	≤D12→D15	≤D16→D18
MMD $\times 10^3$	MeCoGen	**0.074**	**0.068**	**0.090**	**0.064**
	scIMF	0.185	0.139	0.345	0.156
	scNODE	0.393	0.148	0.267	0.220
	LGP-OT	0.393	0.129	0.259	0.190
Energy	MeCoGen	**1.557**	**1.130**	**1.623**	**1.012**
	scIMF	5.321	3.477	7.010	3.440
	scNODE	8.851	3.831	6.053	5.463
	LGP-OT	6.456	3.700	4.996	4.192
KS	MeCoGen	**0.117**	**0.119**	**0.197**	**0.123**
	scIMF	0.662	0.576	0.643	0.604
	scNODE	0.519	0.520	0.590	0.491
	LGP-OT	0.350	0.380	0.503	0.385
Zero	MeCoGen	**0.094**	**0.095**	**0.122**	**0.088**
	scIMF	0.665	0.629	0.685	0.589
	scNODE	0.380	0.495	0.539	0.396
	LGP-OT	0.241	0.347	0.424	0.257

I.2 Full Conditioning-Bottleneck Diagnostics

The controlled conditioning experiment in the main text summarizes the effect of replacing the future structured condition while holding the downstream generator fixed. Figure 3 provides the complete set of condition-level and transcriptome-level diagnostics for the oracle, mechanistic, and data-driven conditioning arms.

At both horizons, the mechanistic condition remains substantially closer to the oracle than the data-driven condition at the structured-state level. For example, state-TV at day 11 is 0.052 for the mechanistic condition, compared with 0.481 for the data-driven condition, while at day 15 the corresponding values are 0.087 and 0.507. A similar pattern is observed for the core-gene distribution, where the mechanistic MMD increases only from 0.93×10^{-3} to 1.34×10^{-3}, whereas the data-driven condition increases from 3.21×10^{-3} to 5.75×10^{-3}.

Because the same frozen decoder is applied to all three conditioning arms, these differences propagate directly to the final transcriptome forecast. The mechanistic full modeled gene-panel W_2 changes only from 46.07 to 47.86, whereas the data-driven value increases from 57.44 to 95.31. Energy shows the same long-horizon deterioration for the data-driven arm. These results support the interpretation that extrapolation quality is limited not only by the expressiveness of the downstream generator, but by the quality of the future condition supplied to it.

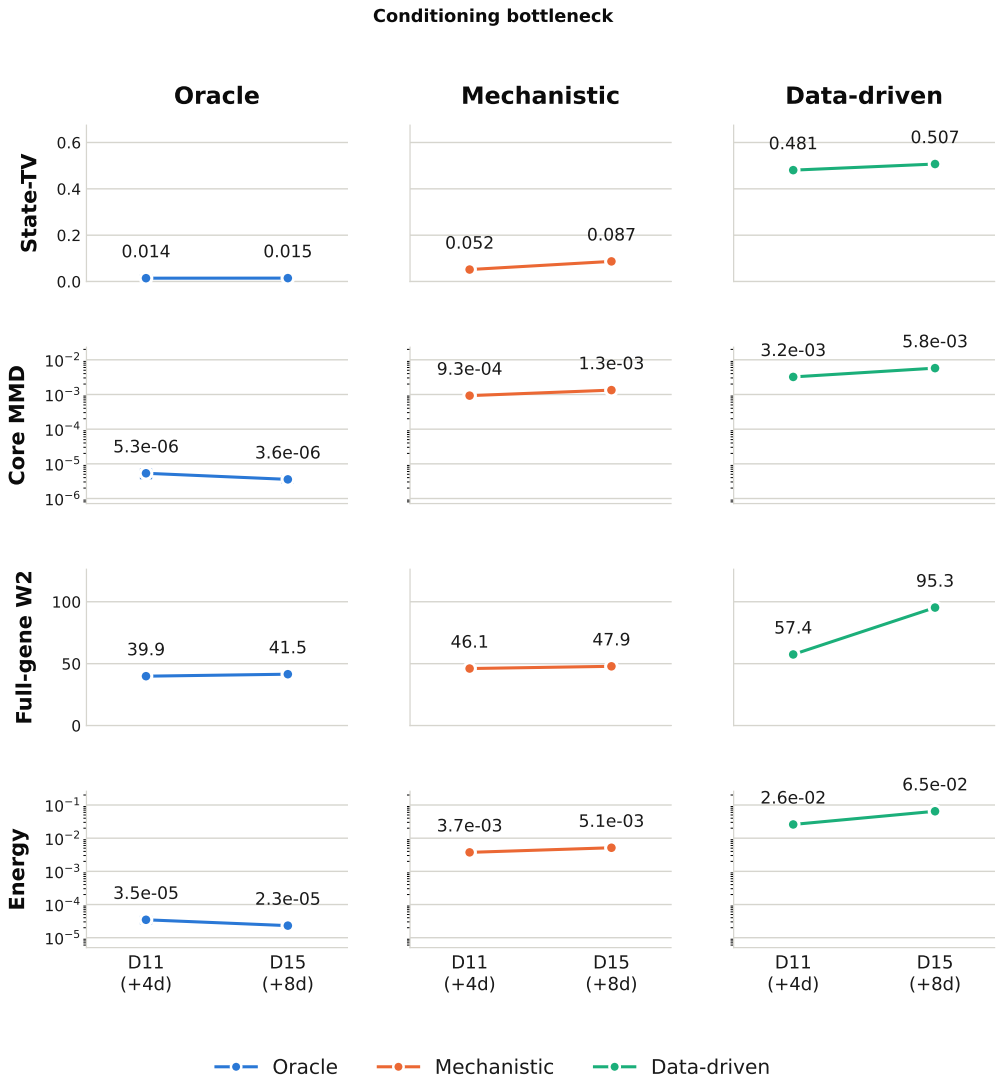

Figure 3: **Absolute diagnostics for the controlled conditioning experiment.** Oracle, mechanistic, and data-driven future conditions are passed through the same frozen downstream generator at day 11 (+4d) and day 15 (+8d). Panels report state-distribution error, core-gene distribution discrepancy, and downstream transcriptome discrepancy. The mechanistic condition remains substantially closer to the oracle across both horizons, whereas the data-driven condition accumulates larger error that propagates to the final transcriptome forecast.

I.3 Expanded Occurrence–Magnitude Dynamics

Figure 4 shows five representative genes to illustrate the distinct temporal behavior of expression occurrence and positive-expression magnitude.

Across these genes, occurrence probability $P(R_g > 0)$ and conditional positive magnitude $\mathbb{E}[R_g \mid R_g > 0]$ evolve on different temporal scales and need not change in the same direction. This supports the hurdle decomposition $R = B \odot A$ as a structural modeling choice.

Occurrence probability and conditional positive magnitude exhibit distinct temporal dynamics across genes, supporting their separate treatment in the generator. MeCoGen more closely tracks the observed occurrence and positive-magnitude trajectories than scIMF across the representative

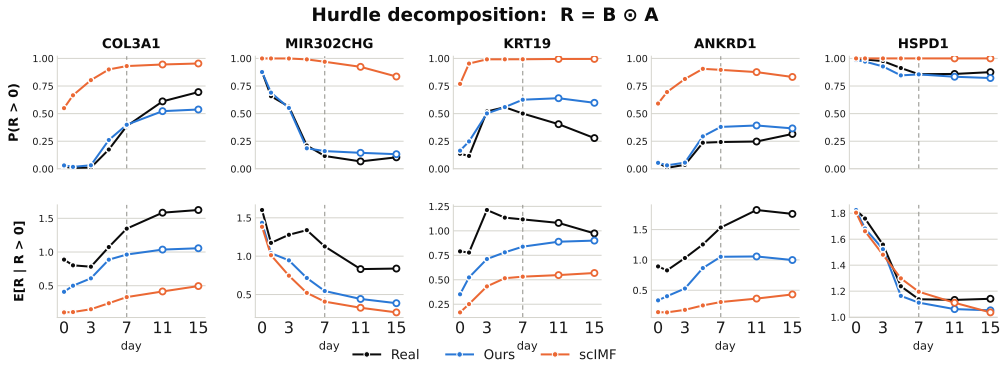

Figure 4: **Gene-level decomposition of occurrence and positive-expression magnitude.** For five representative target genes, the upper row shows $P(R_g > 0)$ and the lower row shows $\mathbb{E}[R_g \mid R_g > 0]$ across time. Filled markers denote observed training days and open markers denote held-out future targets. Occurrence and positive magnitude exhibit distinct temporal behavior, motivating their separate treatment in the hurdle generator. MeCoGen generally tracks occurrence structure more closely than scIMF, although positive-expression magnitudes can remain systematically underestimated for individual genes.

genes, with particularly large improvements in occurrence dynamics where scIMF frequently predicts near-complete expression. These results illustrate that explicitly modeling both components improves recovery of gene-level temporal expression structure.

I.4 Full Sparse Multigene Dependence Analysis

Gene selection for dependence analysis. As a post-hoc diagnostic of dependence recovery, we evaluate gene–gene dependence within a fixed cell state and restrict attention to genes with informative occurrence variation in the held-out population. The selection panel is constructed from the most abundant state in the real day-11 population. Among the 200 downstream genes, we retain genes with real within-state occurrence probability between 0.15 and 0.85 and select the 20 with the largest standard deviation of their binary occurrence indicators. The resulting fixed 20-gene panel (190 gene pairs) is applied identically to Real, MeCoGen, and all baselines. This target-defined selection is used only for this qualitative diagnostic and does not enter model fitting, model selection, or the quantitative benchmark.

The main text highlights the state-stratified co-occurrence matrices. Figure 5 provides the full within-state dependence analysis at day 11, including predicted-versus-real pairwise association scatter plots.

MeCoGen retains substantial agreement with the real association structure, with Spearman correlation $\rho = 0.748$ over the selected gene pairs. By comparison, the corresponding correlations are -0.058 for scIMF and 0.034 for scNODE. Importantly, these near-zero baseline correlations should not be interpreted as systematic inversion of the underlying biological associations. Instead, the predicted occurrence masks for the selected genes are nearly fully occupied for both baselines, so the co-occurrence statistics become pseudocount-dominated and provide little measurable dependence structure.

This failure mode is closely related to sparsity preservation. The mean mask occupancy over the selected genes is approximately 0.51 in the real population and 0.59 under MeCoGen, but approximately 0.999 for both scIMF and scNODE. Thus, preserving realistic occurrence structure is necessary not only for matching gene-wise zero fractions but also for retaining informative cross-gene dependence.

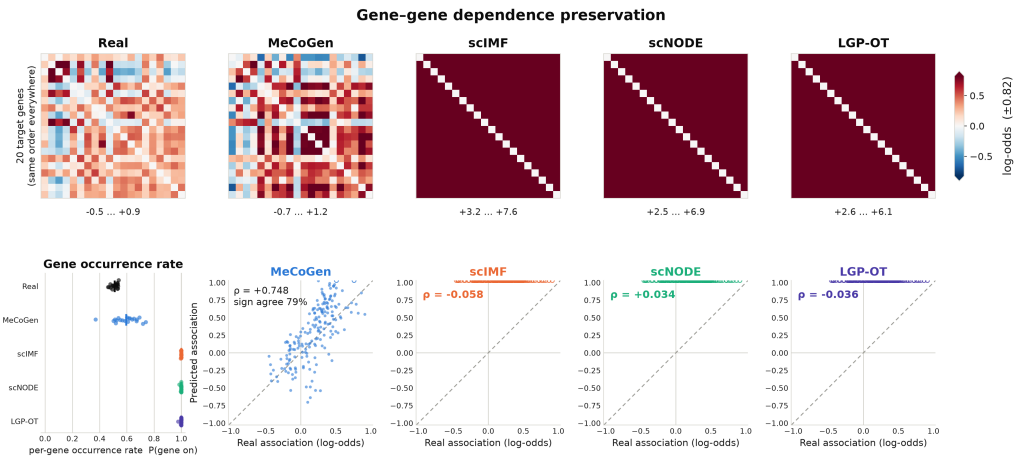

Figure 5: **Sparse gene–gene dependence in a held-out future population.** Top: within-state co-occurrence association matrices for real cells, MeCoGen, scIMF, and scNODE using a shared gene ordering and scale. Bottom: predicted versus real pairwise associations. MeCoGen retains substantial association structure ($\rho = 0.748$), whereas the baseline association estimates become nearly degenerate as their occurrence masks approach complete occupancy.

I.5 Qualitative Validation of the Mechanistic Forecast

Finally, we inspect the structured state produced by the mechanistic component before downstream transcriptome generation. Figure 6 compares representative core-gene trajectories, cell-state proportions, and a population-level positive-expression summary with the corresponding real observations.

The state-composition trajectories remain broadly coherent beyond the training cutoff. In particular, the model captures the disappearance of early IPSC and MES states and the emergence of the divergent late CM and CF populations. This supports its use as a structured future condition for the downstream generator.

Individual gene trajectories are less accurate. For example, TNNT2 is over-extrapolated beyond the observation window, reaching approximately 1.01 at day 15 compared with 0.52 in the real population. The mean positive-expression level over the 200 target genes is also systematically lower than observed. These discrepancies highlight a limitation of the mechanistic forecast: errors in the core genes are retained directly in the final expression panel and can also affect downstream generation through the structured condition. The conditional generator reconstructs the remaining high-dimensional transcriptomic variation but does not correct errors in the mechanistic core.

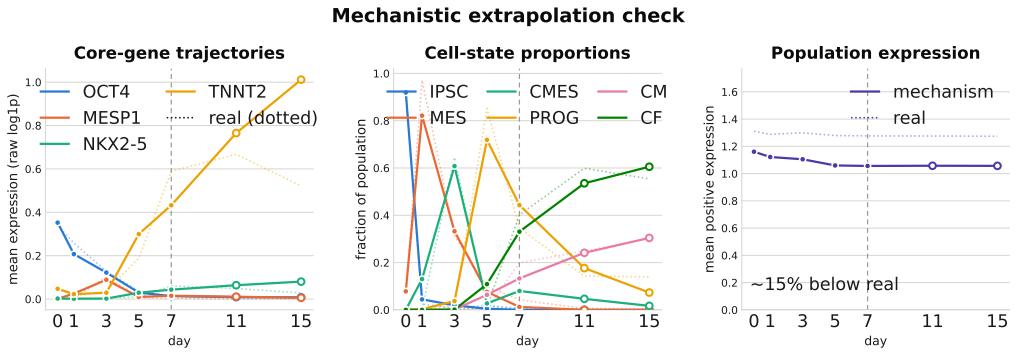

Figure 6: **Qualitative sanity check of mechanistic structured-state extrapolation.** The mechanistic model is fitted on days 0–7 and propagated to held-out days 11 and 15. Shown are representative core-gene trajectories, cell-state proportions, and a population-level positive-expression summary. State composition remains broadly coherent beyond the cutoff, although individual gene trajectories and expression levels exhibit systematic deviations.