DYNAMICVC: A TRAJECTORY-GROUNDED BIOLOGICAL WORLD MODEL FOR VIRTUAL CELL SIMULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Building predictive virtual cells promises to uncover the mechanisms governing cellular behavior. Existing language model-based virtual cell systems rely largely on retrieved knowledge, while continuous generative models are typically evaluated only through their endpoint predictions. We introduce **DynamicVC**, a framework that converts conditional flow-matching trajectories into structured evidence for perturbation reasoning. DynamicVC summarizes control-relative expression changes, vector-field patterns along sampled ODE trajectories, endpoint effects, and directional agreement across stochastic initializations in a **SimContext** (Dynamic **Sim**ulation **C**ontext). It combines model-derived evidence with retrieved biological context to predict whether a drug induces differential expression of a target gene and to generate an evidence-grounded explanation. We evaluate DynamicVC on 55,400 queries involving held-out drugs across five cell lines. It achieves 79.61% AUROC and 64.42% Macro F1, improving over the strongest baseline for each metric by 12.82 pp and 11.06 pp, respectively. Comparisons across reasoning backbones and numerical evidence sources further show that context-specific response predictions consistently improve language-model-based perturbation reasoning. DynamicVC is available at https://anonymous.4open.science/r/DynamicVC-50BF.

1 INTRODUCTION

Virtual cells aim to predict how genetic and chemical perturbations change cellular behavior, offering a computational route to guide experiments and therapeutic discovery (Bunne et al., 2024). Perturb-seq and its genome-scale extensions connect genetic interventions to rich transcriptional readouts, while sci-Plex enables multiplexed chemical-response profiling (Dixit et al., 2016; Replogle et al., 2022; Srivatsan et al., 2020). Reliable prediction depends on the biological context in which an intervention acts. Evaluations show that complex models do not consistently outperform simple baselines (Ahlmann-Eltze et al., 2025), while limited context coverage remains a barrier to transferring perturbation effects (Dibaeinia et al., 2026). These findings motivate methods that make the evidence for a specific cell–perturbation–gene query explicit.

LLM-based approaches formulate perturbation prediction as biological reasoning. PerturbQA evaluates differential expression, response direction, and gene-set enrichment through natural-language tasks (Wu et al., 2025). VCWorld combines drug and gene descriptions, interaction and pathway knowledge, and retrieved experimental analogues to generate structured predictions and mechanistic hypotheses (Wei et al., 2026). Such information can establish a plausible molecular connection, but an analogous observation or pathway association alone does not establish a transcriptional response of the queried gene in the queried cell line. The remaining need is quantitative, context-conditioned evidence that can support or challenge the proposed response.

Generative perturbation models offer a complementary source of evidence. CellFlow and scDFM model perturbed-cell distributions through flow matching (Klein et al., 2025; Yu et al., 2026), while X-Cell combines diffusion modeling with interventional data and biological priors (Wang et al., 2026). In a flow model, ODE integration exposes intermediate states and vector-field values in addition to terminal samples (Lipman et al., 2022). Summaries of these states can describe how predicted deviations evolve along generation; repeated noise initializations can characterize the consistency of those deviations. These are properties of the numerical model, rather than measured biological dy-

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

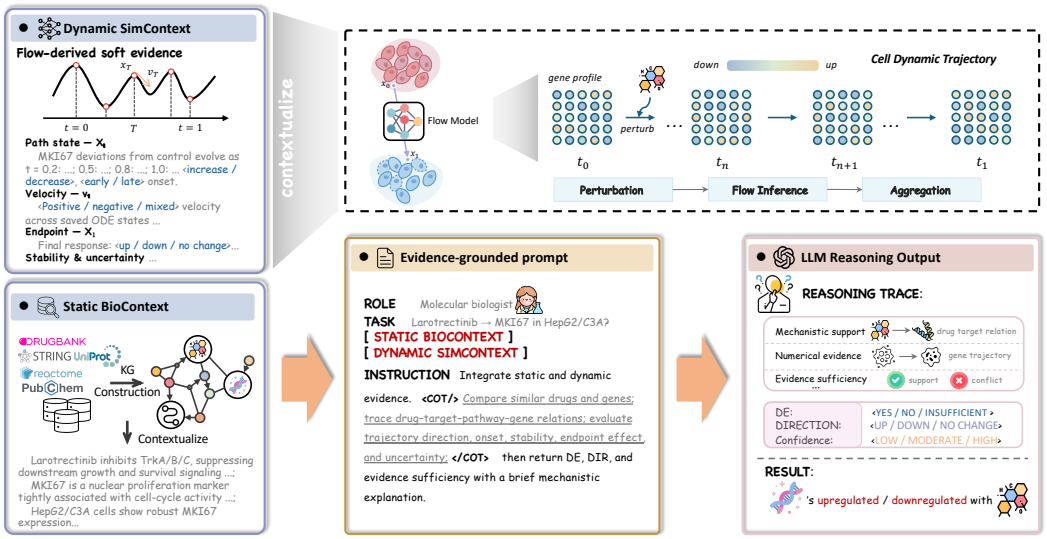


Figure 1: **Overview of DynamicVC.** A conditional flow matching model generates perturbation-response trajectories conditioned on control expression and perturbation identity. DynamicVC summarizes intermediate states, vector-field patterns, endpoint effects, and consistency across initializations into SimContext. This model-derived evidence is combined with Static BioContext in an evidence-grounded prompt, enabling the downstream LLM to assess differential expression, response direction, and evidence sufficiency while providing a mechanistic explanation. The flow coordinate t denotes generative progress rather than biological time, and v_t represents the learned vector field rather than measured RNA velocity. The illustrated query and outputs are schematic.

namics or calibrated confidence estimates. Their potential value for reasoning lies in making the model's response evidence inspectable alongside biological knowledge.

We introduce **DynamicVC**, a trajectory-grounded extension of VCWorld (Figure 1). Using scDFM, we summarize control-relative gene deviations at saved ODE positions, vector-field patterns, endpoint effects, and consistency across initializations and late-flow positions. These features form simulation context (**SimContext**). An optional pathway component organizes trajectory-associated gene sets using direction-specific GO and Reactome enrichment. The downstream LLM receives query-matched evidence together with Static BioContext, source labels, and explicit missing-evidence markers. This interface supports scrutiny of the numerical evidence accompanying a mechanistic hypothesis without treating the hypothesis as experimentally established.

Our contributions are threefold:

- **Trajectory-grounded reasoning.** We introduce DynamicVC, which, for the first time, brings inference-time trajectories from a conditional flow model into LLM-based perturbation reasoning alongside Static BioContext. This grounds LLM reasoning in simulated response dynamics, characterizing not only the terminal state of each gene but also how it deviates from its control state along the trajectory.

- **SimContext, a description language.** We develop SimContext to translate generative trajectories into query-matched, source-labeled evidence describing control-relative response magnitude and direction, vector-field patterns, endpoint effects, and consistency across noise initializations.

- **Empirical evaluation.** On 55,400 shared queries across five cell lines, the gene-level DynamicVC variant achieves 79.61% AUROC and 64.42% Macro-F1, exceeding the strongest respective baselines by 12.82 and 11.06 percentage points.

2 Related Work

2.1 Single-Cell Perturbation Response Prediction

Single-cell perturbation models predict how genetic or chemical interventions change cellular states. scGen combines variational representations with latent perturbation shifts, while the compositional perturbation autoencoder (CPA) combines basal state, intervention, and covariate representations (Lotfollahi et al., 2019; 2023). GEARS uses gene–gene structure to predict unseen genetic combinations, and CellOT learns transport maps between unpaired control and perturbed populations (Roohani et al., 2024; Bunne et al., 2023). CellFlow and scDFM model heterogeneous perturbed-cell populations through conditional flow matching, with scDFM adding distribution alignment to account for population-level response shifts (Klein et al., 2025; Yu et al., 2026). X-Cell combines diffusion modeling with interventional data and multimodal biological priors (Wang et al., 2026), while STATE predicts perturbation effects across cell populations and contexts (Adduri et al., 2026). Context coverage remains a bottleneck for transferring perturbation effects because responses vary across cell types, donors, and activation states (Dibaeinia et al., 2026). Genetic-perturbation benchmarks also show that complex models do not consistently outperform simple baselines (Ahlmann-Eltze et al., 2025). DynamicVC complements numerical response models by exposing intermediate flow traces as evidence for gene-level reasoning.

2.2 Foundation Models for Single-Cell Biology

scVI established a probabilistic framework for modeling noisy single-cell expression (Lopez et al., 2018). Foundation models extend representation learning through pretraining on large observational atlases: scBERT targets cell-type annotation, scFoundation models expression at scale, and Geneformer and scGPT learn transferable gene and cell representations (Yang et al., 2022; Hao et al., 2024; Theodoris et al., 2023; Cui et al., 2024). These representations provide priors for cell identity and co-expression that can support annotation, integration, and perturbation modeling within the broader virtual-cell agenda (Bunne et al., 2024). However, zero-shot evaluations of Geneformer and scGPT identify limitations in cell-type clustering and batch integration (Kedzierska et al., 2025). Such findings motivate task-specific validation; observational pretraining alone does not establish a queried gene's response to a particular intervention and cellular context. DynamicVC obtains this response evidence from a perturbation-conditioned model.

2.3 Generative Paths and Biological Dynamics

Neural ODEs parameterize continuous transformations with learned vector fields (Chen et al., 2018). Diffusion and score-based models generate samples through progressive denoising (Ho et al., 2020; Song et al., 2020), while flow matching learns vector fields along probability paths (Lipman et al., 2022). These paths describe a learned sampling process conditioned on the model and its inputs; their generative coordinate does not itself represent biological time. RNA velocity and its dynamical extensions instead infer transcriptional dynamics from spliced and unspliced RNA under explicit kinetic assumptions (La Manno et al., 2018; Bergen et al., 2020). DynamicVC summarizes numerical flow paths as response evidence without interpreting them as measured biological trajectories.

2.4 Scientific Reasoning Models

Retrieval-augmented generation, chain-of-thought prompting, and ReAct support external evidence, intermediate reasoning, and tool use (Lewis et al., 2020; Wei et al., 2022; Yao et al., 2022). VCWorld applies knowledge retrieval and experimental analogues to perturbation prediction (Wei et al., 2026). In protein science, BioReason-Pro combines protein representations, ontology structure, and biological context for function prediction, while Proteo-R1 converts residue-level reasoning into constraints for diffusion-based design (Fallahpour et al., 2026; Wu et al., 2026). Evidence organization and answer priors can influence LLM decisions (Zhao et al., 2021; Liu et al., 2024). DynamicVC extends VCWorld with SimContext: query-matched trajectory descriptors, consistency summaries, and optional pathway context. Source labels distinguish this model-derived evidence from Static BioContext, allowing the reasoner to assess predicted responses alongside retrieved biological knowledge.

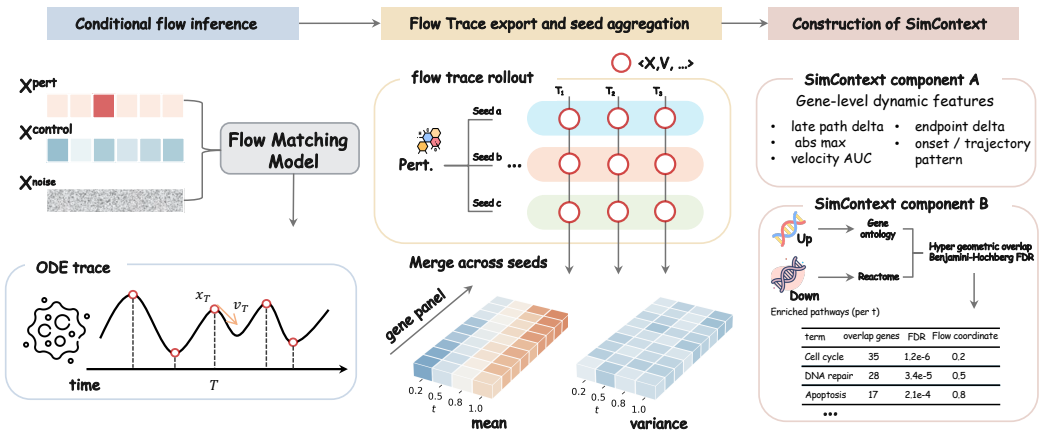


Figure 2: **Constructing SimContext from conditional flow trajectories.** A conditional flow matching model integrates an ODE from noise, conditioned on control expression and perturbation identity. States and vector fields are recorded at selected flow coordinates and aggregated across random seeds to characterize mean responses and variability. The resulting traces are summarized into SimContext, which combines gene-level features—including control-relative deviations, endpoint effects, integrated vector-field values, and trajectory patterns—with pathway-level evidence derived from direction-specific GO and Reactome enrichment analyses.

3 METHODS

3.1 PROBLEM SETUP AND EVIDENCE INTERFACE

We follow the GeneTAK task formulation in VCWorld (Wei et al., 2026), where each query $q = (c, p, g)$ asks whether a gene g is differentially expressed in cell line c under drug perturbation p. The corresponding prediction target is a binary label $y_q \in \{0, 1\}$, while the reasoning model may additionally abstain by returning *insufficient evidence*. VCWorld provides retrieved information about the drug, gene, pathways, interactions, and analogous cases, which we collectively denote as *Static BioContext*, $\mathcal{B}(q)$. DynamicVC complements this retrieved biological context with numerical evidence from a conditional perturbation predictor. Specifically, we construct *SimContext*, $\mathcal{D}(q)$, which summarizes gene-level predictions and their consistency and relates them to pathway-level context before presenting the resulting model-derived evidence to a language model.

3.2 CONTROL-CONDITIONED FLOW EVIDENCE

We instantiate the numerical perturbation predictor with scDFM (Yu et al., 2026), which is based on conditional flow matching (Lipman et al., 2022). Let G denote the set of genes modeled by the predictor. For a fixed cellular context c, we use a panel of N control expression profiles, denoted by $X_c = \{x_i^{\text{ctrl}}\}_{i=1}^{N}$, where i indexes individual control cells and each profile contains expression values for the genes in G. Conditioned on a control profile x_i^{ctrl} and drug perturbation p, the learned vector field transports an initial noise sample toward a predicted perturbed expression profile according to

$$\frac{\mathrm{d}z_{s,i}(t)}{\mathrm{d}t} = v_\theta\big(z_{s,i}(t), t \mid x_i^{\text{ctrl}}, p, G\big), \qquad z_{s,i}(0) = \epsilon_{s,i}, \tag{1}$$

where $z_{s,i}(t)$ denotes the generated expression state associated with control cell i under the s-th random noise initialization, and $s \in \{1, \ldots, S\}$ indexes random draws $\epsilon_{s,i} \sim p_0(\cdot \mid x_i^{\text{ctrl}})$ from the initialization distribution. This notation allows the initialization distribution to be either control-independent or conditioned on the corresponding control profile. The control panel and model parameters are held fixed across noise initializations, and cellular context enters the predictor through the control expression profiles. The coordinate t tracks progress along the generative flow; it is not experimental response time, and the vector field is not measured RNA velocity.

At ordered flow positions $\mathcal{T} = \{t_1, \ldots, t_K\}$ with $t_K = 1$, we record the states, vector-field values, and local endpoint extrapolations

$$\widehat{z}_{s,i}(1 \mid t) = z_{s,i}(t) + (1-t)v_\theta\big(z_{s,i}(t), t \mid x_i^{\mathrm{ctrl}}, p, G\big)\,. \tag{2}$$

We also retain the final integrated prediction, denoted by $\widetilde{z}_{s,i}(1)$ after output post-processing. These quantities form the *FlowTrace*. Path states and vector-field values supply the trajectory descriptors below; the final prediction supplies endpoint evidence. Local extrapolations are retained as complementary numerical records. Predictor training and run-resolved numerical settings are given in Appendix A; implementation details are provided in Appendix B.

3.3 Gene-Level Dynamic Evidence

For each pair (c, p) and gene g, we compare the generated expression profiles with the corresponding control panel. Let $\mu_g^{\mathrm{ctrl}} = \frac{1}{N}\sum_i x_{i,g}^{\mathrm{ctrl}}$ denote the mean control expression of gene g. For random initialization s, its population-averaged deviation at flow position t and the corresponding average across initializations are

$$\Delta_{s,g}(t) = \frac{1}{N}\sum_i z_{s,i,g}(t) - \mu_g^{\mathrm{ctrl}}, \qquad \bar{\Delta}_g(t) = \frac{1}{S}\sum_s \Delta_{s,g}(t). \tag{3}$$

The sequence $\{\bar{\Delta}_g(t) : t \in \mathcal{T}\}$ describes control-relative deviations along the generative path; intermediate states are not measurements of a biological time course. We summarize this trajectory by its maximum amplitude and mean late-flow deviation:

$$A_g = \max_{t\in\mathcal{T}} |\bar{\Delta}_g(t)|, \qquad L_g = \frac{1}{S|\mathcal{T}_{\mathrm{late}}|}\sum_s \sum_{t\in\mathcal{T}_{\mathrm{late}}} \Delta_{s,g}(t), \tag{4}$$

where $\mathcal{T}_{\mathrm{late}} \subseteq \mathcal{T}$ is the specified late-flow window. A context-specific threshold $\tau_{c,p}$, derived from the distribution of A_g across genes in the trace panel, assigns a positive, negative, or near-control direction based on L_g. Separate effect labels describe the relative magnitude of A_g within the same context. These model-derived descriptors are distinct from the benchmark DE labels. In particular, they are not gene-wise significance estimates from an RNA-seq differential-expression procedure such as DESeq2 (Love et al., 2014).

For a gene assigned a direction $d_g \in \{-1, 1\}$, we quantify consistency as the fraction of pairs of noise initializations and late-flow positions that agree with this direction:

$$R_g = \frac{1}{S|\mathcal{T}_{\mathrm{late}}|}\sum_s \sum_{t\in\mathcal{T}_{\mathrm{late}}} \mathbf{1}\{\mathrm{sign}(\Delta_{s,g}(t)) = d_g\}\,. \tag{5}$$

Consistency is left unassessed when neither direction exceeds the threshold. We additionally characterize the trajectory by its shape, the first saved position at which it crosses the directional threshold, and its vector-field pattern. Endpoint evidence is computed separately by averaging the post-processed terminal predictions $\widetilde{z}_{s,i,g}(1)$ across control cells and random initializations and comparing the resulting mean with μ_g^{ctrl}. A dispersion descriptor combines variation across random initializations with dispersion among control and predicted cells; it is a heuristic variability measure, not calibrated predictive uncertainty in the sense of agreement between reported confidence and empirical correctness (Guo et al., 2017). The resulting gene record, $\mathcal{E}_g(c, p)$, contains these numerical summaries and their templated verbal descriptions. Exact labeling, discretization, and dispersion rules are provided in Appendix B.

3.4 Pathway-Level Context

To contextualize the predicted gene-level responses, we construct a pathway record $\mathcal{P}(c, p)$ from the same trajectory summaries. At each saved flow position t, genes are grouped by the sign of $\bar{\Delta}_g(t)$ and selected according to their path-effect and consistency labels. If too few genes satisfy these criteria, we instead retain a bounded set with the strongest signed deviations. The resulting gene sets are tested for over-representation of GO (Aleksander et al., 2023) and Reactome (Milacic et al., 2024) annotations using a hypergeometric model, with the available gene records for (c, p) as the background universe. Benjamini–Hochberg-style multiple-testing adjustment (Benjamini &

Hochberg, 1995) is performed separately for each context, flow position, direction, and annotation source; the complete filtering and adjustment rules are provided in Appendix B.

Each pathway entry records its flow position, direction, annotation, overlapping genes, and adjusted p-value. Any non-significant overlap retained as fallback context is explicitly labeled. These entries indicate putative associations between model-derived gene sets and annotated biological programs; they do not establish pathway activation or biological temporal ordering. The implemented overlap test operates on selected gene sets, unlike the ranked-list enrichment statistic used in GSEA (Subramanian et al., 2005).

3.5 Evidence Integration and Biological Reasoning

For a query $q = (c, p, g)$, we construct the SimContext by serializing the matching gene record, the corresponding pathway context, and evidence-availability metadata:

$$\mathcal{D}(q) = \text{Serialize}(\mathcal{E}_g(c, p), \mathcal{P}(c, p), \mathcal{A}(q)), \tag{6}$$

where $\mathcal{A}(q)$ indicates which evidence components are available, and the serialization preserves source labels for model-derived evidence. Gene-level evidence is matched on (c, p, g), whereas pathway context is shared across queries with the same (c, p). If no matching gene record exists, its absence is stated explicitly; any available pathway context is retained but identified as drug cell-level evidence. This missing-record fallback is distinct from experimental variants that deliberately include or exclude specific evidence components.

The language model receives the query, Static BioContext, and SimContext within the VCWorld reasoning prompt:

$$(r_q, a_q) = \text{LLM}_\phi(q, \mathcal{B}(q), \mathcal{D}(q)), \qquad a_q \in \{\text{yes}, \text{no}, \text{insufficient}\}, \tag{7}$$

where r_q denotes the generated explanation and a_q the requested DE decision. The instructions ask the model to assess numerical response evidence alongside biological priors and explicitly discuss conflicts between them. Evidence constraints are expressed through source labeling, availability information, and reasoning instructions. The gene-level variant uses $\mathcal{D}_{\text{gene}}(q) = \text{Serialize}(\mathcal{E}_g(c, p), \mathcal{A}(q))$. The pathway-augmented variant uses the full context in Eq. 6. Missing gene evidence does not introduce pathway records into the gene-level variant. Prompt field selection, model execution, and the conversion of responses or choice scores into evaluated predictions are specified in the implementation and evaluation protocol in Appendix B.

4 Experiments and Results

We evaluate how SimContext changes perturbation prediction when added to VCWorld's Static BioContext. The experiments first assess overall predictive performance, then compare reasoning backbones and numerical evidence sources, and finally examine the decision behavior and explanatory role of the added evidence.

4.1 Experimental Setup

Task and evaluation scope. Following VCWorld's GeneTAK formulation (Wei et al., 2026), each query asks whether a drug induces differential expression (DE) of a specified gene in a given cell line. Using data derived from Tahoe-100M (Zhang et al., 2025), we evaluate held-out drugs in represented cellular contexts through a five-cell-line benchmark and complementary comparisons of reasoning backbones and numerical evidence sources. Dataset composition, split construction, and query populations are described in Appendix A.

Baselines and evidence variants. We compare DynamicVC with VCWorld and four numerical predictors: STATE (Adduri et al., 2026), CellFlow (Klein et al., 2025), scDFM (Yu et al., 2026), and X-Cell (Wang et al., 2026). DynamicVC uses scDFM to construct gene-level SimContext from control-relative deviations, vector-field patterns, endpoint effects, and consistency across noise initializations and late-flow positions. This condition uses five noise seeds, 12 RK4 integration steps, and saved positions $t \in \{0.2, 0.5, 0.8, 1.0\}$; the optional pathway component is omitted. Backbone comparisons vary the reasoning model, while evidence-source comparisons use Qwen3-4B

with STATE endpoint, scDFM endpoint, CellFlow dynamic, or scDFM dynamic evidence. Detailed configurations are given in Appendix A.

Metrics. The prediction target is binary DE/non-DE. Accuracy uses all evaluation queries, balanced accuracy averages the two class recalls, and Macro-F1 averages the two class F1 scores. Let N_+, N_- denote the true class sizes and $\widehat{N}_+, \widehat{N}_-$ the predicted yes/no counts. Then

$$\mathrm{BAcc} = \frac{1}{2}\left(\frac{TP}{N_+} + \frac{TN}{N_-}\right), \qquad \mathrm{MacroF1} = \frac{1}{2}\left(\frac{2TP}{N_+ + \widehat{N}_+} + \frac{2TN}{N_- + \widehat{N}_-}\right). \tag{8}$$

AUROC and AUPRC assess ranking performance. Main-table metrics are equal-weight means across cell lines; all differences below are in percentage points.

4.2 Main Results

Table 1: **Overall performance on five-cell-line GeneTAK.** The best results are shown in bold, and the second-best results are underlined.

Method	Accuracy↑	Bal. acc.↑	Macro-F1↑	FPR↓	AUROC↑	AUPRC↑
STATE	47.88	52.55	46.65	17.67	53.49	66.75
CellFlow	46.51	42.07	38.33	87.55	52.05	40.98
X-Cell	51.51	51.98	41.13	35.37	56.13	58.70
scDFM	<u>56.52</u>	<u>61.46</u>	52.13	**8.23**	63.13	<u>69.73</u>
VCWorld	51.74	51.77	<u>53.36</u>	36.84	<u>66.79</u>	66.43
DynamicVC	**63.58**	**64.98**	**64.42**	<u>14.28</u>	**79.61**	**79.24**

Scoring conventions are described in Section 4.1.

Joint gains over biological reasoning and numerical prediction. DynamicVC has the highest accuracy, balanced accuracy, Macro-F1, AUROC, and AUPRC in Table 1; scDFM has the lowest FPR. Relative to VCWorld, DynamicVC gains 11.84 points in accuracy and reduces FPR from 36.84% to 14.28%. Relative to scDFM, it gains 7.06 points in accuracy and 16.48 points in AUROC, although its FPR is 6.05 points higher. The two comparisons show complementary effects: predicted response evidence improves the decisions made from biological context, while the reasoning framework improves on the numerical predictor in accuracy and score ranking. Among the baselines, scDFM leads in accuracy and AUPRC, whereas VCWorld leads in Macro-F1 and AUROC. The combined model improves on both across these five measures while retaining a false-positive tradeoff against scDFM.

Improved class balance and score separation. Relative to VCWorld, balanced accuracy rises by 13.21 points and Macro-F1 by 11.06 points, alongside the gain in overall accuracy. The improvement therefore remains visible when DE and non-DE receive equal weight, complementing the overall accuracy result. AUROC and AUPRC also increase by 12.82 and 12.81 points, respectively, showing better ordering of DE and non-DE queries across score thresholds. Thus, the main result combines improved final decisions with improved ranking: SimContext changes both which answers the reasoner selects and how effectively its scores distinguish responsive from non-responsive genes. Class-specific decision behavior is examined in Section 4.4.

4.3 Ablation Study

Additional evidence remains useful across reasoning backbones. All three backbones improve in accuracy, Macro-F1, AUROC, and AUPRC when given SimContext (Table 2). The benefit persists even for the backbone with the highest Static-Only accuracy: GPT-5.6-terra gains 11.62 points, compared with 5.89 for Qwen3-4B and 6.17 for Qwen3.5-27B. Stronger baseline performance therefore does not make the numerical evidence redundant. The pattern across metrics is also informative. The two Qwen models gain approximately six points in both accuracy and AUROC, whereas GPT-5.6-terra gains much more in accuracy than in AUROC (11.62 versus 1.89 points). For GPT-5.6-terra,

Table 2: Backbone comparisons on A-172 cell line (N=20,742)

Backbone	Context	Acc.	Macro-F1	AUROC	AUPRC	ΔAcc.
Qwen3-4B	Static-Only	52.17	54.01	67.82	71.51	–
	+SimContext	58.06	60.55	73.89	77.34	+5.89
Qwen3.5-27B	Static-Only	57.25	58.92	72.47	82.62	–
	+SimContext	63.42	65.23	78.52	85.73	+6.17
GPT-5.6-terra	Static-Only	63.46	62.67	81.53	65.23	–
	+SimContext	75.08	75.06	83.42	73.97	+11.62

the larger change appears in categorical decisions, whereas the Qwen models show similarly sized gains in decisions and discrimination. These differences suggest that the value of the same evidence depends on how the backbone uses it, as well as on the information it contains.

Table 3: Ablation on evidence-source results with Qwen3-4B (N=20,742)

Evidence	Accuracy	AUROC	AUPRC	ΔAcc.
Static-Only	52.17	67.82	71.51	–
STATE endpoint	54.67	70.21	72.99	+2.50
scDFM endpoint	53.85	71.20	73.42	+1.68
CellFlow dynamic	57.07	70.27	72.37	+4.90
scDFM dynamic	**58.06**	**73.89**	**77.34**	**+5.89**

Evidence quality affects decisions and rankings differently. With Qwen3-4B fixed, all four evidence conditions exceed Static-Only in the three metrics reported in Table 3. The endpoint variants already improve score ranking: STATE increases AUROC and AUPRC by 2.39 and 1.48 points, and scDFM endpoint increases them by 3.38 and 1.91 points. The effects on decisions and ranking need not move together. CellFlow dynamic reaches an AUROC of 70.27, close to STATE's 70.21, while its AUPRC of 72.37 is below both endpoint results. scDFM dynamic leads all three reported metrics. Compared with scDFM endpoint, it raises AUROC by 2.69 points and AUPRC by 3.92 points, consistent with trajectory-derived evidence carrying useful information beyond the endpoint estimate. The contrast between CellFlow and scDFM also shows that dynamic evidence quality matters: scDFM dynamic exceeds CellFlow dynamic by 3.62 points in AUROC and 4.97 points in AUPRC.

4.4 Yes Bias and Biological Interpretation

Response evidence makes affirmative decisions more selective. DynamicVC reduces the false-positive rate by 19.92–28.30 points across the five cell lines while increasing DE recall by 3.49–6.20 points (Table 4); the equal-weight mean changes are -22.56 and $+4.78$ points, respectively. The simultaneous changes indicate a better distinction between responsive and non-responsive genes, rather than a uniform shift toward answering no. Two features of knowledge-based reasoning help explain why SimContext can produce this pattern. First, retrieved targets, pathway memberships, and analogous perturbations organize the context around biological associations; their salience can encourage a plausible association to be treated as evidence of a response in the queried cell line. In the A-172 prompt analysis, 63.28% of retrieved-example appearances are positive, rising to 82.02% among examples matching the query gene (Appendix A). Second, language models can exhibit prompt-dependent answer preferences and affirmative tendencies (Zhao et al., 2021; Chen et al., 2023; Braun, 2025). SimContext adds a query-specific check on both influences: a small or inconsistent control-relative effect can weaken an affirmative hypothesis, whereas a substantial, consistent effect can strengthen it. The selected cases in Appendix D illustrate both directions of revision, including Yes→No for Paclitaxel–ESRRG in HOP62 and No→Yes for Encorafenib–AXIN2 in SW480. Together, the aggregate pattern and these examples are consistent with SimContext supporting selective revision, reducing false positives while improving recovery of true responses.

Predicted effects extend the biological explanation. Static BioContext supplies molecular relationships and mechanistic hypotheses; SimContext adds the predicted transcriptional consequences

for the queried drug, gene, and cell line. Their combination connects a proposed mechanism to an expected cellular effect before the final DE decision. Response magnitude helps assess whether the predicted change is appreciable relative to the control, direction makes the implied regulation explicit, and consistency shows how strongly that direction persists across noise initializations and late-flow positions. An explanation can therefore address why a known biological association is expected to produce a response in this context, or why the numerical evidence instead weakens that expectation. Explicit source and availability labels also separate a predicted weak effect from an unavailable prediction. The interpretive contribution is this additional, inspectable link between biological knowledge and predicted response: agreement supplies a more specific rationale for the decision, while disagreement identifies the hypothesis or model prediction that merits further investigation.

Table 4: Class-specific diagnostics for five cell lines.

	False-positive rate		DE recall	
Cell line	Static-Only	DynamicVC	Static-Only	DynamicVC
A-172	39.01	18.39	51.66	57.34
A-427	30.76	10.84	48.23	54.43
HOP62	40.57	12.27	54.89	59.83
NCI-H460	37.02	14.78	47.65	51.14
SW480	36.83	15.14	48.23	51.79
Mean	36.84	14.28	50.13	54.91

4.5 Case Study

Encorafenib–AXIN2 in SW480. At the ODE coordinates $t \in \{0.2, 0.5, 0.8, 1.0\}$, the control-relative expression deviation of AXIN2 increases monotonically from -0.119 to 0.037, 0.226, and 0.369, crossing the control baseline between the first two coordinates. This delayed reversal and widening positive deviation provide dynamic evidence for DE beyond the terminal value alone. Without trajectory evidence, the parsed Static-Only response was No. With SimContext, it changed to Yes, matching the canonical DE truth, while the reported structured choice probability assigned to Yes increased from 0.218 to 0.754. This case is consistent with sustained late positive evidence recovering a missed response.

Paclitaxel–ESRRG in HOP62. The control-relative ESRRG deviations at the four ODE coordinates are 0.078, 0.054, 0.040, and 0.052. All remain small and near control, with a maximum absolute deviation of only 0.078. Rather than moving progressively away from control, the trajectory declines and then shows only a slight terminal rebound, providing limited evidence for a robust DE response. With SimContext, the parsed response changed from Yes to No, matching the canonical non-DE truth, while the reported structured choice probability assigned to No increased from 0.416 to 0.551. This case illustrates how weak, near-control trajectory evidence can reduce a false-positive decision.

5 Conclusion

DynamicVC links generative virtual-cell predictions with biological reasoning by testing whether a proposed drug–gene relationship is supported by a simulated response in the queried cellular context. This combination improves gene-level DE decisions in the evaluated settings and exposes disagreements between numerical predictions and biological knowledge as concrete targets for follow-up experiments.

In future work, we plan to extend DynamicVC beyond held-out drugs in represented cell lines, incorporating pathway-level context into SimContext and evaluating generalization to unseen cell types. Since the generative flow is not biological time, and better predictions or readable explanations do not by themselves establish causal mechanisms or explanation fidelity, we intend to pair DynamicVC's hypotheses with experimental validation. Since errors in predicted responses or retrieved knowledge can yield persuasive yet misleading explanations, we will further develop provenance tracking so that model-derived hypotheses can be audited before any therapeutic use.

REFERENCES

Abhinav K Adduri, Dhruv Gautam, Beatrice Bevilacqua, Mohsen Naghipourfar, Alishba Imran, Rohan Shah, Noam Teyssier, Rishi Verma, Christopher Carpenter, Basak Eraslan, et al. Predicting cellular responses to perturbation across diverse contexts with state. *Cell*, 2026.

Constantin Ahlmann-Eltze, Wolfgang Huber, and Simon Anders. Deep-learning-based gene perturbation effect prediction does not yet outperform simple linear baselines. *Nature Methods*, 22(8): 1657–1661, 2025.

Suzi A Aleksander, James Balhoff, Seth Carbon, J Michael Cherry, Harold J Drabkin, Dustin Ebert, Marc Feuermann, Pascale Gaudet, Nomi L Harris, et al. The gene ontology knowledgebase in 2023. *Genetics*, 224(1):iyad031, 2023.

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*, 57(1):289–300, 1995.

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(12): 1408–1414, 2020.

Daniel Braun. Acquiescence bias in large language models. In *EMNLP (Findings)*, pp. 11341–11355, 2025.

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(11):1759–1768, 2023.

Charlotte Bunne, Yusuf Roohani, Yanay Rosen, Ankit Gupta, Xikun Zhang, Marcel Roed, Theo Alexandrov, Mohammed AlQuraishi, Patricia Brennan, Daniel B Burkhardt, et al. How to build the virtual cell with artificial intelligence: Priorities and opportunities. *Cell*, 187(25):7045–7063, 2024.

Jiangjie Chen, Wei Shi, Ziquan Fu, Sijie Cheng, Lei Li, and Yanghua Xiao. Say what you mean! large language models speak too positively about negative commonsense knowledge. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9890–9908, 2023.

Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

Haotian Cui, Chloe Wang, Hassaan Maan, Kuan Pang, Fengning Luo, Nan Duan, and Bo Wang. scgpt: toward building a foundation model for single-cell multi-omics using generative ai. *Nature methods*, 21(8):1470–1480, 2024.

Payam Dibaeinia, Sudarshan Babu, Mei Knudson, Ali ElSheikh, Yibo Wen, Han Liu, Jason Perera, and Aly A Khan. Virtual cells need context, not just scale. *bioRxiv*, 2026.

Atray Dixit, Oren Parnas, Biyu Li, Jenny Chen, Charles P Fulco, Livnat Jerby-Arnon, Nemanja D Marjanovic, Danielle Dionne, Tyler Burks, Raktima Raychowdhury, et al. Perturb-seq: dissecting molecular circuits with scalable single-cell rna profiling of pooled genetic screens. *cell*, 167(7): 1853–1866, 2016.

Adibvafa Fallahpour, Arman Seyed-Ahmadi, Parsa Idehpour, Omar Ibrahim, Benedict MH Choi, Purav Gupta, Jack Naimer, Kevin Zhu, Abhinav Adduri, Arnav Shah, et al. Bioreason-pro: Advancing protein function prediction with multimodal biological reasoning. *bioRxiv*, pp. 2026–03, 2026.

Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The journal of machine learning research*, 13:723–773, 2012.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pp. 1321–1330. PMLR, 2017.

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(8):1481–1491, 2024.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Kasia Z Kedzierska, Lorin Crawford, Ava P Amini, and Alex X Lu. Zero-shot evaluation reveals limitations of single-cell foundation models. *Genome Biology*, 26(1):101, 2025.

Dominik Klein, Jonas Simon Fleck, Daniil Bobrovskiy, Lea Zimmermann, Sören Becker, Alessandro Palma, Leander Dony, Alejandro Tejada-Lapuerta, Guillaume Huguet, Hsiu-Chuan Lin, et al. Cellflow enables generative single-cell phenotype modeling with flow matching. *bioRxiv*, pp. 2025–04, 2025.

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(7719):494–498, 2018.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474, 2020.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the association for computational linguistics*, 12:157–173, 2024.

Romain Lopez, Jeffrey Regier, Michael B Cole, Michael I Jordan, and Nir Yosef. Deep generative modeling for single-cell transcriptomics. *Nature methods*, 15(12):1053–1058, 2018.

Mohammad Lotfollahi, F Alexander Wolf, and Fabian J Theis. scgen predicts single-cell perturbation responses. *Nature methods*, 16(8):715–721, 2019.

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):MSB202211517, 2023.

Michael I Love, Wolfgang Huber, and Simon Anders. Moderated estimation of fold change and dispersion for rna-seq data with deseq2. *Genome biology*, 15(12):550, 2014.

Marija Milacic, Deidre Beavers, Patrick Conley, Chuqiao Gong, Marc Gillespie, Johannes Griss, Robin Haw, Bijay Jassal, Lisa Matthews, Bruce May, et al. The reactome pathway knowledgebase 2024. *Nucleic acids research*, 52(D1):D672–D678, 2024.

Joseph M Replogle, Reuben A Saunders, Angela N Pogson, Jeffrey A Hussmann, Alexander Lenail, Alina Guna, Lauren Mascibroda, Eric J Wagner, Karen Adelman, Gila Lithwick-Yanai, et al. Mapping information-rich genotype-phenotype landscapes with genome-scale perturb-seq. *Cell*, 185(14):2559–2575, 2022.

Yusuf Roohani, Kexin Huang, and Jure Leskovec. Predicting transcriptional outcomes of novel multigene perturbations with gears. *Nature biotechnology*, 42(6):927–935, 2024.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Sanjay R Srivatsan, José L McFaline-Figueroa, Vijay Ramani, Lauren Saunders, Junyue Cao, Jonathan Packer, Hannah A Pliner, Dana L Jackson, Riza M Daza, Lena Christiansen, et al. Massively multiplex chemical transcriptomics at single-cell resolution. *Science*, 367(6473):45–51, 2020.

Aravind Subramanian, Pablo Tamayo, Vamsi K Mootha, Sayan Mukherjee, Benjamin L Ebert, Michael A Gillette, Amanda Paulovich, Scott L Pomeroy, Todd R Golub, Eric S Lander, et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the national academy of sciences*, 102(43):15545–15550, 2005.

Christina V Theodoris, Ling Xiao, Anant Chopra, Mark D Chaffin, Zeina R Al Sayed, Matthew C Hill, Helene Mantineo, Elizabeth M Brydon, Zexian Zeng, X Shirley Liu, et al. Transfer learning enables predictions in network biology. *Nature*, 618(7965):616–624, 2023.

Chloe Wang, Mehran Karimzadeh, Neal G Ravindra, Lexi R Bounds, Nader Alerasool, Ann C Huang, Shihao Ma, Daniel R Gulbranson, Haotian Cui, Yongju Lee, et al. X-cell: Scaling causal perturbation prediction across diverse cellular contexts via diffusion language models. *bioRxiv*, pp. 2026–03, 2026.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Zhijian Wei, Runze Ma, Zichen Wang, Zhongmin Li, Shuotong Song, and Shuangjia Zheng. Vcworld: A biological world model for virtual cell simulation. In *International Conference on Learning Representations*, volume 2026, pp. 72795–72823, 2026.

Fang Wu, Weihao Xuan, Heli Qi, Hanqun Cao, Heng-Jui Chang, Zeqi Zhou, Haokai Zhao, Ma Jian, Carl Ma, Yu-Chi Cheng, et al. Proteo-r1: Reasoning foundation models for de novo protein design. *arXiv preprint arXiv:2605.02937*, 2026.

Menghua Rachel Wu, Russell Littman, Jacob Levine, Lin Qiu, Tommaso Biancalani, David Richmond, and Jan-Christian Huetter. Contextualizing biological perturbation experiments through language. In *International Conference on Learning Representations*, volume 2025, pp. 73598–73624, 2025.

Fan Yang, Wenchuan Wang, Fang Wang, Yuan Fang, Duyu Tang, Junzhou Huang, Hui Lu, and Jianhua Yao. scbert as a large-scale pretrained deep language model for cell type annotation of single-cell rna-seq data. *Nature machine intelligence*, 4(10):852–866, 2022.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022.

Chenglei Yu, Chuanrui Wang, Bangyan Liao, and Tailin Wu. scdfm: Distributional flow matching model for robust single-cell perturbation prediction. *arXiv preprint arXiv:2602.07103*, 2026.

Jesse Zhang, Airol A Ubas, Richard De Borja, Valentine Svensson, Nicole Thomas, Neha Thakar, Ian Lai, Aidan Winters, Umair Khan, Matthew G Jones, et al. Tahoe-100m: A giga-scale single-cell perturbation atlas for context-dependent gene function and cellular modeling. *BioRxiv*, pp. 2025–02, 2025.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*, pp. 12697–12706. Pmlr, 2021.

A Experimental and Computational Details

This appendix describes data preparation, predictor training, numerical evidence construction, and language-model inference. Appendix B details the evidence-summarization procedures, and Appendix C presents the full benchmark and ablation results.

scDFM checkpoint. The principal scDFM checkpoint was saved at optimization step 100,000 with training seed 1709. Its SHA-256 digest is `f07e346e264993274d770612ce428c30bfc369bca854de7ba1ab9688a78ab7a9`. The corresponding project commit is `2cf6bca1f044e74c4e1dc586892c0495880cf125`.

A.1 Dataset scale and split construction

The experiment used six Tahoe plates and a common 2,000-gene panel. Expression values in the cache were library-size normalized to 10^4 counts per cell and transformed with $\log(1+x)$. Canonical drug identities were partitioned in fold 1 of a five-fold split. The resulting 201 training drugs and 70 held-out drugs were disjoint. Table A.1 summarizes the dataset composition and split.

Table A.1: Audited scale of the formal six-plate Tahoe experiment. A matched group consists of cells matched by cell line, batch, plate, and condition. Counts refer to the immutable cache and split audit used by the formal run.

Item	Value	Definition
Source plates	6	plate3, plate14, plate5, plate6, plate7, and plate9
Single cells	36,747,885	Total observations across the six source plates
Expression panel	2,000 genes	Common processed panel with complete panel coverage in every plate
Cell lines	50	Cellular contexts represented in the cache
Canonical drugs	271	201 train and 70 held-out test drugs
Drug split	Fold 1 of 5	Drug identity is the out-of-distribution split unit
Generalization scope	Held-out drugs	Evaluation in represented cellular contexts
Train drug–cell-line combinations	10,050	201×50 combinations
Test drug–cell-line combinations	3,500	70×50 combinations
Train matched groups	2,423,692	Groups available to the training sampler
Test matched groups	787,443	Groups reserved for held-out evaluation
Descriptor coverage	Complete	A condition descriptor was present for every canonical drug
Train–test drug overlap	None	Canonical training and test drug identities were disjoint
Target-expression access	Evaluation only	Held-out target expression was first accessed after final-checkpoint creation and was not exported into FlowTrace or prompts

Each evaluation query is a cell-line–drug–gene tuple. Table A.2 gives the binary label distributions of the query sets. These counts describe prediction tasks; Table A.1 describes the cells and matched groups used to train and evaluate the expression models.

Leakage control. During trace construction, the inference loader used the `source_control_only` policy. Perturbation conditions were supplied through drug descriptors, and no held-out target expression was included in a trace or language-model prompt. Evaluation targets were unlocked only after the step-100,000 checkpoint had been finalized.

Table A.2: Evaluation-query populations. The main benchmark uses the shared query intersection across compared methods. The complementary A-172 backbone and evidence-source comparisons report the larger population shown in the last column, per condition.

Cell line	Samples	DE	Non-DE	DE (%)	Non-DE (%)
A-172	20,742	11,142	9,600	53.72	46.28
A-427	16,510	6,510	10,000	39.43	60.57
HOP62	19,829	10,229	9,600	51.59	48.41
NCI-H460	21,662	12,062	9,600	55.68	44.32
SW480	18,801	9,001	9,800	47.88	52.12
H4	6,316	2,516	3,800	39.84	60.16
C32	6,475	3,275	3,200	50.58	49.42
Total	110,335	54,735	55,600	49.61	50.39

The five-cell-line main benchmark uses shared query intersections covering 53.11–57.44% of the corresponding full truth sets, with equal weight for each cell line in the reported means. The complementary A-172 analyses use 20,742 queries per condition; the A-172 main-benchmark intersection contains 11,910 queries. The benchmark and ablation populations are detailed in Appendix C.

A.2 Architecture and training of evidence models and baselines

In the main-text comparisons, scDFM, CellFlow, and STATE serve both as standalone numerical baselines and as evidence models for LLM reasoning: scDFM and CellFlow provide dynamic evidence, whereas STATE supplies endpoint evidence. X-Cell is a numerical baseline in the main benchmark. Tables A.3–A.6 summarize their training and execution settings.

scDFM. The conditional flow model uses continuous expression and gene encoders, time embedding, perturbation conditioning, a differential-perceiver fusion module, and an expression decoder. Table A.3 gives the settings for the seed-1709 checkpoint. Its flow-matching and MMD terms are described in Appendix B.

Table A.3: Resolved architecture and optimization settings for the formal scDFM run.

Category	Setting	Value
Architecture	Hidden width	128
	Fusion	Differential perceiver
	Attention	Scaled dot-product attention enabled; 8 attention modules
	Gene panel / genes sampled in training	2,000 / 1,000
	Graph prior	Top-30 observed-neighbor mask with negative edges
	Drug residual dropout	0.35
Optimization	Training seed	1709
	Training length	100,000 optimization steps
	Precision	bfloat16
	Optimizer	Fused AdamW
	Learning-rate schedule	Initial 1×10^{-4}; cosine minimum 1×10^{-6}
	Weight decay	1×10^{-4}
	Gradient clipping	Global norm 1.0
	Global batch	256 cells in 64 matched groups
	Distributed world size	1 (global and per-device batches are identical)
	Data loading	16 workers; prefetch factor 4
Objective	MMD term	Weight 0.5; at most 64 cells
	Geometry term	Weight 0.02
Training output	Scalar logging	Every 50 steps
	Checkpoint interval	Every 5,000 steps
	Selected checkpoint	Step 100,000

CellFlow. CellFlow uses a deterministic pass through the matched training groups, with the sampling and execution settings in Table A.4.

Table A.4: Recorded training and execution settings for the formal CellFlow run.

Setting	Value
Training exposure	One deterministic shuffled pass over 2,423,692 matched groups
Training steps	2,423,692
Cells per group	32 source cells and 32 target cells
Cell draws	77,558,144 draws on each of the source and target sides
Implementation / hardware	JAX on one GPU
Training seed	0
Runtime	12,718 s through checkpoint; 13,341 s total pipeline runtime

STATE. The run used the official STATE architecture with the cell-set-length change in Table A.5.

Table A.5: Recorded training and execution settings for the formal STATE run.

Setting	Value
Architecture	Official STATE architecture; cell-set length changed from 512 to 32
Training exposure	One pass over 2,423,692 matched groups, plus 2 distributed-sampler padding groups
Optimizer steps	12,624
Batch	64 groups per GPU
Learning rate	10^{-4}
Precision / hardware	32-true precision; 3 GPUs
Cell draws	77,558,208 draws on each of the source and target sides

X-Cell. We use an independent reimplementation of the published X-Cell architecture, with settings given in Table A.6.

Table A.6: Recorded training and execution settings for the independent X-Cell reimplementation.

Setting	Value
Architecture	12 layers; width 512; 8 heads; 4 refinement steps
Parameter count	27,144,940
Training exposure	One pass over 2,423,692 matched groups
Optimizer steps	37,871
Batch / sampled cells	64 groups per step; 1 training cell per group
Learning rate / weight decay	10^{-4} / 0.01
Precision / hardware	bfloat16; one NVIDIA H20-3e
Runtime	36,847 s through checkpoint; 40,646 s total pipeline runtime

VCWorld, DynamicVC, and reasoning backbones. VCWorld supplies Static BioContext to a reasoning model; DynamicVC adds SimContext constructed from a numerical predictor. The reasoning backbones include Qwen3-4B, Qwen3.5-27B, and the `gpt-5.6-terra` service alias. Decoding and scoring settings are given in Table A.8.

A.3 Shared training and checkpoint controls

The scDFM, CellFlow, STATE, and X-Cell training configurations share the six-plate Tahoe cache, the 2,000-gene panel, and fold 1 of the drug split: 201 training drugs and 70 disjoint held-out drugs.

For CellFlow, STATE, and X-Cell, "one pass" denotes traversal of the same 2,423,692 training matched groups, including the distributed-sampler padding for STATE. scDFM uses a fixed-step

schedule. Optimization steps, sampled-cell draws, and hardware are reported separately for each model.

A.4 Flow inference, evaluation, and SimContext construction

The learned flow maps Gaussian initial states to a drug-conditioned endpoint while conditioning on matched source-control cells. Inference used a fixed-step fourth-order Runge–Kutta integrator over $[0, 1]$, 12 integration steps, independently seeded Gaussian initial states, bfloat16 mixed precision, and non-negative clipping at the endpoint. The exporter stores the state x_t, the predicted velocity, and the endpoint estimate $\widehat{x}_1$ at the selected coordinates of the learned generative transport.

Expression-level evaluation. The expression-evaluation configuration in Table A.7 combines an FDR threshold of 0.05 with a minimum absolute effect of 0.1. Supplementary aligned-expression exports use FDR-only DE calls at 0.05 and $-\log_{10}(\mathrm{FDR})$ as the ranking score.

Table A.7: Formal scDFM evaluation, FlowTrace, and SimContext settings.

Stage	Setting	Value
Expression evaluation	Cells per matched batch group	32
	Maximum batches per drug–cell combination	2
	Integration steps	12
	Differential-expression rule	Wilcoxon test; FDR $\alpha = 0.05$; minimum absolute delta 0.1
	Ranking / calibration	Top 50; effect temperature 0.05; 15 calibration bins
	Uncertainty	1,000 bootstrap resamples; seed 20260731
	Cell-level reference sampling	128 control cells; DE batch size 32; 4 CPU threads
	Output constraint	Predicted expression clipped below at zero
FlowTrace	Split and data policy	Held-out test; source controls only
	Conditions per completed cell line	70 held-out drugs
	Source sampling	16 source cells per matched group; at most 6 groups per combination
	Stochastic traces	5 trace seeds per condition
	Recorded coordinates	$t \in \{0.2, 0.5, 0.8, 1.0\}$
	Integration / precision	12 steps; bfloat16
	Expression dimensionality	2,000 genes
SimContext	Genes retained per perturbation	At most 2,000
	Program size	Top 20 genes
	Earliest stable coordinate	$t = 0.5$
	Effect quantiles	0.60, 0.80, and 0.95; threshold at 0.60
	Uncertainty quantiles	0.33 and 0.66
	GO/Reactome selection	Fallback top 100 genes; minimum 10 selected; at most 5 terms per group
	Prompt evidence caps	5 program terms and 30 genes
	Enrichment threshold	FDR 0.25
	Retrieval	Budget 10; seed 42

Evidence coverage and computation. Trace and SimContext construction completed for 38 of the 50 cell lines; the remaining 12 were excluded from the coverage totals because trace inference did not complete. Across the 38 completed cell lines, the exporter produced 2,660 held-out drug–cell-line contexts, 13,300 trace-seed files, and 5,320,000 gene-level SimContext rows. Summed trace time was 145,457.5 seconds (40.40 hours); per-cell-line trace time ranged from 3,634.0 to

4,012.6 seconds. The completed prompt sets contain 671,548 paired Static-Only and SimContext queries. Query count per cell line ranged from 10,808 to 21,937, with median 18,513.5. The 12 incomplete cell lines were AN3_CA, C-33_A, COLO_205, HepG2_C3A, Hs_766T, KATO_III, LS_180, MIA_PaCa-2, Panc_03.27, SW_1088, SW_1271, and SW_900.

A-172 produced 350 trace-seed files (70 conditions times 5 seeds), 140,000 gene-level SimContext rows, 4,279 program-enrichment rows, 2,100 prompt-ready rows, and 20,742 benchmark queries. Of these queries, 11,390 had a direct SimContext match and 9,352 used the missing-evidence fallback.

Retrieved-example composition. Across the 20,742 A-172 Static-Only prompts, retrieved examples appear 130,577 times with positive labels and 75,764 times with negative labels; 63.28% of appearances are positive. Among the 8,573 appearances matching the query gene, 7,032 (82.02%) are positive. Counts are weighted by prompt appearance, so an example retrieved for multiple queries contributes multiple times. This concentration of positive examples characterizes the evidence presented to the reasoner. In this prompt set, query-drug and retrieved-example drug identities are disjoint after case and whitespace normalization.

A.5 LANGUAGE-MODEL INFERENCE

All classification prompts request exactly one of three semantic labels: yes, no, or insufficient evidence. Decoding is deterministic ($\mathrm{temperature} = 0$). Table A.8 separates locally hosted vLLM execution from remote API inference and labels utility defaults separately from model-specific settings.

Table A.8: Language-model inference settings. "Structured scoring" denotes a second deterministic request that asks for a parseable yes/no/insufficient decision and differential-expression probability.

Backend	Model / execution	Settings
Local vLLM	Qwen3.5-27B; vLLM 0.19.0	bfloat16; tensor parallelism 2; maximum sequence length 8,192; seed 42; temperature 0; GPU-memory utilization 0.88; prefix caching and chunked prefill enabled; eager execution; safetensors weights. A-172 SimContext inference used batch size 256. Generation budgets are 512 or 2,048 tokens across configurations.
Local vLLM utility	Qwen3-4B-Instruct-2507 and other local checkpoints	Utility defaults: temperature 0, seed 42, maximum 512 new tokens, batch size 8, and tensor parallelism 1. Execution variants use batch size 256 and tensor parallelism 1 or 2. Optional deterministic choice scoring uses a one-token decision pass and first-token scores.
OpenAI-compatible API	`gpt-5.6-terra` service alias	Temperature 0; top-$p = 1$; concurrency 8; request timeout 90 s; at most 3 retries after the initial attempt; resume enabled; structured scoring enabled. The unanswered-prompt retry used 2,048 maximum generation tokens and 256 maximum scoring tokens.
Gemini API utility	Default `gemini-2.5-flash`	Temperature 0; concurrency 2; request timeout 90 s; hard timeout 150 s; at most 3 retries; launch interval 0.2 s; resume and choice scoring enabled. Utility defaults for this auxiliary backend are shown here.

A.6 API REQUEST VOLUME

Table A.9 quantifies the request volume of the A-172 API comparison with 20,742 Static-Only prompts and 20,742 SimContext prompts. Each successful generation was followed by a structured scoring request. A second pass retried 9,683 unanswered Static-Only prompts and 6,865 unanswered SimContext prompts. Counts refer to logical generation and scoring calls; each call permits up to four HTTP attempts under the retry policy.

Table A.9: Audited A-172 API volume and character-based token proxy. Token volumes use the explicit approximation of four Unicode characters per token

	API calls		Tokens (M)	
Prompt set	Generation	Scoring	Input	Output
Static-Only	20,742	20,733	128.103	20.578
SimContext	20,742	20,742	135.949	21.141
Total	41,484	41,475	264.052	41.719

B Implementation and Evidence-Summarization Details

This appendix specifies the trace representation, gene-level features, optional program evidence, and prediction channels used by the implementation. Experimental settings are listed in Appendix A; defaults for the earlier trace utility are listed separately in Table B.1.

B.1 Inherited Flow-Matching Training

The simulator follows scDFM (Yu et al., 2026); DynamicVC adds trace export and evidence construction. For the Gaussian configuration, training samples $\epsilon \sim \mathcal{N}(0, I)$ and $t \sim \mathcal{U}(0, 1)$ and uses

$$x_t = (1-t)\epsilon + ty, \qquad u_t = y - \epsilon, \qquad \mathcal{L}_{\mathrm{FM}} = \mathbb{E}\big[|G|^{-1}\|v_\theta(x_t, t \mid x^{\mathrm{ctrl}}, p, G) - u_t\|_2^2\big], \quad \text{(B.1)}$$

where y is perturbed expression and x^{ctrl} is a sampled control profile. The implementation samples controls and targets independently; its optional optimal-transport pairing call is inactive. The training data and model settings are described in Appendix A.

When enabled, regularization compares the minibatch of local extrapolations $\hat{y} = x_t + (1-t)v_\theta$ with observed targets using the mean of four unbiased squared MMD estimators (Gretton et al., 2012). The Gaussian kernels have

$$k_j(a, b) = \exp\left[-\frac{\|a-b\|_2^2}{2\sigma_j^2}\right], \quad \sigma_j^2 = a_j \ \mathrm{median}_{i \neq i'} \|y_i - y_{i'}\|_2^2, \quad a_j \in \{0.5, 1, 2, 4\}. \quad \text{(B.2)}$$

Within-sample kernel diagonals are excluded. The equations above define the flow-matching and MMD components; the scDFM training configuration is listed in Table A.3.

B.2 Trace Export

In the earlier trace exporter, within each selected context, a fixed panel of $N = \min(N_{\mathrm{ctrl}}, N_{\max})$ controls is sampled without replacement using selection seed zero. This panel is reused across perturbations and noise seeds. The model runs in evaluation mode; seeds $s = 0, \ldots, S-1$ change initial noise, not model parameters or the control panel. A context label selects control cells through the grouping field; it is not an additional input to the velocity network. An empty grouping field pools controls.

For this exporter, let $X_c = \{x_i^{\mathrm{ctrl}}\}_{i=1}^N$. Inference solves $\dot{z}_{s,i}(t) = v_\theta(z_{s,i}(t), t \mid x_i^{\mathrm{ctrl}}, p, G)$ with $z_{s,i}(0) = \epsilon_{s,i} \sim p_0(\cdot \mid x_i^{\mathrm{ctrl}})$. RK4 uses $\{0\} \cup \mathcal{T} \cup \{1\}$ as its integration grid, with no separate step size. This produces four RK4 steps at the default saved positions; the experimental configuration instead uses 12 integration steps (Table A.7). States and local extrapolations $\hat{z}_{s,i}(1 \mid t)$ are untruncated, whereas the terminal output is $\tilde{z}_{s,i}(1) = \max(z_{s,i}(1), 0)$ coordinatewise.

Table B.1: Defaults of the inspected legacy utility. Actual formal-run settings are reported in Appendix A.

Setting	Code default
Initial noise	Standard Gaussian
Saved positions $\mathcal{T}$	$0.2, 0.5, 0.8, 1.0$
Maximum controls / noise seeds	128 / 5
Trace grouping / gene panel	Pooled / all available panel genes
Trace export / cell-level storage	Disabled / disabled
Late-position cutoff	0.5
Effect quantiles / direction threshold	$(0.60, 0.80, 0.95)$ / 0.60
Uncertainty quantiles	$(0.33, 0.66)$
MMD switch / weight	Disabled / $\gamma = 0$
Gene-row limit / top-gene lists	Unlimited / disabled

"All" denotes the simulator's available gene panel. Evidence construction uses source controls and simulated responses; held-out target expression is reserved for evaluation.

B.3 Gene-Level Features

Let $b_g = N^{-1}\sum_i x_i^{\mathrm{ctrl}}(g)$, $m_{s,t,g} = N^{-1}\sum_i z_{s,i}(t,g)$, and $V_{s,t,g} = N^{-1}\sum_i v_{s,i}(t,g)$. The primary deviation is

$$\Delta_{s,t,g} = m_{s,t,g} - b_g, \qquad \bar{\Delta}_{t,g} = S^{-1}\sum_s \Delta_{s,t,g}. \tag{B.3}$$

Thus averaging first occurs over simulated cells, then over noise seeds. Local endpoint extrapolations are stored separately and do not determine the primary trajectory labels. Define $\mathcal{T}_{\mathrm{late}} = \{t \in \mathcal{T} : t \geq 0.5\}$, $L_{s,g} = |\mathcal{T}_{\mathrm{late}}|^{-1}\sum_{t\in\mathcal{T}_{\mathrm{late}}} \Delta_{s,t,g}$, $L_g = S^{-1}\sum_s L_{s,g}$, and $A_g = \max_t |\bar{\Delta}_{t,g}|$. If the late set is empty, all saved positions are used.

For each perturbation–context group, let q_r be the empirical r-quantile of $\{A_g\}$ over its available genes, before optional row truncation. Magnitudes below $q_{.60}$, $q_{.80}$, and $q_{.95}$ receive weak, small, and moderate labels; remaining values are large. With $\tau_{cp} = q_{.60}$, direction is positive or negative when $L_g > \tau_{cp}$ or $L_g < -\tau_{cp}$; otherwise it is unclear. Endpoint direction instead uses $S^{-1}\sum_s[N^{-1}\sum_i \tilde{z}_{s,i}(1,g) - b_g]$, with the same threshold. These are relative descriptive labels, not differential-expression tests.

For a nonzero direction $d_g \in \{-1, 1\}$, stability is

$$R_g = \frac{1}{S|\mathcal{T}_{\mathrm{late}}|}\sum_s \sum_{t\in\mathcal{T}_{\mathrm{late}}} \mathbf{1}\{\mathrm{sign}(\Delta_{s,t,g}) = d_g\}. \tag{B.4}$$

The stable/partially-stable/unstable cutoffs are 0.8 and 0.5. Unclear directions are not assessed. This statistic measures directional agreement jointly across noise seeds and late positions.

Using population standard deviations, the uncertainty proxy is

$$\begin{aligned} U_g = {} & \mathrm{sd}_s(L_{s,g}) + \tfrac{1}{4}\sqrt{S^{-1}\sum_s \mathrm{Var}_i(\tilde{z}_{s,i}(1,g))} \\ & + \tfrac{1}{4}\,\mathrm{sd}_i(x_i^{\mathrm{ctrl}}(g)) + \tfrac{1}{4|\mathcal{T}|}\sum_{t\in\mathcal{T}} \mathrm{sd}_s(\Delta_{s,t,g}). \end{aligned} \tag{B.5}$$

Low/medium/high labels use within-group $0.33/0.66$ quantiles, with equality assigned to the lower category. With fewer than two seeds, uncertainty is marked not estimated. The proxy combines sensitivity to initial noise with dispersion within the source and simulated cell populations.

B.4 Shape, Onset, and Velocity Rules

Shape labels are assigned in order: flat when all deviations are below threshold; reversal when threshold-exceeding deviations change sign; then direction-specific monotonic, delayed, or transient patterns; otherwise mixed. Monotonicity permits at least 75% direction-consistent increments after discarding increments smaller than $q_{.60}/4$ (with a 10^{-8} floor). Delayed patterns start below threshold and end beyond it; transient patterns return below half their peak magnitude. Each label summarizes the pattern across saved flow positions.

Onset is the first zero-based index k satisfying $d_g\bar{\Delta}_{t_k,g} \geq \tau_{cp}$, categorized by $k/\max(|\mathcal{T}| - 1, 1)$ into early (≤ 0.34), middle (≤ 0.67), or late; absent crossings or unclear directions are labeled none. It is an index-based descriptor, including for nonuniform grids.

Velocity intensity is the trapezoidal integral of $S^{-1}\sum_s |V_{s,t,g}|$ over saved positions, excluding any unsaved initial interval. It takes absolute values after cell averaging. A below-0.60-quantile intensity or negligible mean velocity is weak. Otherwise, 75% agreement between mean-velocity signs and d_g yields sustained positive/negative velocity. Remaining cases compare mean absolute velocities in the first and second index halves: a ratio above 1.5 yields early/late push; failing these tests, a last magnitude below half the first is decelerating, and otherwise fluctuating.

B.5 Optional Program Evidence and Prompt Fields

The optional program-evidence branch performs over-representation analysis separately by perturbation, context, saved position, direction, and knowledge source. Candidate genes have the re-

quested deviation sign, moderate/large path effects, and non-unstable labels. If fewer than ten qualify, selection falls back to at most 100 genes with the strongest deviations in the requested direction. The background is the genes retained in that SimContext group. After intersection with this background, Reactome term sizes must be 3–200 with overlap at least three; GO term sizes must be 5–100 with overlap at least four, excluding generic terms.

Upper-tail hypergeometric probabilities receive a BH-style monotone adjustment. Let M count all eligible size-filtered terms and $p_{(1)}, \ldots, p_{(H)}$ be sorted probabilities only for hits passing the minimum-overlap filter. The implementation uses

$$q_{(r)} = \min\left(1, \min_{j=r,\ldots,H} \frac{M p_{(j)}}{j}\right). \tag{B.6}$$

Ranks use retained hits, whereas the test-count factor uses all eligible terms. The earlier utility retains at most ten terms per group at $q \leq 0.25$; when none qualify, top overlaps are marked as non-significant context. The experimental program-export configuration caps both retained terms and prompt terms at five per group (Table A.7). The main gene-level SimContext condition omits this optional program branch.

Joins normalize drug names to lowercase with collapsed whitespace, genes to uppercase, and cell lines to lowercase after removing common suffixes. Injected gene evidence contains path-state, velocity, and endpoint sentences; trajectory direction, effect, shape, onset, stability, and uncertainty labels; and optional top-gene lists. Program statements provide shared drug–cell evidence, including when a query gene lacks a direct match. The direct injector assembles these fields into evidence sentences. The separate prompt-ready exporter additionally filters genes and applies an output budget.

B.6 LLM Execution and Prediction Channels

The runners provide generated-answer parsing and choice scoring as separate output channels. The local vLLM runner scores candidate completions using a sum, average, or first-token log-probability. The API runner obtains choice scores through a separate multiple-choice request. Generated labels represent the parsed textual answer, whereas choice labels are selected from the scored candidates. Decoding configurations are listed in Appendix A, and the channels represented in the supplementary evaluations are defined in Appendix C.

C SUPPLEMENTARY BENCHMARK AND ABLATION RESULTS

Table C.1: Current benchmark and ablation results (Panels A–B) under the binary DE evaluation protocol.

Method/ backbone	Context/ evidence	N	N_s	Acc. ↑	Macro-F1 ↑	DE rec. ↑	FPR ↓	AUROC ↑	AUPRC ↑	Ins./Unres. ↓
A. Main benchmark: equal-weight means over five cell lines										
STATE	Standalone	55,400	55,400	47.88	46.65	38.77	17.67	53.49	66.75	0.00
CellFlow	Standalone	55,400	55,400	46.51	38.33	71.69	87.55	52.05	40.98	0.00
X-Cell	Standalone	55,400	55,400	51.51	41.13	59.34	35.37	56.13	58.70	0.00
scDFM	Standalone	55,400	55,400	56.52	52.13	31.15	8.23	63.13	69.73	0.00
VCWorld	Static-Only	55,400	55,400	51.74	53.36	50.13	36.84	66.79	66.43	6.03
DynamicVC	SimContext	55,400	55,400	63.58	64.42	54.91	14.28	79.61	79.24	8.74
B. A-172 full LLM query set: Qwen3-4B evidence comparison										
Qwen3-4B	Static-Only (none)	20,742	20,742	52.17	54.01	51.66	38.11	67.82	71.51	7.01
Qwen3-4B	STATE endpoint	20,742	20,742	54.67	55.02	52.80	16.48	70.21	72.99	1.58
Qwen3-4B	scDFM endpoint	20,742	20,742	53.85	54.79	52.10	32.51	71.20	73.42	7.53
Qwen3-4B	CellFlow dynamic	20,742	20,742	57.07	54.13	53.10	15.12	70.27	72.37	1.29
Qwen3-4B	scDFM dynamic	20,742	20,742	58.06	60.55	54.59	18.39	73.89	77.34	8.21

D SELECTED RESPONSE-SHIFT CASES

Table D.1 presents 13 selected Qwen3-4B response shifts after adding SimContext. The rows comprise 10 unique drug–gene pairs: nine pairs shown for one cell line each and four cell-line-specific instances of Pasireotide–HES1. Using the canonical binary DE truth described in Section 4.1, the parsed Static-Only response was incorrect or insufficient for every row, whereas the parsed SimContext response was correct.

All 13 rows have a positive change in the probability assigned to the truth class. The mean change is 0.487 across these selected cases.

Table D.1: Selected Qwen3-4B response shifts after adding SimContext. $P(\mathrm{truth})$ denotes the probability assigned to the canonical DE truth class.

Cell line	Drug–gene	Truth	Response shift	$P(\mathrm{truth})$: O→S	ΔP
SW480	Encorafenib–AXIN2	Yes	No→Yes	0.218→0.754	+0.536
HOP62	Paclitaxel–ESRRG	No	Yes→No	0.416→0.551	+0.135
SW480	Clotrimazole–NKD1	Yes	No→Yes	0.096→0.685	+0.589
A-172	Pasireotide (acetate)–HES1	Yes	No→Yes	0.067→0.568	+0.502
NCI-H460	Pasireotide (acetate)–HES1	Yes	No→Yes	0.136→0.592	+0.456
HOP62	Pasireotide (acetate)–HES1†	Yes	Insufficient→Yes	0.270→0.553	+0.282
SW480	Pasireotide (acetate)–HES1	Yes	No→Yes	0.350→0.610	+0.259
A-427	Paclitaxel–PCDH9	Yes	No→Yes	0.111→0.713	+0.602
SW480	Encorafenib–DLEU1	Yes	No→Yes	0.095→0.708	+0.613
A-172	Fenofibrate–KCNQ5	Yes	No→Yes	0.063→0.683	+0.620
NCI-H460	Pasireotide (acetate)–FN1	Yes	No→Yes	0.137→0.651	+0.513
A-427	Fenofibrate–NR4A3	Yes	No→Yes	0.059→0.731	+0.672
HOP62	Hydroxyfasudil–TEX49†	No	Yes→No	0.202→0.749	+0.547

Notes. O, Original; S, SimContext. Response shifts use the parsed `pred_label` field; probabilities are taken from the structured choice-scoring output. Displayed probabilities are rounded to three decimal places, whereas ΔP is computed from unrounded outputs. † For the two HOP62 Original outputs, the parsed and choice labels differ: HES1 has `pred_label=Insufficient` and `choice_pred_label=No`; TEX49 has `pred_label=Yes` and `choice_pred_label=Insufficient`.